INTERNATIONAL WORKSHOP

NATURAL LANGUAGE PROCESSING METHODS AND CORPORA IN TRANSLATION, LEXICOGRAPHY, AND LANGUAGE LEARNING

held in conjunction with the International Conference

RANLP - 2009, 14-16 September 2009, Borovets, Bulgaria

PROCEEDINGS

Edited by
Iustina Ilisei, Viktor Pekar and Silvia Bernardini

Borovets, Bulgaria

17 September 2009

International Workshop

NATURAL LANGUAGE PROCESSING METHODS AND CORPORA IN TRANSLATION, LEXICOGRAPHY, AND LANGUAGE LEARNING

PROCEEDINGS

Borovets, Bulgaria
17 September 2009

ISBN 978-954-452-010-6

Designed and Printed by INCOMA Ltd.
Shoumen, Bulgaria

Foreword

In recent years corpora have become an indispensable tool in research and everyday practice for translators, lexicographers, second language learners. Specialists in these areas share a general goal in using corpora in their work: corpora provide the possibility of finding and analysing linguistic patterns characteristic of various kinds of language users, monitoring language change, and revealing important similarities and divergences across different languages.
By this time, Natural Language Processing (NLP) technologies have matured to the point where much more complex analysis of corpora becomes possible: more complex grammatical and lexical patterns can be discovered, and new, more complex aspects of text (pragmatic, stylistic, etc.) can be analysed computationally.
For professional translators, corpora represent an invaluable linguistic and cultural awareness tool. For language learners, they serve as a means to gain insights into specifics of competent language use as well as to analyse typical errors of fellow learners. For lexicographers, corpora are key for monitoring the development of language vocabularies, making informed decisions as to lexicographic relevance of the lexical material, and for general verification of all varieties of lexicographic data.

While simple corpus analysis tools such as concordancers have long been in use in these specialist areas, in the past decade there have been important developments in Natural Language Processing technologies: it has become much easier to construct corpora, and powerful NLP methods have become available that can be used to analyse corpora not only at the surface level, but also at the syntactic, and even semantic, pragmatic, and stylistic levels.

We believe that 2009 was an appropriate moment for the RANLP workshop on Natural Language Processing Methods and Corpora in Translation, Lexicography, and Language Learning. It presented recent studies covering the following topics: term and collocation extraction, corpora in translator training, construction of lexical resource, lexical substitution techniques, word alignment, and automatic tree alignment. The event was complemented by two invited speakers who presented several studies where NLP methods and corpora have proved to be helpful.
The workshop brought together the developers and the users of NLP technologies for the purposes of translation, translation studies, lexicography, terminology, and language learning in order to present their research and discuss new possibilities and challenges in these research areas.
We are grateful to the organisers of the Seventh International Conference on Recent Advances in Natural Language Processing, RANLP 2009, for holding this workshop in conjunction to the main conference. We are also thankful to the Programme Committee for their commitment and support in the reviewing process and to the researchers who submitted papers to this workshop.

Iustina Ilisei, Viktor Pekar, and Silvia Bernardini

17th September, 2009

Programme Committee

Marco Baroni, University of Trento
Jill Burstein, Educational Testing Service
Michael Carl, Copenhagen Business School
Gloria Corpas Pastor, University of Málaga
Le An Ha, University of Wolverhampton
Patrick Hanks, Masaryk University
Federico Gaspari, University of Bologna
Adam Kilgarriff, Lexical Computing
Marie-Claude L'Homme, Université de Montréal
Ruslan Mitkov, University of Wolverhampton
Roberto Navigli, University of Rome "La Sapienza"
Miriam Seghiri, University of Málaga
Pete Whitelock, Oxford University Press
Richard Xiao, Edge Hill University
Federico Zanettin, University of Perugia

Organising Committee

Iustina Ilisei, University of Wolverhampton, United Kingdom
Viktor Pekar, Oxford University Press, United Kingdom
Silvia Bernardini, University of Bologna, Italy

Table of Contents

Conference Program

Thursday, September 17, 2009

10:30 - 11:00 Finding Domain Specific Collocations and Concordances on the Web
Caroline Barrière

11:00 - 11:30 HMMs, GRs, and N-Grams as Lexical Substitution Techniques – Are They Portable to Other Languages?
Judita Preiss, Andrew Coonce and Brittany Baker

11:30 - 13:00 The Web a Corpus: Going Beyond Page Hit Frequencies
Preslav Nakov (invited talk)

13:00 - 14:00 Lunch Break

14:00 - 14:30 Unsupervised Construction of a Multilingual WordNet from Parallel Corpora
Dimitar Kazakov and Ahmad R. Shahid

14:30 - 15:00 Search Techniques in Corpora for the Training of Translators
Verónica Pastor and Amparo Alcina

15:00 - 16:00 Translation Universals: Experiments on Simplification, Convergence and Transfer
Gloria Corpas and Ruslan Mitkov (invited talk)

16:00 - 16:30 Break

16:30 - 17:00 Evidence-Based Word Alignment
Jörg Tiedemann

17:00 - 17:30 A Discriminative Approach to Tree Alignment
Jörg Tiedemann and Gideon Kotzé

Finding domain specific collocations and concordances on the Web

Caroline Barrière
National Research Council of Canada
Gatineau, QC, Canada
caroline.barriere@nrc-cnrc.gc.ca

Abstract

TerminoWeb is a web-based platform designed to find and explore specialized domain knowledge on the Web. An important aspect of this exploration is the discovery of domain-specific collocations on the Web and their presentation in a concordancer to provide contextual information. Such information is valuable to a translator or a language learner presented with a source text containing a specific terminology to be understood. The purpose of this article is to show a proof of concept that TerminoWeb, as an integrated platform, allows the user to extract terms from the source text and then automatically build a related specialized corpus from the Web in which collocations will be discovered to help the user understand the unknown specialized terms.

Keywords

term extraction, collocation extraction, concordancer, Web as corpus, domain-specific corpus

1. Introduction

Collocations and concordances found in corpora provide valuable information for both acquiring the sense and usage of a term or word. Corpora resources are usually complementary to dictionaries, and provide a more contextual understanding of a term. Collocations and concordances are rarely viewed as "static" resources, the way dictionary definitions would, but are rather often considered the disposable result of a tool's process (a concordancer, a collocation extractor) on a particular corpus.

The use and value of corpora for vocabulary acquisition and comprehension is quite known. In language learning mostly [2], its use obviously has advantages and disadvantages compared to dictionaries, and its context of usage might influence its value (self-learning or classroom). Early work on vocabulary acquisition [21] argued that the learning of a new word is frequently related to the incidence of reading or repetitive listening. Even earlier [23], one experiment illustrated that the frequency of a word and the richness of the context facilitates the identification of a word by a novice reader. Even so, computer-assisted techniques for the understanding of unknown words [15] in second language learning are still not widely studied.

In translation studies, the value of corpora has been repeatedly shown [6, 19, 22, 28] and concordancers are the tools of choice for many translators to view a word in its natural surrounding.

Concordances are usually defined clearly as a window of text surrounding a term or expression of interest. Most often, a fixed small window size is established (ex. 50 characters) and the results are called KWIC (KeyWord In Context). Such KWIC views are usually supplemented one-click away by a larger context view (a paragraph), potentially even another click away to access the source text.

Collocations are words which tend to co-occur with higher than random probability. Although conceptually the definition is quite simple, results will largely differ because of two main variables. A first variable is the window size in which co-occurrences are measured. A small window (2-3 words maximum before or after) is usually established for collocations. Longer distances are considered associations, or semantically-related words, which tend to be together in sentences or paragraphs or even documents. A second variable is the actual measure of association used, and there have been multiple measures suggested in the literature, such as Overlap, Mutual Information, Dice Coefficient, etc [10][1].

Even more fundamentally, one key element will largely influence the results of both concordancers and collocation extractors: the source corpus. For the general language, the BNC (British National Corpus) has been widely used by corpus linguists, and recently a Web interface has been developed (BNCweb) to access it [14].

Domain-specific corpora or corpora in other languages than English are not as easily found[2], especially not packaged with a set of corpus analysis tools. The notion of "disposable" or "do-it-yourself" corpora has been suggested as a corpus that translators would build themselves to quickly search for information [7, 26]. Language learners would also often be in need of domain-specific corpora. But the problem resides in the overhead work involved in building such corpus. A process of selection, upload

[1] For detailed explanations and a tutorial on multiple association measures: http://www.collocations.de/AM/

[2] See Serge Sharoff's multi-language Web copus collection (http://corpus.leeds.ac.uk/internet.html).

(for personal texts) or download (for Web texts) and management (easy storage and retrieval of texts) is involved. Only a few tools exist for such purpose, such as Corpografo [20] and TerminoWeb [5].

This paper presents a new version of the TerminoWeb system[3] which provides the user with the capability of automatically building a "disposable" domain-specific corpus and study some terms by finding collocations and concordances in that corpus. Different steps are necessary for such task. Section 2 presents a working scenario with a single detailed example to explain the algorithms underlying each step. Section 3 links to related work for different aspects of the system, although we do not know of any system which integrates all the modules as TerminoWeb does. Section 4 concludes and hints at some future work.

2. Collocations and concordances

Becoming familiar with the vocabulary in a source text is essential both for reading comprehension (a language learner's task) and text translation (a translator's task).

The understanding process for unknown terms/words could rely on a search for appropriate definitions in a dictionary, as well as a search for collocations and concordances in corpus. We look at this second option and present the following scenario to perform the discovery of collocations and concordances:

1) Source text upload
2) Term extraction on source text
3) Query generation from terms
4) Domain-specific corpus construction
5) Collocations and concordances search

Step 1. Text upload

We take as a starting point a text to translate or a text to understand. TerminoWeb provides an interface for the user to upload (copy-paste) the source text. For illustrating purpose, we arbitrarily take a text on banking fraud issues (http://bankfraudtoday.com/).

Step 2. Term extraction

The term extractor finds single-word and/or multi-word terms in the source document. The number of terms to be found can be set by the user, or estimated automatically based on the document's length and the actual term statistics. The term extraction module implements Smadja's algorithm [25] which is purely statistical and based on frequencies. Such a purely statistical approach has the advantage of being largely language independent, with only a list of stop words necessary for each different language.

TerminoWeb allows term sorting in alphabetical or frequency order, but Figure 1 shows a sample of terms from the uploaded document on bank fraud ordered by specificity. Specificity is approximated by a "hit count" measure which we discuss in the next step of query generation.

Step 3. Query generation

This step is to launch a set of queries on the Web to find documents that are both domain specific (related to the source text) and containing information about the unknown words (words less familiar to the language learner or the translator). The purpose of the query generation (QG) module is to make this task relatively easy for the user. Nevertheless, the following factors which will impact the results must be understood:

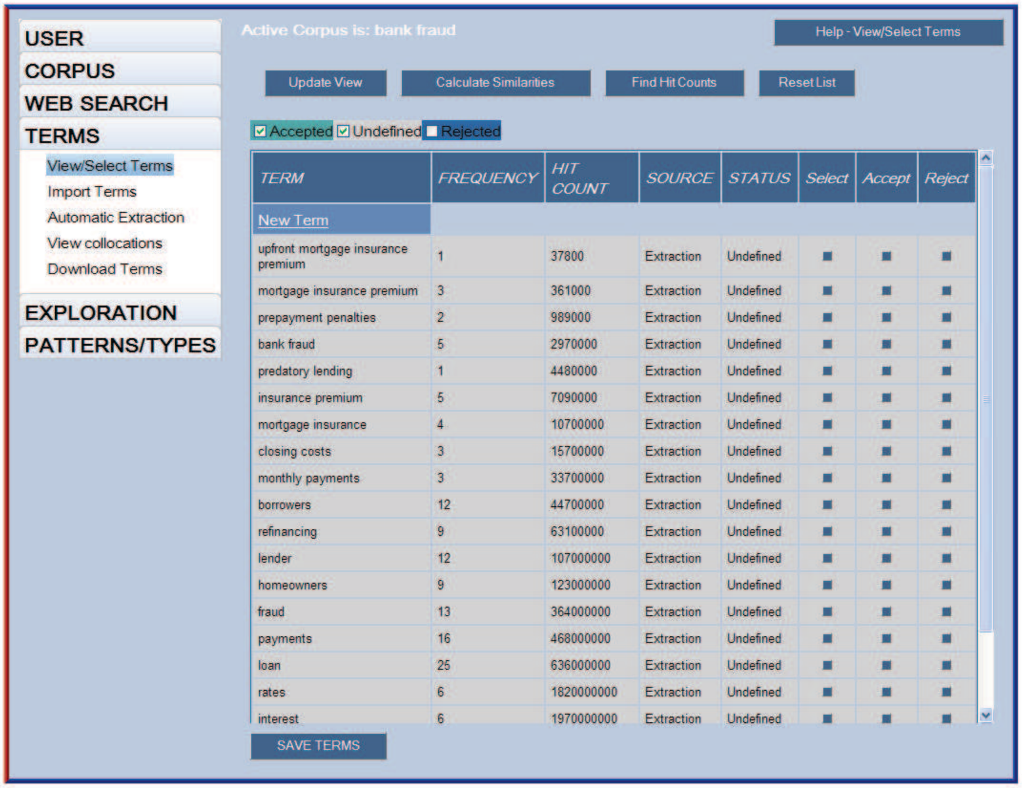
a. Unknown terms
b. Domain terms
c. Number of terms per query
d. Number of queries launched
e. Term frequencies

Unknown terms (factor a.), are the ones the user is interested in understanding. In the bank fraud example, they are "closing costs" or "insurance premium" or "predatory lending" (words shown in Figure 1). When the unknown terms are not polysemous (which is more often the case for multi-word terms), domain terms are not necessary to disambiguate them.

But sometimes, unknown terms are common single-word terms taking on a specific meaning in a particular domain, and then domain terms (factor b.) are important for query disambiguation. For example the term "interest" in our present bank fraud domain has a different meaning then in the expression "having an interest in" from the general language. In such case, domain terms "bank" and "fraud" can be specified to be added to all queries.

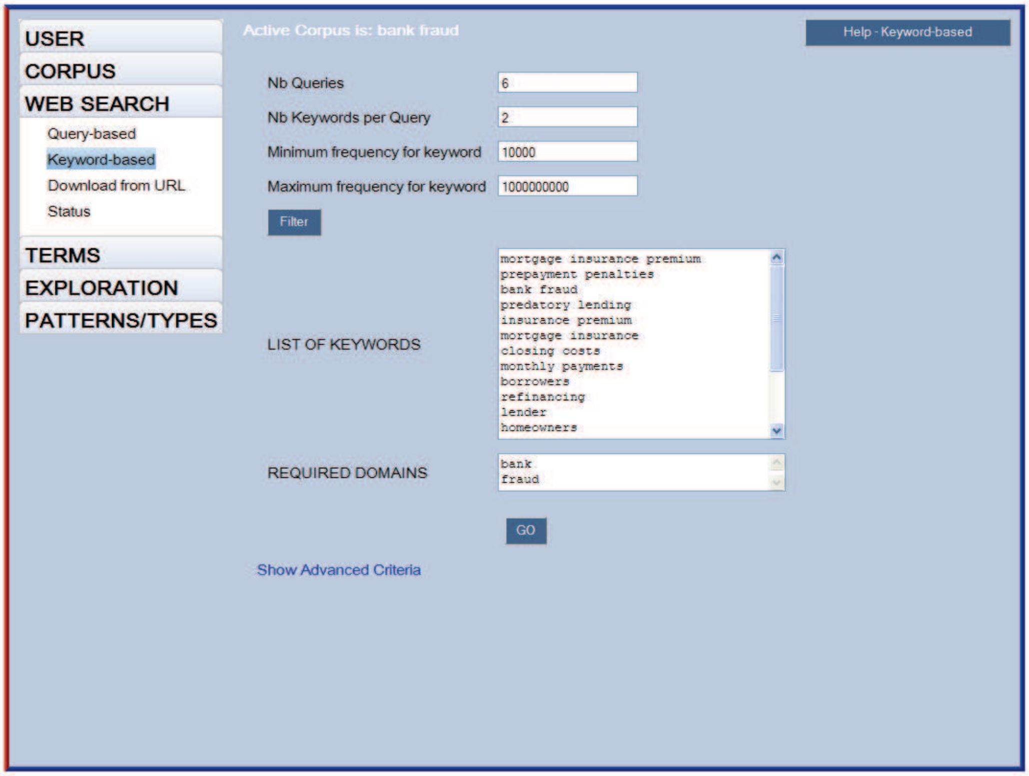
The following two factors (c. and d.) are number of words per query and number of queries. If for example, 10 terms are unknown, the QG module can generate 10 queries of 1 term each, 4 queries of 3 terms each, or 15 queries of 2 terms each, as the user decides. The QG module will make random combinations of terms to generate the required queries. The number of queries would in theory be better if higher, but this becomes a trade-off between the information gained by more corpus data and a longer waiting period. It will be important in our future work to better measure the gain from more queries versus better chosen or targeted queries.

[3] TerminoWeb 2.0 is available online since June 2009 at http://terminoweb.iit.nrc.ca.



TERM	FREQUENCY	HIT COUNT	SOURCE	STATUS	Select	Accept	Reject
upfront mortgage insurance premium	1	37800	Extraction	Undefined	■	■	■
mortgage insurance premium	3	361000	Extraction	Undefined	■	■	■
prepayment penalties	2	989000	Extraction	Undefined	■	■	■
bank fraud	5	2970000	Extraction	Undefined	■	■	■
predatory lending	1	4480000	Extraction	Undefined	■	■	■
insurance premium	5	7090000	Extraction	Undefined	■	■	■
mortgage insurance	4	10700000	Extraction	Undefined	■	■	■
closing costs	3	15700000	Extraction	Undefined	■	■	■
monthly payments	3	33700000	Extraction	Undefined	■	■	■
borrowers	12	44700000	Extraction	Undefined	■	■	■
refinancing	9	63100000	Extraction	Undefined	■	■	■
lender	12	107000000	Extraction	Undefined	■	■	■
homeowners	9	123000000	Extraction	Undefined	■	■	■
fraud	13	364000000	Extraction	Undefined	■	■	■
payments	16	468000000	Extraction	Undefined	■	■	■
loan	25	636000000	Extraction	Undefined	■	■	■
rates	6	1820000000	Extraction	Undefined	■	■	■
interest	6	1970000000	Extraction	Undefined	■	■	■

Figure 1 – Extracted Terms with source text frequencies and Web hit counts



Nb Queries: 6

Nb Keywords per Query: 2

Minimum frequency for keyword: 10000

Maximum frequency for keyword: 1000000000

LIST OF KEYWORDS:

```
mortgage insurance premium
prepayment penalties
bank fraud
predatory lending
insurance premium
mortgage insurance
closing costs
monthly payments
borrowers
refinancing
lender
homeowners
```

REQUIRED DOMAINS:

```
bank
fraud
```

Figure 2 – Query Generator Module Interface

Figure 2 shows the QG Module interface which gives the user much flexibility in specifying domain terms, unknown terms, number of queries and number of terms per query.

When multiple very specific terms are combined, the resulting set of documents is likely to be empty (no documents found). When few general terms are used (one at the limit) the resulting set is likely to be extremely large and inappropriate (imagine a query with "credit" or "stolen"). Empirically, we have found that queries of more than 3 terms often lead to empty sets, although the size of the result set is not solely dependent on the number of terms in the query but rather very closely related to the actual frequency of those terms.

A quick estimate of how specific or general a word or expression is can be provided by a "hit count" measure using a search engine. In our experiment, we use Yahoo Search Engine[4]. Figure 1 shows the term list sorted on hit counts. The sample of terms shown is to provide the reader a sense of the large range from specificity to generality. The term "mortgage insurance premium" is more specific (hit counts: 36100) than "monthly payments" (hit counts: 33700000) which is more specific than "rates" (hit counts: 1820000000).

The QG interface, shown in Figure 2, allows the user to filter query terms based on lower-bound (too specific) and upper-bound (too general) hit counts (factor e.).

Figure 3 shows the queries status. It shows combinations of 2 unknown terms combined with two mandatory domain terms. In TerminoWeb, queries can be "in progress" still looking for documents, "finished" as they have retrieved the requested number of documents (here 10) or "aborted" if something went wrong during the search.

Step 4. Corpus construction

The resulting documents from all the queries are put together to form a large corpus. The maximum number of documents would be equal to the Number of Queries * Number of documents per query, but that is an upper bound since queries could return a smaller set than what is desired (if too specific), some queries could "abort" and also, there will be document overlaps in the returned sets[5].

When a query leads to many documents, then a larger set is analyzed and scored to only keep the 10 most *informative* ones as part of the corpus. Although not the purpose of the present article, we briefly mention that TerminoWeb's focuses on the discovery of informative texts on the Web. Much research efforts have been devoted to TerminoWeb's capability to attribute an "informative score" to each text based on a few criteria such as domain specificity, definitional indicators, text size, sentence length, etc. Much effort has been spent on the exploration of definitional indicators, in the form of knowledge patterns representing different semantic relations. For example, "is a kind of" indicates hyperonymy and "is also known as" indicates synonymy. The presence of such knowledge patterns in a document will increase its informative score. TerminoWeb can show the documents in order of their informative score.

The corpus management module allows the user to inspect each document by providing a link to the original web page. The user can then decide to accept or reject some pages, limiting the documents in the corpus. This step is optional in the present process and mostly useful for thematic searches in which terminologists would like to inspect each source text from which they will select terms and contexts. If this step is not performed, the user will simply perform the next step (explore documents) on a larger set of documents.

Step 5. Collocations and concordances search

The user can now select a term to study and see (1) concordances for this term, (2) collocations generated from the term and (3) concordances for the collocations.

Figure 4 shows concordances for the term "refinancing", illustrating TerminoWeb's capability at providing KWIC views, larger context views, and links to source Web pages.

Figure 5 shows collocations with the word "refinancing". Only two collocations would have been found in the original source text, and many more domain-specific collocations are found in the extended corpus. Calculation of collocations is performed the same way as terms were found. Smadja's algorithm [25] allows the search for non-contiguous collocations. We indicate them with a % for a missing word. The maximum number of missing words was set to one, but could be larger if needed.

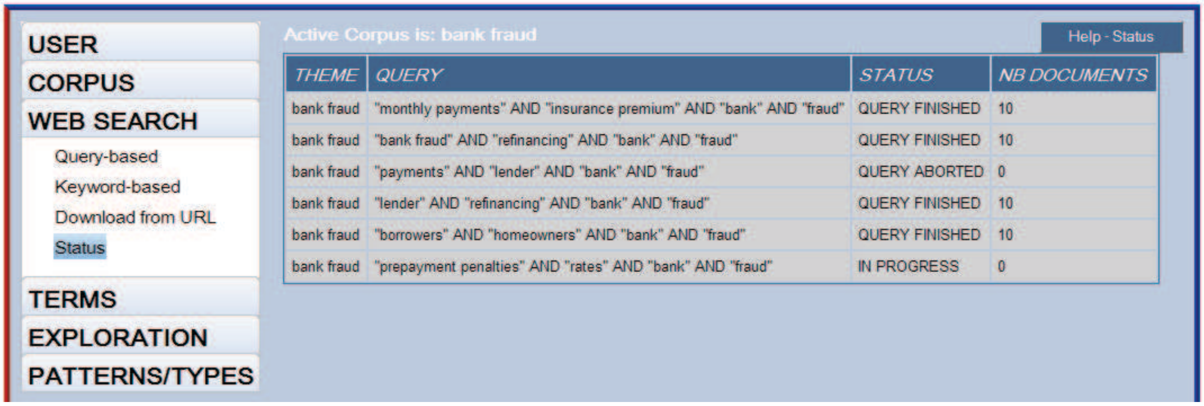
Figure 6 shows the concordancer used to highlight concordances around the found collocations. These are ordered alphabetically[6].

Another interesting feature of TerminoWeb is to allow users to find hit counts for collocations to approximate their specificity/generality, the same way as we presented earlier for terms. Figure 5 shows the hit counts for the different collocations.

[4] Yahoo! provides a java API which can be used for research purposes.

[5] As a research prototype, TerminoWeb can only process html and text documents, and it also filters out "almost-empty documents" containing only links or a few lines.

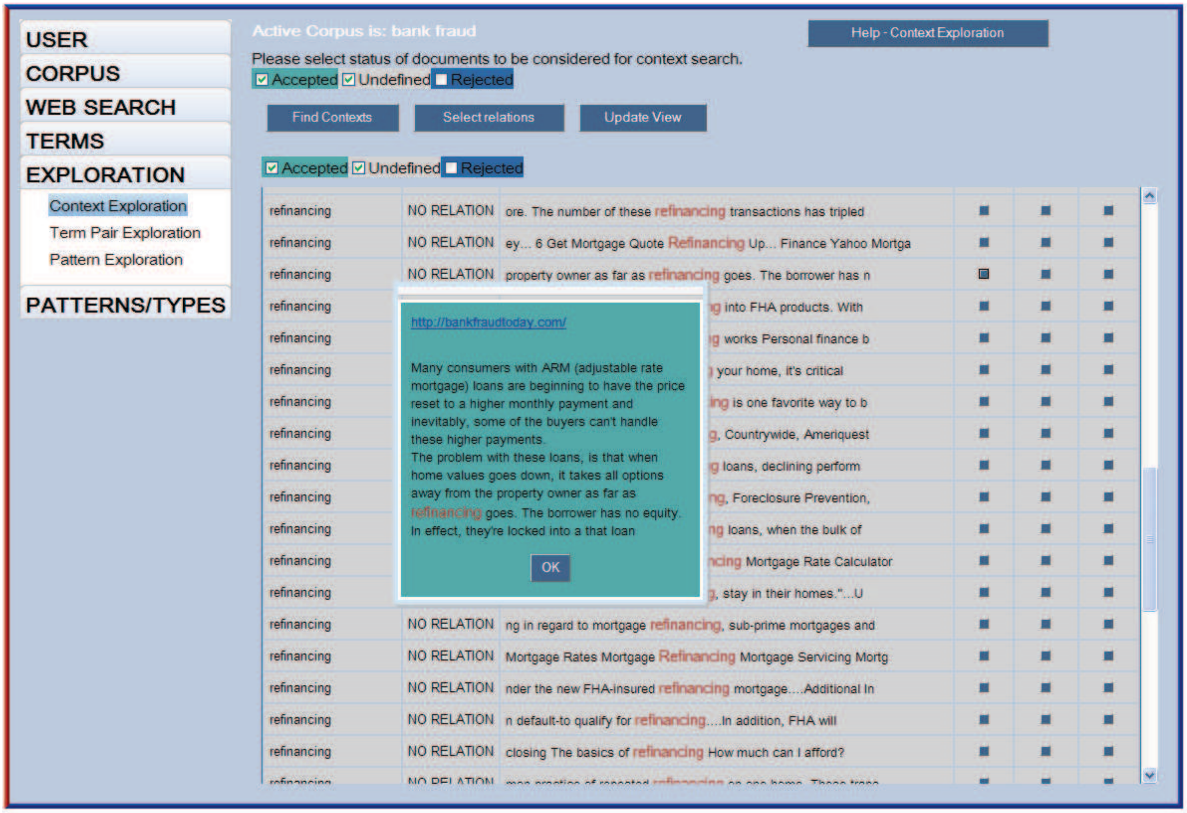
[6] Figures 4 and 6 contain a "no relation" column, meaning the contexts shown do not contain predefined knowledge patterns for different semantic relations.



Active Corpus is: bank fraud

THEME	QUERY	STATUS	NB DOCUMENTS
bank fraud	"monthly payments" AND "insurance premium" AND "bank" AND "fraud"	QUERY FINISHED	10
bank fraud	"bank fraud" AND "refinancing" AND "bank" AND "fraud"	QUERY FINISHED	10
bank fraud	"payments" AND "lender" AND "bank" AND "fraud"	QUERY ABORTED	0
bank fraud	"lender" AND "refinancing" AND "bank" AND "fraud"	QUERY FINISHED	10
bank fraud	"borrowers" AND "homeowners" AND "bank" AND "fraud"	QUERY FINISHED	10
bank fraud	"prepayment penalties" AND "rates" AND "bank" AND "fraud"	IN PROGRESS	0

Figure 3. Status of queries launched



Active Corpus is: bank fraud

Please select status of documents to be considered for context search.

☑ Accepted ☑ Undefined ☐ Rejected

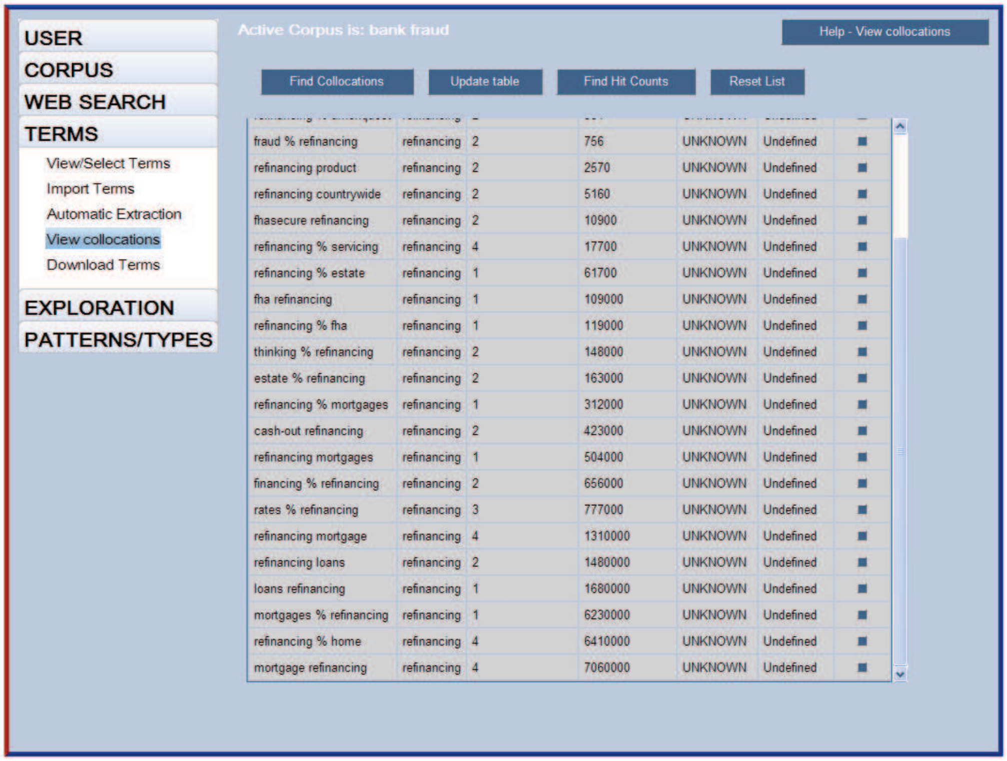
☑ Accepted ☑ Undefined ☐ Rejected

refinancing	NO RELATION	ore. The number of these refinancing transactions has tripled	■	■	■
refinancing	NO RELATION	ey... 6 Get Mortgage Quote Refinancing Up... Finance Yahoo Mortga	■	■	■
refinancing	NO RELATION	property owner as far as refinancing goes. The borrower has n	▣	■	■
refinancing		g into FHA products. With	■	■	■
refinancing		g works Personal finance b	■	■	■
refinancing		) your home, it's critical	■	■	■
refinancing		ing is one favorite way to b	■	■	■
refinancing		g, Countrywide, Ameriquest	■	■	■
refinancing		g loans, declining perform	■	■	■
refinancing		ng, Foreclosure Prevention,	■	■	■
refinancing		ng loans, when the bulk of	■	■	■
refinancing		ncing Mortgage Rate Calculator	■	■	■
refinancing		g, stay in their homes."...U	■	■	■
refinancing	NO RELATION	ng in regard to mortgage refinancing, sub-prime mortgages and	■	■	■
refinancing	NO RELATION	Mortgage Rates Mortgage Refinancing Mortgage Servicing Mortg	■	■	■
refinancing	NO RELATION	nder the new FHA-insured refinancing mortgage....Additional In	■	■	■
refinancing	NO RELATION	n default-to qualify for refinancing....In addition, FHA will	■	■	■
refinancing	NO RELATION	closing The basics of refinancing How much can I afford?	■	■	■
refinancing	NO RELATION	mon practice of repeated refinancing on one home. These trans	■	■	■

http://bankfraudtoday.com/

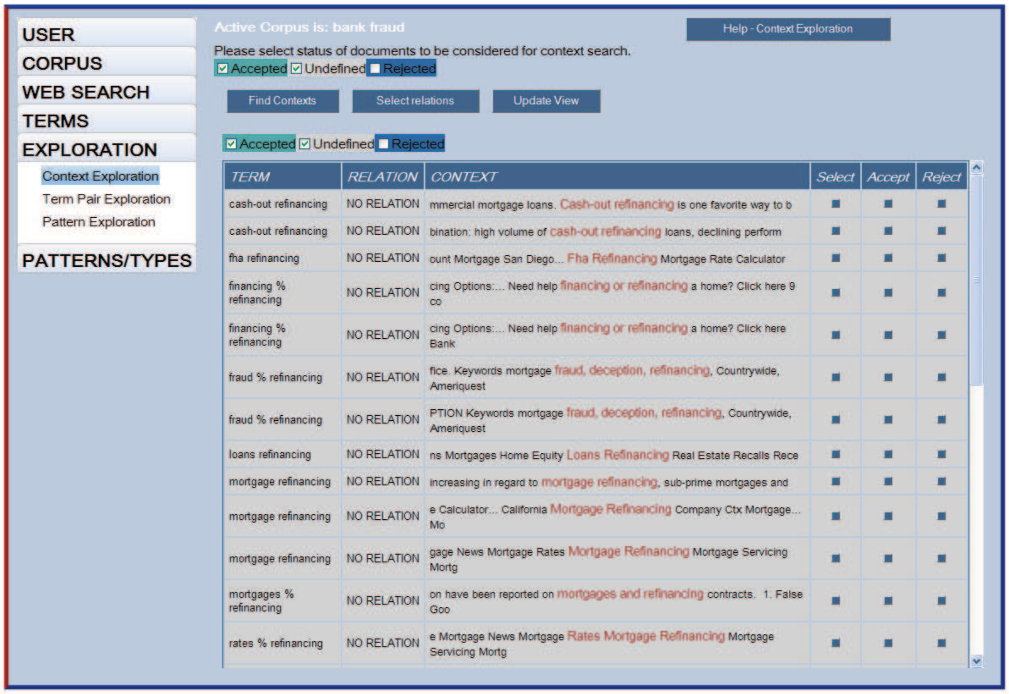
Many consumers with ARM (adjustable rate mortgage) loans are beginning to have the price reset to a higher monthly payment and inevitably, some of the buyers can't handle these higher payments.
The problem with these loans, is that when home values goes down, it takes all options away from the property owner as far as refinancing goes. The borrower has no equity. In effect, they're locked into a that loan

Figure 4. Term “refinancing” concordances



fraud % refinancing	refinancing	2	756	UNKNOWN	Undefined	■
refinancing product	refinancing	2	2570	UNKNOWN	Undefined	■
refinancing countrywide	refinancing	2	5160	UNKNOWN	Undefined	■
fhasecure refinancing	refinancing	2	10900	UNKNOWN	Undefined	■
refinancing % servicing	refinancing	4	17700	UNKNOWN	Undefined	■
refinancing % estate	refinancing	1	61700	UNKNOWN	Undefined	■
fha refinancing	refinancing	1	109000	UNKNOWN	Undefined	■
refinancing % fha	refinancing	1	119000	UNKNOWN	Undefined	■
thinking % refinancing	refinancing	2	148000	UNKNOWN	Undefined	■
estate % refinancing	refinancing	2	163000	UNKNOWN	Undefined	■
refinancing % mortgages	refinancing	1	312000	UNKNOWN	Undefined	■
cash-out refinancing	refinancing	2	423000	UNKNOWN	Undefined	■
refinancing mortgages	refinancing	1	504000	UNKNOWN	Undefined	■
financing % refinancing	refinancing	2	656000	UNKNOWN	Undefined	■
rates % refinancing	refinancing	3	777000	UNKNOWN	Undefined	■
refinancing mortgage	refinancing	4	1310000	UNKNOWN	Undefined	■
refinancing loans	refinancing	2	1480000	UNKNOWN	Undefined	■
loans refinancing	refinancing	1	1680000	UNKNOWN	Undefined	■
mortgages % refinancing	refinancing	1	6230000	UNKNOWN	Undefined	■
refinancing % home	refinancing	4	6410000	UNKNOWN	Undefined	■
mortgage refinancing	refinancing	4	7060000	UNKNOWN	Undefined	■

Figure 5 – Collocations found with “refinancing” in the domain specific corpus.



Please select status of documents to be considered for context search.
☑ Accepted ☑ Undefined □ Rejected

☑ Accepted ☑ Undefined □ Rejected

TERM	RELATION	CONTEXT	Select	Accept	Reject
cash-out refinancing	NO RELATION	mmercial mortgage loans. Cash-out refinancing is one favorite way to b	■	■	■
cash-out refinancing	NO RELATION	bination: high volume of cash-out refinancing loans, declining perform	■	■	■
fha refinancing	NO RELATION	ount Mortgage San Diego... Fha Refinancing Mortgage Rate Calculator	■	■	■
financing % refinancing	NO RELATION	cing Options:... Need help financing or refinancing a home? Click here 9 co	■	■	■
financing % refinancing	NO RELATION	cing Options:... Need help financing or refinancing a home? Click here Bank	■	■	■
fraud % refinancing	NO RELATION	fice. Keywords mortgage fraud, deception, refinancing, Countrywide, Ameriquest	■	■	■
fraud % refinancing	NO RELATION	PTION Keywords mortgage fraud, deception, refinancing, Countrywide, Ameriquest	■	■	■
loans refinancing	NO RELATION	ns Mortgages Home Equity Loans Refinancing Real Estate Recalls Rece	■	■	■
mortgage refinancing	NO RELATION	increasing in regard to mortgage refinancing, sub-prime mortgages and	■	■	■
mortgage refinancing	NO RELATION	e Calculator... California Mortgage Refinancing Company Ctx Mortgage... Mo	■	■	■
mortgage refinancing	NO RELATION	gage News Mortgage Rates Mortgage Refinancing Mortgage Servicing Mortg	■	■	■
mortgages % refinancing	NO RELATION	on have been reported on mortgages and refinancing contracts. 1. False Goo	■	■	■
rates % refinancing	NO RELATION	e Mortgage News Mortgage Rates Mortgage Refinancing Mortgage Servicing Mortg	■	■	■

Figure 6 – Concordances around collocations for “refinancing”

.

3. Related Work

Our research covers a wide range of topics, uses diverse natural language processing strategies, and includes the development of multiple algorithms for all steps, from term extraction to query generation to collocation search. As our purpose in this article is to present a proof of concept of an integrated system, we do not present any quantitative comparisons with other algorithms or systems, but rather highlight some research related to corpus building and analysis.

Our research relies mainly on the principle of "Web as corpus"[7] [17] and exploiting the Web for language learners and translators. In the book Corpus Linguistics and the Web [16], a distinction is made between "accessing the web as corpus" and "compiling corpora from the internet". Our system relates to both views. The hit count specificity/generality approximations relate to the former view. The corpus building modules gathering results from the query generation module relates to the latter view.

Search for Web documents is usually associated to the field of information retrieval. A large body of research exists within that area and we borrow from it. Searching for a particular document to answer a particular question (an information retrieval task) is different than searching for domain-specific documents to "augment" a user's knowledge. The former has a specific goal, finding an answer to a question, and the latter has a discovery purpose.

Nevertheless our query generation module faces the same problems as those of query expansion in information retrieval [12, 27]. Query expansion is a delicate task, as using general terms which tend to be polysemous can lead to off-topic documents, and using very specific terms will not help as they will not return any documents. Our approach was to allow the inclusion of domain-words for restriction and then do a random selection of terms for expansion.

Our query generation module was inspired by the work of Baroni [3, 4] who suggested query combinations of common words to build a corpus of general knowledge or specialized language. Earlier work by Ghani et al. [11] presented a similar idea for minority languages. TerminoWeb includes a unique re-ranking of documents based on an "informative score" as defined in [1]. It then builds informative sub-corpora from the Web.

Although, systems such as WebCorp [24] and KWiCFinder [13] do not build sub-corpora, they use the Web as a large corpus to find collocations and concordances, providing user with easy-to-use real-time systems.

For corpus analysis per se, TerminoWeb combines different modules performing term extraction, collocation extraction and concordance findings. A large pool of research exists in computational terminology around the problem of term extraction. Although a simple frequency based approach is implemented in TerminoWeb, there are more sophisticated algorithms being developed in the community (see [8] for a review of earlier systems and [9] for a new trend of term extraction based on comparing corpora). For collocations, we refer the reader to Smadja [25] for the algorithm we implemented, and to [10] for a review of different measures. Finding concordances does not require any statistical corpus linguistic knowledge, and is simply a window of text capture.

The Sketch Engine [18] system provides a good comparison point to position TerminoWeb. Overall, TerminoWeb's corpus analysis capabilities are simpler than the ones in Sketch Engine. The purpose is quite different, as TerminoWeb's main objective is to provide an integrated platform for understanding terms related to a domain or a source text. For doing so, the emphasis is on easy real-time construction and simple analysis of disposable corpora. No text-preprocessing is necessary, but then, no part-of-speech analysis is available either. We want the user to be able to quickly search for specialized information on the Web to understand important concepts via an integrated system for term extraction and term collocation and concordances finding. This is different from studying language patterns and understanding the uses of words or phrases as can be done in a better way in Sketch Engine [18].

4. Conclusions

Overall, although the value of "disposable corpora" for translators [7, 26] and for language learners [2] is acknowledged, the difficulty of performing text selection based on some principles implemented by natural language processing algorithms, and then the difficulty of doing efficient corpus management certainly prevents most users from building their own corpus. They are in need of tools, such as TerminoWeb, which provide corpus building and analysis capabilities.

TerminoWeb's contribution is actually more at the level of the workflow that the combination of its modules allows than at the level of the strength or novelty of any particular module (except for the "informative" scoring). Such combination makes multiple corpus gathering and analysis task possible.

TerminoWeb is a bit more complex than systems such as WebCorp [24] or KWiCFinder [13] as it

[7] The notion of Web as Corpus is a current research perspective as shown by the Web as Corpus workshops often run in parallel of larger conferences (Corpus Linguistics, 2005, European Association for Computational Linguistics EACL-2006, LREC 2008).

provides an integration of multiple modules, and therefore requires a longer learning curve, but the integration also makes it quite powerful, allowing a workflow such as described in this article, to start from a source text and find valuable information from the automatically extracted terms of that source text.

Our main future work is to gather feedback from users as they experiment with the prototype. This will allow us to better understand the particular needs of different users (language learners versus translators). This will help refine our modules and refine our choice of appropriate natural language processing techniques in support of each module.

5. References

[1] Agbago, A. Barrière, C. Corpus Construction for Terminology, Corpus Linguistics Conference, Birmingham, UK, July 14-17, 2005.

[2] Aston, G. Learning with Corpora, Houston: Athelstan, 2003.

[3] Baroni, M. and Bernardini, S. BootCaT: Bootstrapping Corpora and Terms from the Web, Proceedings of LREC, 2004.

[4] Baroni, M., Kilgarriff, A., Pomikalek, J. and Pavel, R. WebBootCaT: instant domain-specific corpora to support human translators, Proceedings of the 11th Annual Conference of the European Association for Machine Translation, EAMT-2006, Norway, 2006.

[5] Barrière C. and Agbago, A. TerminoWeb: a software environment for term study in rich contexts, International Conference on Terminology, Standardization and Technology Transfer, Beijing, 103-113, 2006.

[6] Bowker, Lynne and Pearson, Jennifer. Working with Specialized Text: A Practical Guide to Using Corpora. Routledge, 2002.

[7] Bowker, Lynne. "Working Together: A Collaborative Approach to DIY Corpora". First International Workshop on Language Resources for Translation Work and Research, Gran Canaria, 2002.

[8] Cabré Castellvi, M.T., Estopa R., Palatresi, J.V. Automatic term detection: A review of current systems. In Bourigault D., Jacquemin C., L'Homme M.C. (eds) Recent advances in Computational Terminology, vol. 2, pp. 53-87, 2001.

[9] Drouin P. Term extraction using non-technical corpora as a point of leverage. Terminology 9(1), pp. 99-117, 2003.

[10] Evert, Stefan and Krenn, Brigitte (2001). Methods for the qualitative evaluation of lexical association measures. In Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics, Toulouse, France, 188-195, 2001.

[11] Ghani, R., Jones, R., Mladenic, D., Mining the web to create minority language corpora, CIKM 2001, 279-286, 2001.

[12] Greenberg, J. Automatic query expansion via lexical – semantic relationships. Journal of the American Society for Information Science and Technology, 52(5), 402 – 415, 2001.

[13] Fletcher, W.H.. Concordancing the web: promise and problems, tools and techniques, in Hundt, M. Nesselhauf, N. and Biewer, C. (Eds) Corpus Linguistics and the Web, 2007.

[14] Hoffmann, S., Evert, S., Smith, N., Lee, D. and Ylva B.P. Corpus Linguistics with BNCweb - a Practical Guide. Frankfurt am Main: Peter Lang, 2008.

[15] Hulstijn, J. H., Hollander, M. & Greidanus, T. Incidental vocabulary learning by advanced foreign language students: The influence of marginal glosses, dictionary use, and reoccurrence of unknown words. *The Modern Language Journal, 80,* 327-339, 1996.

[16] Hundt, M. Nesselhauf, N., Biewer, C. Corpus Linguistics and the Web, Amsterdam/New York, NY, 2007, VI, 2007.

[17] Kilgariff, Adam and Gregory Grefenstette, Special Issue on Web as Corpus, Computational Linguistics, 29 (3), 2003.

[18] Kilgariff, A., Rychly, P., Smrz, P., Tugwell, D. The Sketch Engine, EURALEX, Lorient, France, 2004.

[19] Laviosa, Sara. Corpus-based Translation Studies: Theory, Findings, Applications, Amsterdam: Rodopi. 2002.

[20] Maia, B., Matos, S. Corpografo V.4 – Tools for Researchers and Teachers using Comparable Corpora, LREC 2008, Marrakech, Morocco, 2008.

[21] Nagy, W. E., Herman, P., and Anderson, R. C. Learning words from context. Reading Research Quarterly, 20, 233 – 253, 1985.

[22] Olohan, Maeve. Introducing Corpora in Translation Studies. London and New York: Routledge. 2004.

[23] Pearson, P. D. and Studt, A. Effects of word frequency and contextual richness on children's word identification abilities. Journal of Educational Psychology, 67(1), 89 – 95, 1975.

[24] Renouf, A., Kehoe, A., Banerjee, J. WebCorp: an integrated system for web text search, in Hundt, M. Nesselhauf, N. and Biewer, C. (Eds) Corpus Linguistics and the Web, 2007.

[25] Smadja F. Retrieving collocations from text: Xtract, Computational Linguistics, 19(1), 134-177, 1993.

[26] Varantola, K. Disposable corpora as intelligent tools in translation, Cadernos de Traduçao IX – Traduçao e Corpora, Vol. 1, No 9 (2002),171-189, 2002.

[27] Vechtomova, O., Robertson, S., & Jones, S. Query expansion with long-span collocates. Information Retrieval, 6, 251 – 273, 2003.

[28] Zanettin, Federico, Bernardini Silvia and Stewart Dominic. Corpora in Translation Education, Manchester: St Jerome, 2003.

Unsupervised Construction of a Multilingual WordNet from Parallel Corpora

Dimitar Kazakov and Ahmad R. Shahid
Department of Computer Science
University of York
Heslington, York YO10 5DD, UK
kazakov|ahmad@cs.york.ac.uk

Abstract

This paper outlines an approach to the unsupervised construction from unannotated parallel corpora of a lexical semantic resource akin to WordNet. The paper also describes how this resource can be used to add lexical semantic tags to the text corpus at hand. Finally, we discuss the possibility to add some of the predicates typical for WordNet to its automatically constructed multilingual version, and the ways in which the success of this approach can be measured.

Keywords

Parallel corpora, WordNet, unsupervised learning

1 Introduction

Lexical ambiguity is inherent and widespread in all languages; it emerges spontaneously in computer simulations of language evolution [23], and its origin probably stems in the interplay between geographic divisions and interaction between communities, diachronic linguistic changes, and evolutionary pressures on the cost of communication. Two challenges arise when dealing with lexical ambiguity: firstly, to define the elementary semantic concepts employed in a given language, and, secondly, given one or more utterances, to identify the semantic concepts to which the words in those utterances refer. Throughout history, numerous attempts have been made to address both challenges through the construction of artificial languages with unambiguous semantics (*e.g.*, see Ecos detailed and entertaining review [3]). Arguably, there are two possible ways of defining a semantic concept, either by trying to map it onto some sensory experience (leading to a philosophical discussion about the extent to which they are shared and the notion of *qualia*), or by describing it through other concepts, in a way that is mutually recursive and unbounded (cf. Peirces Sign Theory and the notion of *infinite semiosis*).

The last twenty five years saw academic and commercial efforts directed towards the creation of large repositories combining the description of semantic concepts, their relationship, and, possibly, common knowledge statements expressed in those terms. This includes, among others, the Cycorp Cyc project [11] and the lexical semantic database WordNet [14]. Both approaches use a number of predicates to make statements, such as “concept A is an instance of concept B” or “concept A is a specific case of concept B” (in other terms, all instances of concept A form a subset of the instances of concept B). WordNet also employs the notion of *synsets*, defining a semantic concept through the list of words (synonyms) sharing that meaning, *e.g.*, {mercury, quicksilver, Hg}. The original version of WordNet developed in Princeton covered the English language, but this effort has been replicated for other languages [25]. All these databases are monolingual; they are also handcrafted, and while very comprehensive in many aspects, it is difficult to assume they could be kept abreast of the new developments, including the use of newly coined words, and giving new meanings to the old ones.

The dawn and rapid growth of dynamic online encyclopedic resources with shared authorship, such as Wikipedia, in the last decade, have begun to draw attention as a potential source of semantic concepts and ontologies [7]. Recently, there have also been efforts to use the likes of Wikipedia to the existing hand-crafted resources [13].

2 Multilingual Synsets

The synsets in WordNet clarify a concept (or, from another point of view, narrow down the sense of a word) by paraphrasing it repeatedly, using other words of the same language. This approach is based on the fact that words rarely share all their meanings: {step, pace} specifies a different meaning from {step, stair}. The same result, however, can be achieved through the use of words of different languages that share at least one sense and therefore can be seen as translations of each other. So, {EN: bank, FR: banque} could refer to a financial institution or a collection of a particular kind (*e.g.*, a blood bank), as these words share both meanings, but eliminates the concept of ‘river bank’ that the English word alone might denote. Increasing the number of languages could gradually remove all ambiguity, as in the case of {EN: bank, FR: banque, NL: bank}. Insofar these lists of words specify a single semantic concept, they can be seen as synsets of a WordNet that makes use of words of several languages, rather than just one. The greater the number of translations in this multilingual WordNet, the clearer the meaning, yet, one might object, the fewer the number of such polyglots, who could benefit from such translations. However, these multilingual synsets can also be useful in a monolingual context, as unique indices that distinguish the individual meanings of a word. For instance, if the English word bank is translated variously as {EN: bank, FR: banque}, and {EN: bank, FR: rive} one does not need to understand French to suspect that 'bank' may have (at least) two different meanings. The greater the number of languages in which the two synsets differ, the stronger the intuition that they indicate different meanings of the word in the pivotal language.

Synsets, whether monolingual or multilingual, can be used to tag the lexical items in a corpus with their intended meaning (see Table 1). The benefits of such lexical semantic tags are evident. Focussing now on the novel case of multilingual synsets, one can consider the two separate questions of how to produce a collection of such synsets, and how to annotate the lexical items in a text with them. Kazakov and Shahid [22] present an interesting study in that direction, where the titles of 'equivalent' Wikipedia pages are collected for a number of preselected languages on the assumption that they represent an accurate translation of each other (see Fig.1).

The authors repeat the same exercise for a number of specific domains, and also with the names of Wikipedia categories. The latter case demonstrates the potential to use Wikipedia not only to collect multilingual synsets, but also to add a hierarchical relationship between them, as this information is explicitly present there. A number of other researchers have in fact used Wikipedia to extract ontologies [6], [7], but in all cases this was done for a single language.

Semantically disambiguated corpora, including those using WordNet synsets as semantic tags, are valuable assets [10], [9], but creating them requires a considerable effort. Here we propose an alternative approach, where a text is automatically annotated with lexical semantic tags in the form of multilingual synsets, provided the text forms part of a multilingual, parallel corpus.

Table 1: *Examples of lexical semantic annotation using standard English WordNet synsets (above) and multilingual synsets (below).*

A darkened outlook for Germany's banks:**SS1** The banks:**SS2** of the river Nile **SS1**={bank, depository financial institution} Gloss="a financial institution that accepts deposits and lends money" **SS2**={bank} Gloss="sloping land"

A darkened outlook for Germany's banks:**mSS1**
The banks:**mSS2** of the river Nile
 	
mSS1={EN:bank, FR:banque, CZ:banka}
mSS2={EN:bank, FR:rive, CZ:břeh}

Table 2: *Examples of variation between synsets due to the use of different word forms (above) and synonyms (below).*

EN	**FR**	**CZ**	...
work	*travail*	práce	...
works	*travaux*	práce	...
work	*travail*	práce	...
work	*bouleau*	práce	...

3 Annotating Parallel Corpora with Lexical Semantics

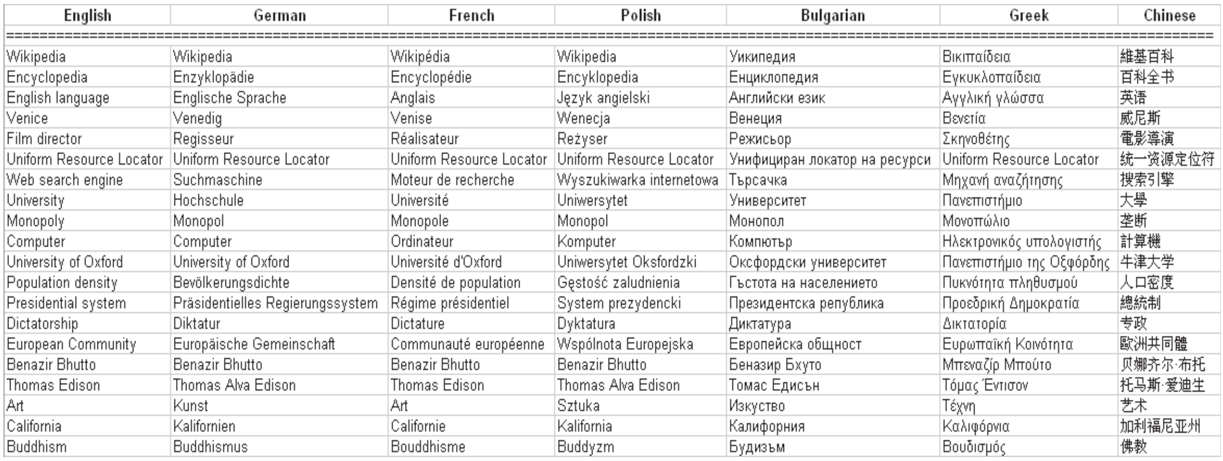
In our approach the multilingual synsets become the sense tags and the parallel corpora are tagged with the corresponding tags (see Fig.2).

The idea is as simple as it is elegant: assuming we have a word-aligned parallel corpus with n languages, annotate each word with a lexical semantic tag consisting of the n-tuple of aligned words. As a result, all occurrences of a given word in the text for language $\mathcal{L}$ are considered as having the same sense, provided they correspond to (are tagged with) the same multilingual synset.

Two great advantages of this scheme are that it is completely unsupervised, and the fact that, unlike manually tagged corpora using WordNet, all words in the corpus are guaranteed to have a corresponding multilingual synset. Since we are only interested in words with their own semantics, a stop list can be used to remove the words of the closed lexicon before the rest are semantically tagged. Also the need for word alignment should not be an issue, at least in principle, as there are standard tools, such as GIZA++ [16] serving that purpose.

The approach as described so far needs to deal with two main issues, both related to the fact that the simple listing of n-tuples as synsets may produce many more synsets than the real number of concepts to which the words in the text refer. The first issue stems from the fact that a lexeme (word entry) corresponds to several word forms in most languages, so, for instance, the word forms {EN: work} and {EN: works} will correspond to two different synsets (Table 2, top), even if they are used with the same meaning. The second of the above mentioned issues is related to the use of synonyms in one language, whereas the translation in another makes use of the same word (lexeme) (Table 2, bottom).

From this point of view, we could consider the original n-tuples as *proto-synsets*, and then strife to recognize the variation due to the use of different word forms and synonyms, and eliminate it, if possible, by grouping these proto-synsets into genuine synsets corresponding to different concepts. As much of the appeal of the whole approach stems from its unsupervised nature, we shall also consider unsupervised ways of solving this specific problem. For several languages, there are detailed, explicit models of their word morphology [19], [20], [17], which makes mapping word forms onto the list of lexemes they may represent a straightforward task.



English	German	French	Polish	Bulgarian	Greek	Chinese
Wikipedia	Wikipedia	Wikipédia	Wikipedia	Уикипедия	Βικιπαίδεια	維基百科
Encyclopedia	Enzyklopädie	Encyclopédie	Encyklopedia	Енциклопедия	Εγκυκλοπαίδεια	百科全书
English language	Englische Sprache	Anglais	Język angielski	Английски език	Αγγλική γλώσσα	英语
Venice	Venedig	Venise	Wenecja	Венеция	Βενετία	威尼斯
Film director	Regisseur	Réalisateur	Reżyser	Режисьор	Σκηνοθέτης	電影導演
Uniform Resource Locator	Uniform Resource Locator	Uniform Resource Locator	Uniform Resource Locator	Унифициран локатор на ресурси	Uniform Resource Locator	统一资源定位符
Web search engine	Suchmaschine	Moteur de recherche	Wyszukiwarka internetowa	Търсачка	Μηχανή αναζήτησης	搜索引擎
University	Hochschule	Université	Uniwersytet	Университет	Πανεπιστήμιο	大學
Monopoly	Monopol	Monopole	Monopol	Монопол	Μονοπώλιο	垄断
Computer	Computer	Ordinateur	Komputer	Компютър	Ηλεκτρονικός υπολογιστής	計算機
University of Oxford	University of Oxford	Université d'Oxford	Uniwersytet Oksfordzki	Оксфордски университет	Πανεπιστήμιο της Οξφόρδης	牛津大学
Population density	Bevölkerungsdichte	Densité de population	Gęstość zaludnienia	Гъстота на населението	Πυκνότητα πληθυσμού	人口密度
Presidential system	Präsidentielles Regierungssystem	Régime présidentiel	System prezydencki	Президентска република	Προεδρική Δημοκρατία	總統制
Dictatorship	Diktatur	Dictature	Dyktatura	Диктатура	Δικτατορία	专政
European Community	Europäische Gemeinschaft	Communauté européenne	Wspólnota Europejska	Европейска общност	Ευρωπαϊκή Κοινότητα	歐洲共同體
Benazir Bhutto	Benazir Bhutto	Benazir Bhutto	Benazir Bhutto	Беназир Бхуто	Μπεναζίρ Μπούτο	贝娜齐尔·布托
Thomas Edison	Thomas Alva Edison	Thomas Edison	Thomas Alva Edison	Томас Едисън	Τόμας Έντισον	托马斯·爱迪生
Art	Kunst	Art	Sztuka	Изкуство	Τέχνη	艺术
California	Kalifornien	Californie	Kalifornia	Калифорния	Καλιφόρνια	加利福尼亚州
Buddhism	Buddhismus	Bouddhisme	Buddyzm	Будизъм	Βουδισμός	佛教

Fig. 1: *Wikipedia page titles seen as multilingual synsets.*

Using morpho-lexical analyzers for the languages in the corpus will produce a list of lexical entries for each language, which can be narrowed down through the use of conventional lexicons listing the possible pairs of lexical entries between given two languages. In this way, the word form ‘works’ will be mapped onto the lexemes *work*, **n.**, *works*, **n.**, and *work*, **v.**, but in the context of the French *travail*, only the first lexeme will be retained, as the other two would not form a translation pair in an English-French lexicon.

One could also consider doing away with any models of morphology, assuming complete ignorance in this respect, and employing unsupervised learning of word morphology [8], [4]. In their latest form, these approaches can identify word form paradigms, which would allow all forms of a lexical entry to be mapped consistently onto it.

It is also possible to automate the process of identifying synonyms among the words of a given language. For instance, Crouch’s approach [2] is based on the discrimination value model [21]. Other approaches include Bayesian Networks [18], Hierarchical Clustering [24], and link co-occurrences [15], etc. Such automated approaches have certain advantages over the manually generated thesauri, both in terms of cost and time of development, and also in the level of detail, with the latter often being too fine grained, and reflecting usages that are not common and irrelevant in practice [12].

4 Extracting a Fully-Fledged Multilingual WordNet

So far, we have described a procedure that extracts multilingual synsets from a parallel corpus and reduces spurious ambiguity by merging synsets corresponding to multiple word forms of the same lexeme, resp. by combining those only varying in the use of synonyms of a given language. Extraction of hierarchical semantic relationships between words in a corpus has been studied for almost two decades [5], and the same procedures can be applied here, leading to the acquisition of a lexical resource akin to WordNet, which also contains some of the predicates (*e.g.*, hyponym/2, resp. hypernym/2). Such semantic hierarchies can then be used to tag the text corpus with synsets of a certain level of granularity, depending on the task at hand.

5 Evaluation

The evaluation of this approach could be twofold: directly, using an already semantically annotated gold standard, and indirectly, through the measured benefits of lexical semantic tagging in other tasks. The limitations of the direct approach are given by the lack of semantically annotated parallel corpora, however, there is at least one such corpus at the time of writing, namely, multi-Semcor [1]. Indirectly, the potential benefits of tagging text with such multilingual synsets can be measured in tasks, such as document clustering, where the lexical semantic tags can be used to discriminate between various word senses. Any improvement in the within-clusters and between-clusters quality measures would indicate relative (and measurable) success.

References

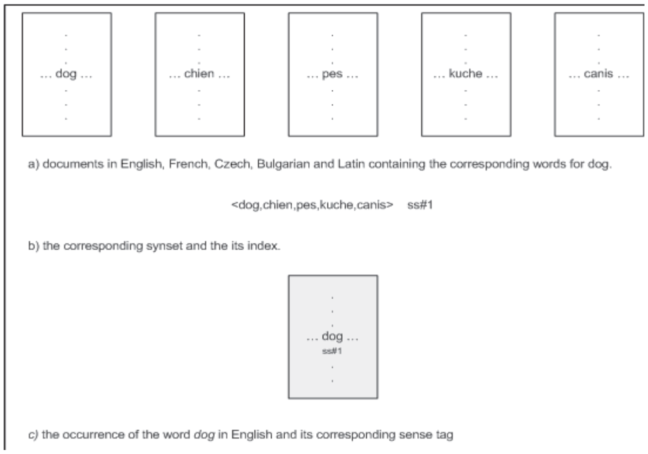
[1] L. Bentivogli, E. Pianta, and M. Ranieri. Multisemcor: an English Italian aligned corpus with a shared inventory of senses. In *Proceedings of the Meaning Workshop*, page 90, Trento, Italy, February 2005.

[2] C. J. Crouch. A Cluster-Based Approach to Thesaurus Construction. In *Proceedings of ACM SIGIR-88*, page 309320, 1988.

[3] U. Eco. *La recherche de la langue parfaite dans la culture européenne.* Seuil, Paris, 1994.

[4] J. Goldsmith. Unsupervised acquisition of the morphology of a natural language. *Computational Linguistics*, 27(2):153–198, 2001.

[5] M. A. Hearst. Automatic acquisition of hyponyms from large text corpora. In *Proceedings of the 14th International Conference on Computational Linguistics (COLING)*, 1992.



a) documents in English, French, Czech, Bulgarian and Latin containing the corresponding words for dog.

b) the corresponding synset and the its index.

c) the occurrence of the word *dog* in English and its corresponding sense tag

Fig. 2: *Assignment of sense tags in aligned documents.*

[6] M. Hepp, D. Bachlechner, and K. Siorpaes. Harvesting wiki consensus – using wikipedia entries as ontology. In *Proc. ESWC-06 Workshop on Semantic Wikis*, pages 132–46, 2006.

[7] A. Herbelot and A. Copestake. Acquiring ontological relationships from wikipedia using RMRS. In *Proc. ISWC-06 Workshop on Web Content Mining with Human Language*, 2006.

[8] D. Kazakov. Unsupervised learning of nave morphology with genetic algorithms. In W. van den Bosch and A. Weijters, editors, *in Workshop Notes of the ECML/MLnet Workshop on Empirical Learning of Natural Language Processing Tasks*, pages 105–112, Prague, Czech Republic, April 1997.

[9] D. Kazakov. Combining LAPIS and WordNet for the learning of LR parsers with optimal semantic constraints. In S. Dzeroski and P. Flach, editors, *The Ninth International Workshop ILP-99*, volume 1634 of *LNAI*, Bled, Slovenia, 1999. Springer-Verlag.

[10] S. Landes, C. Leacock, and R. Tengi. *WordNet: An Electronic Lexical Database*, chapter Building semantic concordances. MIT Press, Cambridge, Mass., 1998.

[11] D. Lenat. Cyc: A large-scale investment in knowledge infrastructure. *Communications of the ACM*, 38(11), 1995.

[12] D. Lin. Automatic Rretrieval and Clustering of Similar Words. In *Proceedings of the 17th international conference on Computational linguistics*, pages 768–774, Montreal, Quebec, Canada, August 1998.

[13] O. Medelyan and C. Legg. Integrating Cyc and Wikipedia: Folksonomy meets rigorously defined common-sense. In *Proceedings of the WikiAI Workshop at AAAI-2008*, Chicago, 2008.

[14] G. A. Miller, R. Beckwith, C. Fellbaum, D. Gross, and K. Miller. Five papers on WordNet. In C. Fellbaum, editor, *WordNet: An Electronic Lexical Database*. MIT Press, May 1998.

[15] K. Nakayama, T. Hara, and S. Nishio. Wikipedia mining for an association web thesaurus construction. In *In Proceedings of IEEE International Conference on Web Information Systems Engineering*, pages 322–334, 2007.

[16] F. J. Och and H. Ney. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51, 2003.

[17] K. Oflazer. Two-level description of turkish morphology. In *EACL*, 1993.

[18] Y. C. Park and K.-S. Choi. Automatic thesaurus construction using Bayesian networks. *Information Processing and Management: an International Journal*, 32(5):543–553, September 1996.

[19] E. Paskaleva. A formal procedure for Bulgarian word form generation. In *COLING*, pages 217–221, 1982.

[20] G. D. Ritchie, G. J. Russell, A. W. Black, and S. G. Pulman. *Computational Morphology: Practical Mechanism for the English Lexicon.* MIT Press, 1991.

[21] G. Salton, C. Yang, and C. Yu. A Theory of Term Importance in Automatic Text Analysis. *Journal of the American Society for Information Science*, 26(1):33–44, 1975.

[22] A. R. Shahid and D. Kazakov. Automatic Multilingual Lexicon Generation using Wikipedia as a Resource. In *Proc. of the Intl. Conf. on Agents and Artificial Intelligence (ICAART)*, Porto, Portugal, January 2009.

[23] L. Steels. Emergent adaptive lexicons. In P. Maes, M. Mataric, J.-A. Meyer, J. Pollack, and S. W. Wilson, editors, *Fourth International Conference on Simulation of Adaptive Behavior.* The MIT Press/Bradford Books, 1996.

[24] T. Tokunaga, M. Iwayama, and H. Tanaka. Automatic thesaurus construction based-on grammatical relations. In *In Proceedings of the 14th International Joint Conference on Artificial Intelligence*, pages 1308–1313, 1995.

[25] P. Vossen, editor. *EuroWordNet.* Kluwer, 1998.

Search techniques in corpora for the training of translators

Verónica Pastor
TecnoLeTTra
Universitat Jaume I
Av. de Vicent Sos Baynat, s/n. 12071
Castellón de la Plana, Spain
vpastor@trad.uji.es

Amparo Alcina
TecnoLeTTra
Universitat Jaume I
Av. de Vicent Sos Baynat, s/n. 12071
Castellón de la Plana, Spain
alcina@trad.uji.es

Abstract

In recent years, translators have increasingly turned to corpora as a resource in their terminology searches. Consequently university translation courses should include training for future translators in the effective use of corpora, thus enabling them to find the terminology they need for their translations more quickly and efficiently.

This paper provides a classification of search techniques in electronic corpora which may serve as a useful guide to the efficient use of electronic corpora both in the training of future translators, and for professional translators.

Keywords
Electronic corpora for translators, search techniques, corpus queries, translation resources, translation training.

1. Introduction

Terminology is a key factor in translators' work. The development of specialized fields has grown hand in hand with advancements in science and technology. These market demands explain why translators are calling for resources to satisfy their terminological needs quickly and effectively [1].

Dictionary creation cannot keep pace with developments in specialized fields. Many studies show dictionaries to be deficient in the lack of information they include, speed of content update, and the limited ways of accessing contents. For this reason, translators are increasingly turning to other resources, such as the Internet and corpora, to search for the terminology they need.

In this paper we analyze the search techniques offered by a range of electronic corpora. Our search technique classification is aimed to provide translation teachers with a reference to help them teach students how to use corpora efficiently. This classification may also be of interest to professional translators who want to further their knowledge of electronic corpora techniques in order to improve their query results.

2. The need for corpora in translation

Market demands require translators to work against tight deadlines and with rapidly evolving vocabulary. According to Varantola [22], fifty per cent of the time spent on a translation is taken up with consulting reference resources. Many studies have revealed that dictionaries do not satisfy all translators' terminological queries [5, 9, 16]. Gallardo and Irazazábal [10] suggest that the terminology translators need, apart from equivalents in different languages, should also include contexts and information about the concept that allow translators to decide how and where to use a term.

In this vein, Zanettin [23] states that the use of corpora in translation training was commonplace even before the development of electronic corpora. Snell-Hornby [21] and Shäffner [18], for instance, argue that by studying similar texts in the source and target languages translators may identify prototypical features that are useful for the target text production.

Since the development of electronic corpora, the need for these tools has become more evident, especially as a terminology resource for translators. Several authors state that translators need new terminological resources, such as corpora [3, 4, 11, 15], which complement dictionary and database use [8, 12, 19] and satisfy specific terminological problems quickly and reliably.

Some studies have demonstrated that translation quality improves when translators use corpora in their terminology searches. Zanettin [23] conducted an experiment with translation students from the School for Translators and Interpreters at the University of Bologna. He shows that comparable corpora[1] help translation students to compare the use of similar discourse units in two languages and facilitate the selection of equivalents adapted to the translation context. Bowker [3] carried out a study with translation students from the School of Applied Language and Intercultural Studies at Dublin City University. She found that corpus-aided translations are of higher quality than translations carried out only with the aid of dictionaries. In a subsequent study, Bowker [4] suggests various ways a target language corpus can be used as a terminological resource for translators.

Despite the usefulness of corpora, the need to use a range of resources to access terminology is a daily problem facing translators. According to Alcina [1], if translators

[1] Zanettin [23] defines a comparable corpus as a collection of independent original texts in two or more languages with similar content, domain and communicative function.

have to undertake terminological tasks, whether searching in a corpus or on the Internet, time is wasted and their translation efficacy is poorer. As Varantola [22] states, the success of a query depends on the intelligent use of search tools.

Translation students should receive quality training at university level in the use of new electronic resources in order to respond to the demands of companies and institutions [2]. Any training in electronic resources should also include electronic corpora search techniques[2]. If this were the case, translators would spend less time and effort acquiring the competences to query corpora efficiently once they have embarked on their professional career. If translators know how to use search techniques in electronic corpora, they will be able to satisfy their terminological needs more quickly and efficiently and the quality of their translations will improve.

3. Corpora examined in the analysis

This study analyzed the search functions of various stable online corpora interfaces. Because we wanted to analyze corpora that are easily accessible to translators, we selected those that are available online. All the corpora analyzed incorporate interfaces that allow different types of queries.

It is worth noting that many of these corpora are not specifically designed for translators. In addition, each corpus explains its own query options, but few studies provide a comprehensive and systematized classification of all the search techniques that can be used in a corpus.

Our classification will provide an overview of all the search techniques that have been incorporated in electronic corpora to date. We will use this classification in future research to discover which of these search techniques are useful for translators, in order to create electronic corpora adapted to translators needs, as well as to teach translators the range of search techniques used in electronic corpora.

In this section we briefly describe the corpora analyzed, focusing on the particular features of each corpus. Specific examples of queries in the corpora are included in our search technique classification.

The **Corpus de referencia del español** (CREA) and the **Corpus diacrónico del español** (CORDE) are two monolingual online corpora developed by the Real Academia Española. CREA contains modern Spanish texts from 1975 to 2004. CORDE includes Spanish texts written up to 1975. Both corpora allow the use of distance criteria between words. Corpus filters such as field, author and work, date, register, and geographic area can be applied. Statistical data on search results, concordances and clusters are also available.

At Brigham Young University (BYU), Professor Davies created an online interface for a set of monolingual corpora: the **Corpus del español** (Spanish from 1200s-1900s), **Corpus of Contemporary American English** (US English from 1990-2008), **BYU-British National Corpus** (British English from 1980-1993), **TIME Corpus** (US English from 1923-present), **BYU-OED Oxford English Dictionary** (Old English-1990) and **Corpus do Português** (Portuguese from 1300s-1900s).

This interface allows the user to search one or more word forms, lemmas or parts of speech. Part of speech restrictions can be applied. Searches can also be limited by genre or over time. It compares the frequency of words, phrases or grammatical constructions by genre or over time. The user can search for collocates of a word or compare collocates of two words. Another particular feature is the semantically-oriented search, which enables the user to search for synonyms[3] of a word. Finally, customized lists of words or phrases may be created for use in a query.

The **British National Corpus** (BNC) is a monolingual corpus of modern spoken and written British English. The online interface allows the user to search for a word or phrase. More complex queries can be carried out using the SARA/XAIRA search tool (www.oucs.ox.ac.uk/rts/xaira) or directly from the online search box using the BNC Corpus Query Language, or the online CQP edition of the BNC[4].

The **Hellenic National Corpus** (HNC) is an online monolingual corpus containing 47 million words developed by the Institute of Language and Speech Processing. It covers written Modern Greek since 1990. One feature of this corpus is that it allows the user to define the distance between three words, lemmas or parts of speech within the same query[5].

BwanaNet is an online corpus search tool developed to query a collection of specialized corpora from the Institut Universitari de Lingüística Aplicada (IULA) at the Universitat Pompeu Fabra. This collection of corpora includes original and parallel texts in Catalan, Spanish and English, from the fields of Computing, Environment, Law, Medicine, Genome, Economy, and other specialized areas.

This interface generates lists of word forms, lemmas or parts of speech. Users can search for concordances of one or more words, lemmas or parts of speech. Part of speech restrictions have two features: 1) the option to delimit, in a grammatical construction, the number of subsequent occurrences of the same category (between 0 and 9), and 2)

[2] Alcina [2] presents a didactic proposal divided into four levels of specialization in Computerized Terminology. In this proposal she includes training to query online corpora or other formats, as well as the use of corpora search tools.

[3] For more information on semantically-oriented searches, see Davies [7]. We include examples of this type of search in our search technique classification.

[4] Available at http://bncweb.lancs.ac.uk after registration.

[5] Most corpora allow distance to be defined between two elements only.

the search for a word form or lemma that excludes a particular part of speech. The user can also limit the search to a section of the corpus (titles, lists, tables, text). Other queries can be carried out using the Corpus Query Processor language[6].

COMPARA is an online bidirectional aligned corpus of English and Portuguese. To query this corpus, the user needs to be familiar with the CQP language. The interface allows the user to limit the search to linguistic variants of Portuguese or English, date, author, etc. In addition, concordance formats can be modified, for instance by displaying alignment properties or part-of-speech tags.

4. Classification of search techniques in corpora

Search techniques are options that a user can apply to a resource to obtain a result. We distinguish three elements in a search technique: a query probe, a query resource and a query outcome. The *query probe* is the word or phrase introduced by the user in the interface of a resource. The *query resource* is the resource or part of the resource in which the word or phrase is searched. The *query outcome* is the result obtained in a query when a probe is searched in a resource.

In this paper we present a classification of search techniques in electronic corpora that focuses on the query probe, the query resource and the query outcome. An example of a corpus search technique could be to use an exact word as a probe, e.g., we look for the word *play* in an English monolingual corpus (resource) to obtain a list of concordances—the outcome—of the word *play*, which includes expressions such as *play the piano, play football* or *play the role of.*

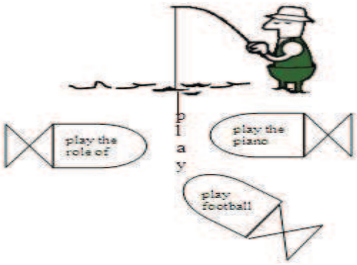


Figure 1. Representation of a search technique in an electronic corpus

Below, we explain in more detail the search techniques that can be used in an electronic corpus, and provide examples of how these search techniques are applied in the corpora analyzed.

4.1 Query probe

The query probe is an expression the user tries to find by introducing it in a corpus interface. We categorize query probes as follows: lexical expressions, grammatical expressions, numbers, hybrid expressions, and non-continuous combinations of expressions. Filters may be applied to restrict probes.

4.1.1 Lexical expressions

Lexical expressions can take a single form or a lemma, or a sequence of forms or lemmas.

A **lemma** is the base form of a word, i.e., it is a word without inflectional morphemes. The lemma of a noun is its form with no gender or number morphemes. The lemma of a verb is the infinitive.

A lemma is a useful way of retrieving all the forms in a corpus that are tagged with that lemma. For example, if we introduce the lemma *do* in the BYU-British National Corpus, the corpus retrieves all the forms of this verb that appear in the corpus: *do, did, does, done, doing, etc.*

A **form** can be exact or partial. An **exact form** is a complete word. It can be useful for finding a particular form of a word in the corpus. For instance, we can search the plural word *houses* in any of the corpora analyzed.

A **partial form** is an incomplete word. The omitted part of the word is replaced by a wildcard. The most frequent wildcards are the asterisk (*), which replaces one or more characters, and the question mark (?), which replaces only one character. Partial forms can be useful if we want to search all the words that start, end or contain a specific sequence of characters. For example, if we introduce the partial form *hous** in the COMPARA corpus, the following complete forms are retrieved: *house, housewife, housekeeper, house-doctor, houses, housing, household*, etc.

Lexical expressions can also be **sequences of two or more forms or lemmas**, which may be exact or partial. An **exact sequence** is a phrase or combination of forms or lemmas that appear in the corpus in the same order as those searched for. Exact sequences can be introduced to see the context in which a particular expression is used. In the following example we introduce the exact sequence of the forms *raining cats and dogs* in the BNC. Two contexts are retrieved: 1) “It was raining cats and dogs and the teachers were running in and out helping us get our stuff in and just couldn’t do enough for us.” 2) “What must you be careful of when it’s raining cats and dogs?”

A **partial sequence** is a combination of forms or lemmas in which one or more forms or lemmas are replaced by a wildcard. It can be used to search for an expression when we only know some of the words contained in it. In this example we introduce a partial sequence in the BYU Corpus del español: the verb *llover* as a lemma, followed by the preposition *a,* and then a wildcard, [llover] a *. Our search results include Spanish

[6] The Corpus Query Processor (CQP) manual is available at http://www.ims.uni-stuttgart.de/projekte/CorpusWorkbench/CQPUserManual/HTML/.

expressions referring to ‘raining heavily’, such as *llovía a cántaros, llueve a torrentes, lloviendo a mares, lloviendo a raudales, lloviendo a chuzos, llovía a baldes,* etc.

Frequent sequences representing concept relationships, also called linguistic patterns[7], can also be introduced. Some of these patterns can be used to retrieve, for instance, defining contexts in a corpus (*is a, known as, is defined*, *is called,* etc.).

4.1.2 Grammatical expressions

Grammatical expressions are constructions made up of parts of speech. They may contain a single part of speech or a sequence of parts of speech. Grammatical expressions can be useful to find words or sequences of words by introducing their parts of speech in the corpus.

This search technique is a feature of BwanaNet, BNC, BYU corpora, COMPARA and HNC. For example, if we introduce the grammatical expression “adjective+noun” in the BwanaNet English Law corpus, the following expressions are retrieved: *commercial legislation, fiscal protection, Social Fund,* etc.

4.1.3 Numbers

Numbers can be exact or partial. If we introduce an **exact number** in the corpus, it is retrieved in the same form as it was introduced. A **partial number** is a number combined with a wildcard. In this case, the corpus retrieves all the numbers containing the sequence of numbers introduced with the wildcard. A number search can be useful to find words that appear in the same context as a significant number. For instance, if we introduce the number 640 in the BwanaNet Spanish Computing corpus, the word *píxel* appears, because the corpus retrieves the typical computing measurement *640×480 píxels*; the term *memoria RAM* is also found, because another specific measurement retrieved by the corpus is *640 Kb de memoria RAM.*

4.1.4 Hybrid expressions

Hybrid expressions combine lexical expressions, grammatical expressions and numbers. They can be useful to find expressions in which we know the form or the lemma of some of the words and the part of speech of other words. For example, we introduce in the BwanaNet English Law corpus a hybrid expression made up of a grammatical expression followed by a lexical expression: “adjective+*law*”. The following expressions are retrieved *organic law, civil law, common law, Federal law, budgetary law*, etc.

4.1.5 Non-continuous combination of expressions

This search technique consists of introducing an element in a corpus and establishing the distance in which a second element must also appear. The first and the second element can be any of the query probes explained above: a lexical expression, a grammatical expression, a number or a hybrid expression.

In the following example we combine two lexical expressions in the BNC within a distance of 5 positions. The first expression is the form *Cytomegalovirus* and the second, an exact sequence of forms, the linguistic pattern *is a*. As a result we obtain some defining contexts of the word *Cytomegalovirus*.

Table 1. Results of the search for the form *Cytomegalovirus* within 5 positions of distance from the linguistic pattern *is a*

Cytomegalovirus (CMV) **is a** virus with many similarities to the herpes virus.
Cytomegalovirus **is a** less well-known infection which affects considerably greater numbers of babies than rubella.

In another example we use the BYU Corpus del español to combine the exact form *metros* within a distance of 5 positions from the number 100. Results include the expressions *100 metros libre* (100 meters free style) or *100 metros cuadrados* (100 square meters).

Table 2. Results of the search for the number 100 within 5 positions of distance from the form *metros*

terminó ayer su participación en Phoenix, Arizona, con el quinto lugar en los **100 metros** libre. […]
) del lugar, con una área mínima de construcción de 200 **metros** cuadrados y **100 metros** cuadrados para parqueo. […]

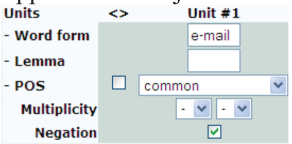
In this example, we use the BYU Corpus of Contemporary American English to combine the exact form *brake* within 3 positions of distance from the part of speech “verb”. Results include expressions such as: *have a brake, have to brake, set the brake, released the hand brake,* etc.

The Hellenic National Corpus is the only corpus analyzed that allows the user to combine more than two elements noncontinuously without having to be familiar with the CQP interrogation language. Three forms, lemmas, numbers or parts of speech can be combined within 5 positions of distance.

4.1.6 Query probe filters

Filters add a search restriction to the query probe introduced, such as **part-of-speech filters**. For example, BwanaNet allows the user to search for forms or lemmas that may or not belong to a particular part of speech.

The following figure shows a query in the BwanaNet English Computing corpus. We introduce the form *e-mail* with the exclusion of the part of speech “noun” (option *negation* below the box *POS*). The result is a concordance in which *e-mail* appears as an adjective.



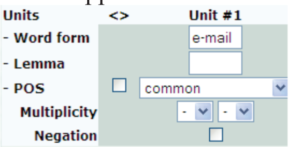
Units	<>	Unit #1
- Word form		e-mail
- Lemma		
- POS	☐	common
Multiplicity		- -
Negation		☑

[…] it can be obtained through CD-ROM, **e-mail** server, […].

[7] Many authors have studied linguistic patterns. See, for example, Sánchez [17], Faber et al. [8], López and Tercedor [13] or Meyer [14].

Figure 2. Search for a form or lemma with the exclusion of a particular part of speech

In contrast, if we introduce the form *e-mail* and limit the search to nouns, the result is a list of concordances in which the form *e-mail* appears as a noun.



Units <> Unit #1
- Word form: e-mail
- Lemma:
- POS ☐ common
Multiplicity - -
Negation ☐

By sending **e-mail** to clinton-info@campaign92.org, I was able to request press releases on foreign policy.	
[…] suggestions on how to use everything from **e-mail** to remote databases, tutorials, lists of frequently asked questions, […].	

Figure 3. Search for a form or lemma as a particular part of speech

Part-of-speech filters are also available in the BNC, HNC, COMPARA corpus, and BYU corpora.

4.2 Query resource

The corpus resource is always the collection of texts that constitute that corpus. Nevertheless, depending on the query probe and outcome of a search technique, we can distinguish different types of electronic corpora[8]: monolingual corpora, aligned corpora and tagged corpora. Filters can also be used to restrict the corpus.

4.2.1 Monolingual corpora

An electronic corpus can be **monolingual,** i.e., all the texts in the corpus are written in the same language. In this type of corpora the query probe and outcome will always be in one specific language.

4.2.2 Aligned corpora

An electronic corpus can also be **aligned.** An aligned corpus is a parallel corpus composed of source texts and their translations. The introduction of query probes in monolingual and aligned corpora is usually the same but the query outcomes obtained vary.

In aligned corpora the query probe is normally introduced in one of the corpus languages, as in monolingual corpora. However, in aligned corpora, search results include the segments in one language, as well as equivalent segments in the second language. For example, when we introduce the form *run* in the English part of the COMPARA corpus, the concordances in English are given with the form *run* highlighted in bold. Equivalent segments in Portuguese appear next to the concordances of the form *run*. The equivalents of *run* (*montar, correr* and *comprar*) are not highlighted.

Table 3. Results of the search of the form *run* in the aligned corpus COMPARA

I said to Nizar, «You could probably **run** a little rental business […]»)	Em resposta, sugeri ao Nizar: «Talvez pudesse montar um negócio de aluguer […]»
We had to **run** for a train once at Euston: […]	Uma vez em Euston tivemos de correr para apanharmos um comboio: […]
Neither of our families could afford to **run** a car in those far-off days.	Naquele tempo nem a minha família nem a dela tinham dinheiro para comprar carro.

Some aligned corpora also allow the user to introduce query probes simultaneously in both corpus languages. The COMPARA corpus offers an alignment restriction option that allows the user to introduce a query probe in one language with the condition that its equivalent segments contain another query probe. In the following example we introduce the form *run* in the English part of the COMPARA corpus, and the lemma *correr* in the Portuguese part. The corpus retrieves concordances of the form *run* in English whose equivalent segments in Portuguese include the lemma *correr.* In the concordances both the form *run* and all the forms of the lemma *correr* are highlighted in bold.

Table 4. Results of the simultaneous search for the form *run* in English and the lemma *correr* in Portuguese in the aligned corpus COMPARA

We had to **run** for a train […]	[…] tivemos de **correr** para apanharmos um comboio […]
If they had broken into a **run**, […].	Se tivessem desatado a **correr**, […].
But I feel we **run** a grave risk by doing so.	Mas eu acho que **corremos** um risco grave se o fizermos.

4.2.3 Tagged corpora

Corpora may either be tagged or not, and if they are, they may be tagged at different levels. In **POS-tagged corpora** all the words are tagged with their part of speech. Grammatical expressions can only be introduced in tagged corpora. In **lemmatized corpora** all the words in the corpus are tagged with their lemma. Lemmas can only be introduced in lemmatized corpora.

4.2.4 Query resource filters

The corpus search can be restricted to one section of the corpus using filters, such as thematic field, text type, geographic area, author, date, and text area.

The **thematic field filter** limits the corpus to sections of a selected thematic field. The BwanaNet, CQP edition of the BNC, CREA, and CORDE corpora offer this option. The **text type filter** limits the search to texts of a specific genre. This filter is available in the CREA, CORDE, CQP edition of the BNC, and BYU corpora. The **geographic area filter** limits the search to texts from a specific language area. For example, in the COMPARA corpus the search can be restricted to Portuguese from Angola, Portugal, Brazil and Mozambique, or to English from South Africa, United Kingdom or the United States. The CQP edition of the BNC also offers a geographic area filter.

[8] Many authors have elaborated wider typologies of corpora in which all the types of corpora are described. See, for example, Corpas [6], Zanettin [24] and Sinclair [20]. In this paper, we have limited our classification of corpora according to what can we introduce in the corpus (the probe) and what can we obtain (the outcome).

The **author filter** limits the search to texts published by one or more authors. This filter is offered in the COMPARA, CREA and CORDE corpora. The **date filter** limits the search to texts published on a specific date or within a time period, and is a feature of the COMPARA, the CQP edition of the BNC, CREA, CORDE and BYU corpora. The **text area filter** limits the search to titles, lists, tables, etc. BwanaNet offers this filter. The CQP edition of the BNC allows the user to search in titles and keywords.

4.3 Query outcome

When an electronic corpus is queried, the user can select different types of query outcomes depending on the result he/she desires. These query outcomes may be a list of monolingual or aligned concordances, a list of words, a list of synonyms, a list of collocates or a list of clusters.

4.3.1 List of concordances

Concordances are the contexts in which the query probe appears. Most of the corpora provide concordances in an easy to read format called KWIC (key words in context), which means that the query probe is highlighted in the center of the context. Depending on the query resource used in a search technique, lists of concordances can be monolingual or bilingual.

4.3.1.1 List of monolingual concordances

Monolingual corpora can generate lists of monolingual concordances, i.e., lists of contexts in one language. Monolingual concordances are mainly used to observe a word in context. For example, if we look for the concordances of the lexical expression *Prime Minister* in the BNC we access contexts in English where this expression is used.

Another function of concordances is to find a word by searching for words that appear in a nearby context. For example, we can search in the BYU-OED Oxford English Dictionary corpus to find a word that refers to "a case where an archer holds arrows". In this case, we introduce the lemma *arrow* within 9 positions of distance to the part of speech "noun". The corpus retrieves concordances of nouns appearing near the forms of *arrow*; one of these nouns is the word *quiver*.

Table 5. Concordances of the lemma *arrow* within 9 positions of distance from the noun *quiver*

A gaily-painted **quiver**, full of **arrows**
He could draw an **arrow** from his **quiver** […]

4.3.1.2 List of bilingual concordances

Aligned corpora can generate lists of bilingual concordances, which are lists of contexts in one language with equivalent contexts in another language. Bilingual concordances allow the user to decide on a more reliable translation equivalent because both the query probe in the source language and its equivalent in the target language are situated in a context that can be compared with the context of the translation, thus allowing the translator to verify equivalence.

In the following example we introduce the form *play* in the COMPARA aligned corpus and search for its concordances. Depending on the context, *play* is translated in Portuguese as *tocar* (when it refers to a music instrument)*, jogar* (when it refers to a sport) or *fazer a* (when it refers to playing a role).

Table 6. List of concordances of the form *play* with its equivalents in Portuguese (highlight in Portuguese concordances added)

«[…] and not being able to **play** the piano.»	«[…] e à incapacidade de **tocar** piano.»
Joe wanted to switch partners and **play** the best of three sets, […]	Joe queria trocar de parceiros e **jogar** de novo, uma melhor de três, […]
([…] he likes to **play** the father in our relationship.)	([…] gosta de **fazer a** figura paterna no nosso relacionamento.

4.3.2 Word lists

There are two types of word lists. One type includes the most frequent words in a corpus. The other is a list of keywords, which are extracted by comparing the word frequency lists of two corpora; the result is a list of words that are typical of one corpus, which are different from the other corpus[9].

Word lists can provide a useful overview of the specific terminology in a field. Of the corpora analyzed, BwanaNet provides lists of words with the option *isolated tokens*. The lists in BwanaNet may be of forms, lemmas or parts of speech. The BYU corpora also generate word lists. In these corpora, the user must introduce a part of speech and the corpora generate word lists that are tagged with that part of speech. The CQP edition of the BNC generates word or lemma frequency lists and allows the user to limit the lists introducing word patterns or using part-of-speech filters. This corpus also generates lists of keywords comparing the frequency lists of the whole BNC, the written BNC, and the spoken component of the BNC.

For example, if we generate a list of lemmas in the BwanaNet English Economy corpus, the first lemmas in the list are, logically, general language words, mainly prepositions and articles, since these are the most frequent words in every corpus. However, the sign = appears at the top of the list, as a typical component of economic texts. Other words from this field, such as *rate*, *market, price, good, capital, investment,* etc, also appear near the beginning.

4.3.3 List of synonyms

Some corpora have incorporated semantically-based searches. This option allows the user to find synonyms for the word introduced. Of the corpora analyzed in this study, only the BYU corpora provide this option.

[9] These list types are extracted from specialized corpora which are compared with general language corpora, known as *reference* corpora.

In the following example, we use the BYU Corpus of Contemporary American English to search for synonyms of the form *beautiful*, by introducing [=beautiful]. A list of synonyms is provided: *wonderful, attractive, striking, lovely, handsome,* etc. The frequency of each synonym in the corpus and access to concordances of the synonyms are given. We can also compare the frequency and distribution of the synonyms in the corpus by text type and dates.

4.3.4 List of collocates

A collocate is a word that frequently appears near another word. Lists of collocates can be useful to access the words in the context of a term without having to read all its concordances. This function helps to speed up the search process.

In all the corpora, collocates of a word can be seen by reading all the contexts of that word. However, of the corpora analyzed, only the BYU corpora generate lists of word collocates in which the part of speech of the collocate is specified. For example, if we search in the BYU Corpus of Contemporary American English for the noun collocates of the form *television*, the retrieved list of collocates includes: *radio, news, show, cable, network, station, series,* etc.

Collocates of word synonyms can also be accessed. For instance, the BYU Corpus del español lists the nouns that appear near the synonyms of *sucio* (dirty); retrieved collocates include the words *pocilga* (pigsty) or *tugurio* (hovel or dive).

4.3.5 List of clusters

Clusters are sequences of two or more words that are frequent in a corpus. Various query probes can be introduced in a search for clusters. We may choose not to specify a query probe and only specify the number of words we want the cluster to have (two or more). We can specify a word that must appear in the cluster, for instance *mesita* (table). We can also specify the grammatical sequence of the cluster, for example clusters of "noun+adjective+adjective". Words and parts of speech that must be included in the clusters can also be specified, for example "mesita+preposition+adjective".

Lists of clusters can provide a useful overview of how terminology is frequently combined in a field. They can also be used to find a word if we know other words it is frequently combined with, or a typical construction in which the word appears.

Of the corpora analyzed, BwanaNet generates two-word clusters without specifying a query probe. The BYU corpora retrieve clusters specifying words or parts of speech that must appear in the clusters. CREA and CORDE generate clusters specifying one or more words that must appear in the clusters. For example, in the CREA corpus we can search for clusters of three words that include the word *mesita* (table). The retrieved list includes the following clusters: *mesita de noche* (bedside table)*, mesita de madera* (wooden table)*, mesita de luz* (lamp table)*, mesita del teléfono* (telephone table), etc.

5. Conclusion

This study has shown how search techniques can vary from one corpus to another. Within the context of translator training in the use of corpora, there is a need to systematize the search techniques that can be used in electronic corpora. The classification of search techniques provided in this paper, focusing on the query probe, resource and outcome, attempts to meet that need. These three elements have been considered to explore the range of search possibilities corpora offer.

Table 7. Classification of search techniques in corpora

QUERY PROBE	QUERY RESOURCE	QUERY OUTCOME
- Lexical expression - Grammatical expression - Numbers - Hybrid expression - Non-continuous combination of expressions ○ Probe filters	- Monolingual corpora - Aligned corpora - Tagged corpora ○ Resource filters	- List of monolingual or bilingual concordances - Word list - List of synonyms - List of collocates - List of clusters

SEARCH TECHNIQUES

Although our search technique classification is subject to further additions and variations, it has two main applications. First, it will help us to reflect on the most useful search techniques for translators, thus enabling us to consider improvements in corpora to adapt these resources to translators needs. Second, it may serve as a guide in teaching translation students search techniques in electronic corpora.

6. Corpora examined

British National Corpus, version 3 (BNC XML Edition). (2007). Distributed by Oxford University Computing Services on behalf of the BNC Consortium. Available at <http://www.natcorp.ox.ac.uk/> [20/05/09].

Bwananet, Programa de explotación del corpus técnico del IULA (Universitat Pompeu Fabra). Available at < http://brangaene.upf.es/bwananet/indexes.htm> [22/05/09].

COMPARA. Available at < http://www.linguateca.pt/COMPARA/> [19/05/09].

Davies, M. (2002-). *Corpus del Español* (100 million words, 1200s-1900s). Available at <http://www.corpusdelespanol.org> [16/05/09].

Davies, M. (2004-). *BYU-BNC: The British National Corpus*. Available at <http://corpus.byu.edu/bnc>[16/05/09].

Davies, M. (2007-). *TIME Magazine Corpus* (100 million words, 1920s-2000s). Available at <http://corpus.byu.edu/time> [16/05/09].

Davies, M. (2008-). *The Corpus of Contemporary American English (COCA)*: 385 million words, 1990-present. Available at <http://www.americancorpus.org> [16/05/09].

Davies, M. (2009-). *BYU-OED: The Oxford English Dictionary*. Available at <http://corpus.byu.edu/oed> [16/05/09].

Davies, M. and M. Ferreira. (2006-). *Corpus do Português* (45 million words, 1300s-1900s). Available at <http://www.corpusdoportugues.org> [16/05/09].

Hellenic National Corpus (HNC). Available at < http://hnc.ilsp.gr/en/info.asp> [19/05/09].

Real Academia Española: *Corpus de referencia del español actual* (CREA) *Corpus de referencia del español actual.* Available at <http://www.rae.es> [01/05/09].

Real Academia Española: *Corpus diacrónico del español* (CORDE). Available at <http://www.rae.es> [01/05/09].

7. References

[1] Alcina Caudet, A. (forthcoming). "Metodología y tecnologías para la elaboración de diccionarios terminológicos onomasiológicos", in A. Alcina Caudet *Terminología y sociedad del conocimiento*. Bern, Peter Lang.

[2] Alcina Caudet, A. (2003). "La programación de objetivos didácticos en Terminótica atendiendo a las nuevas herramientas y recursos", in N. Gallardo San Salvador *Terminología y traducción: un bosquejo de su evolución*. Granada, Atrio.

[3] Bowker, L. (1998). "Using Specialized Monolingual Native-Language Corpora as a Translation Resource: A Pilot Study", *Meta* 43(4): 631-651.

[4] Bowker, L. (2000). "Towards a Methodology for Exploiting Specialized Target Language Corpora as Translation Resources", *International Journal of Corpus Linguistics* 5(1): 17-52.

[5] Bowker, L. and J. Pearson (2002). "Working with Specialized Language. A practical guide to using corpora", London/New York, Routledge.

[6] Corpas Pastor, G. (2004). "Localización de recursos y compilación de corpus vía Internet: aplicaciones para la didáctica de la traducción médica especializada. Manual de documentación y terminología para la traducción especializada", in C. Gonzalo García and V. García Yebra. Madrid, Arcos/Libros: 223-274.

[7] Davies, M. (2005). "The advantage of using relational databases for large corpora. Speed, advanced queries, and unlimited annotation", *International Journal of Corpus Linguistics* 10(3): 307-334.

[8] Faber, P., C. López Rodríguez and M. I. Tercedor Sánchez (2001). "Utilización de técnicas de corpus en la representación del conocimiento médico", *Terminology* 7(2): 167-197.

[9] Fraser, J. (1999). "The Translator and the Word: The Pros and Cons of Dictionaries in Translation", in G. Anderman and M. Rogers *Word, Text, Translation. Liber Amicorum for Peter Newmark*. England, Multilingual Matters.

[10] Gallardo San Salvador, N. and A. de Irazazábal (2002). "Elaboración de un vocabulario multilingüe del campo temático de la siderurgia", in A. Alcina Caudet and S. Gamero Pérez *La traducción científico-técnica y la terminología en la sociedad de la información*. Castellón, Publicaciones de la Universitat Jaume I: 189-198.

[11] Hull, D. (2001). "Software tools to support the construction of bilingual terminology lexicons", in D. Bourigault, C. Jacquemin and M.-C. L'Homme *Recent Advances in Computational Terminology*. Amsterdam/Philadelphia, John Benjamins: 225-244.

[12] Kraif, O. (2008). "Extraction automatique de lexique bilingue: application pour la recherche d'exemples en lexicographie", in F. Maniez, P. Dury, N. Arlin and C. Rougemont *Corpus et dictionnaires de langues de spécialité*. Bresson, Presses Universitaires de Grenoble.

[13] López Rodríguez, C. I. and M. I. Tercedor Sánchez (2008). "Corpora and Students' Autonomy in Scientific and Technical Translation Training", *The Journal of Specialised Translation*, 9. Available at <http://www.jostrans.org/issue09/art_lopez_tercedor.pdf>. [25/05/09].

[14] Meyer, I. (2001). "Extracting knowledge-rich contexts for terminography: a conceptual and methodological framework", in D. Bourigault, C. Jacquemin and M.-C. L'Homme: *Recent Advances in Computational Terminology*. Amsterdam/Philadelphia, John Benjamins: 279-302.

[15] Montero Martínez, S. and P. Faber Benítez (2008). *Terminología para traductores e intérpretes*. Granada, Tragacanto.

[16] Nesi, H. (1999). "A User's Guide to Electronic Dictionaries for Language Learners", *International Journal of Lexicography* 12(1): 55-66.

[17] Sánchez Gijón, P. (2003). *Els documents digitals especialitzats: utilització de la lingüística de corpus com a front de recursos per a la traducció*. Thesis available at < http://www.tdx.cbuc.es/ >. Barcelona, Universidad Autónoma de Barcelona.

[18] Schäffner, C. (1996). "Parallel Texts in Translation". *Unity in Diversity? International Translation Studies Conference*. Dublin City University. 9-11 May 1996

[19] Shreve, G.M. (2001). "Terminological Aspects of Text Production", in S.E. Wright and G. Budin *Handbook of Terminology Management. Volume 2. Application-Oriented Terminology Management*. Amsterdam/Philadelphia, John Benjamins.

[20] Sinclair, J. M. (1996). "EAGLES Preliminary recommendations on Corpus Typology, EAG-TCWG-CTYP/P". Available at <http://citeseer.ist.psu.edu/cache/papers/cs/21540/ftp:zSzzSzftp.ilc.pi.cnr.itzSzpubzSzeagleszSzcorporazSzcorpustyp.pdf/eagles-preliminary-recommendations-on.pdf > [01/05/09]

[21] Snell-Hornby, M. (1988). *Translation Studies. An Integrated Approach*. Amsterdam/Philadelphia, John Benjamins.

[22] Varantola, K. (1998). "Translators and their use of dictionaries", in B. T. S. Atkins *Using Dictionaries*. Tübingen, Niemeyer: 179-192.

[23] Zanettin, F. (1998). "Bilingual Comparable Corpora and the Training of Translators", *Meta* 43(4): 616-630.

[24] Zanettin, F. (2002). "Corpora in Translation Practice", in *Proceedings of the First International Workshop on Language Resources (LR) for Translation Work and Research*. Las Palmas de Gran Canaria.

HMMs, GRs, and n-grams as lexical substitution techniques – are they portable to other languages?

Judita Preiss
Department of Linguistics
The Ohio State University
judita@ling.ohio-state.edu

Andrew Coonce
The Ohio State University
coonce.3@osu.edu

Brittany Baker
The Ohio State University
baker.1189@osu.edu

Abstract

We introduce a number of novel techniques to lexical substitution, including an application of the Forward-Backward algorithm, a grammatical relation based similarity measure, and a modified form of *n*-gram matching. We test these techniques on the SEMEVAL-2007 lexical substitution data [McCarthy and Navigli, 2007], to demonstrate their competitive performance. We create a similar (small scale) dataset for Czech, and our evaluation demonstrates language independence of the techniques.

Keywords

Lexical substitution, synonyms, Google n-gram corpus, grammatical relations, HMMs, Forward-Backward algorithm, Lucene, Czech, word sense disambiguation

1 Introduction

We present a number of novel approaches to lexical substitution, a task which for a given target word, requires the selection of a suitable alternative word. Our highly modular system not only allows a trivial addition of new modules, but also explores the applicability of the techniques to English and Czech.

Lexical substitution was suggested as a way of evaluating word sense disambiguation [McCarthy, 2002], accounting for the difficulties with selecting a sense inventory in the traditional direct sense evaluations (e.g., [Preiss and Yarowsky, 2002]). In the lexical substitution task, instead of being presented with a set of possible senses to choose from, a system is given a word and is required to find a suitable alternative given the context. For example, the word *bright* in the sentence

> His parents felt that he was a bright boy.

can be replaced with the word *intelligent*. However, the same substitution for the same word in the context of the word *star* (e.g., in the sentence *Our Sun is a bright star.*) is unlikely to reflect the intended meaning. The applications of a system capable of making such substitutions lie in question answering, summarisation, paraphrase acquisition [Dagan et al., 2006], text simplification and lexical acquisition [McCarthy, 2002].

An evaluation task was set up as part of SEMEVAL-2007 evaluation exercises [McCarthy and Navigli, 2007], in which participants were given a target word and its context and were expected to find a suitable substitutable word or phrase. A second task is proposed for SEMEVAL-2010 [Sinha et al., 2009], which expects participants to select a possible substitute from another language, given an input word and context in English.

As with many natural language processing tasks, most work on lexical substitution has been carried out in English. As the lexical substitution task requires an annotated corpus, it is non-trivial to carry out large-scale experiments in other languages. We create a small corpus for Czech, and evaluate our lexical substitution modules[1] not only on the SEMEVAL-2007 lexical substitution data in English, but also on our Czech dataset. Unlike the proposed [Sinha et al., 2009] cross-lingual lexical substitution task in SEMEVAL-2010, in our experiment the target words and contexts as well as substitute are all in Czech.

For English, we demonstrate

1. the importance of refining the set of input candidate substitutes prior to a candidate ranking module being run, and
2. show our modules' suitability to be used in lexical substitution tasks

We create a communal set of candidates, which are used by three independent modules: a grammatical relation [Briscoe et al., 2002] based module investigating the syntactic (and to a certain extent semantic) similarity of contexts, an *n*-gram module exploiting the Google Web 1T 5-gram corpus [Brants and Franz, 2006], and a module discovering the optimal path through the sentence based on the Forward-Backward HMM algorithm (e.g., [Roark and Sproat, 2007]).

[1] Note that such an evaluation is not possible for all of our modules, due to the lack of available tools for the Czech language.

Our paper is organized as follows: Section 2 describes the technique used to create a weighted candidate set, Sections 3, 4, and 5 contain the GR, n-gram and HMM modules respectively. An initial evaluation on English is presented in Section 6. Our experiment on Czech, and the data used to enable this, appears in Section 7, with conclusions drawn in Section 8.

2 Building a candidate set

We create a very modular system, where each of our lexical substitution selection methods is entirely independent of the others. The modules share a common input: the possible set of candidates for each word. In his work, [Yuret, 2007] presents the most successful system in Semeval-2007 and comments that "I spent a considerable amount of effort trying to optimize the substitute sets". We therefore explore performance with two different candidate sets to investigate the hypothesis that the approach used is as important as the candidates selected.

The first approach, which finds the maximum possible performance of the modules (an upper bound), is given all candidates which appeared for the target word in the gold standard data. I.e., all the possible substitutes from the gold standard are gathered together, and given to the modules as possible candidates. (However, as no module is designed to cope with multiword candidates, all the multiword candidates are removed.)

Our second set of candidates is constructed from WordNet [Miller et al., 1990] and the online encyclopedia Encarta (`http://encarta.msn.com`) as follows:

- All WordNet (WN) synonyms of the target word are included (i.e., synonyms of all the possible senses of the correct part of speech)[2].
- The hypernym synset and the hyponym synset are also included for all possible senses.
- All possible Encarta synonyms of the correct part of speech are extracted.

A probability distribution is placed on these candidates based on these (manually selected) weights:

Source	Weight
WN synonym	3
WN hypernym	1
WN hyponym	2
Encarta	3

I.e., if a candidate appears both as a WN synonym and in Encarta, it will get a weight of 6, while if it is only appearing as a hyponym, it's weight will be 2. For example, for the test word *account* (noun):

1. WN synonyms: *history, chronicle, story, bill, invoice, report, story, explanation, . . .*
2. WN hypernyms: *record, importance, profit, gain, statement, . . .*
3. WN hyponyms: *etymology, annals, biography, life, recital, reckoning, tally, . . .*
4. Encarta: *report, description, story, narrative, explanation, version, interpretation, tally, . . .*

PoS	Average
Noun	56
Verb	127
Adjective	37
Adverb	9

Table 1: *Average number of candidates*

the Encarta synonyms add new candidates, while also boosting the weights of, e.g., the synonym *story*, or the hyponym *tally*. Once all the candidates for a target word are generated, the weights are converted into a probability distribution.[3] The average numbers of candidates for each part of speech are presented in Table 1.

While the GR and the n-gram modules only require a set of candidates for the target words, the HMM module requires potential candidates for all words in the sentence in order to find an optimal path through the data. These candidates were generated in the same manner, with PoS tags drawn from the [Elworthy, 1994] tagger (executed as part of RASP [Briscoe et al., 2006]), with the exception that for the non-target words, the original word was also included in the candidate set.

3 Grammatical relations

Given the candidates generated in Section 2, we create several different (hopefully complementary) modules. A combination of these can then utilize the different strengths and weakness of each approach to create a more accurate ranking of proposed candidates overall. The modules can therefore run independently to select the most likely of any given candidates.

For each target word, its context was parsed with RASP [Briscoe et al., 2006] producing grammatical relations (GRs). GRs, mainly binary relations expressing information about predicates and arguments, provide a good means for capturing both syntactic structural information, but also some sense of semantic meaning as well [Briscoe et al., 2002]. GR such as

```
(dobj give dog)
```

where the GR is `dobj` (direct object), it not only tells us that give directly dominates dog (syntax), but there is also a description about a patient relationship.

The main advantage of GRs, as opposed to, for example, n-grams, is the possibility of GRs encoding long distance dependencies. Even with simple sentences, such as:

- *Bob Smith gave the bone to the dog.*
- *Bob Smith gave the big juicy bone to the dog.*

[2] The part of speech of the target word is given in the data.

[3] The low hypernym score is due to relatively rare occurrence of the correct candidate being in the hypernym set.

the GRs will contain the `dobj give bone` relation for both sentences, while a five word n-gram centered on the target word *give* will not even mention the word *bone* in the second case.

The motivation behind this approach is in the assumption that a word which is a valid lexical substitute will appear in the same GRs as the target word. This requires a large corpus annotated with GRs: to this end we employ Gigaword [Graff, 2003], a large collection of English text, which we parsed with the RASP parser and collected information about frequencies of GR occurrences. The GR occurrences are indexed using Lucene, to allow incremental building and searching of the dataset. Each word can be queried, producing a listing of every applicable GR in which said word appeared in the Gigaword corpus, along with a frequency count of occurrence(s) for each GR. A preliminary search was performed on this index to obtain initial probabilities for each GR.

For each target word, all the GRs from its context are extracted and the target word is substituted with a possible candidate. The frequency of this GR is extracted from the parsed corpus, and divided by the probability of that GR, in order to account for unequal GR occurrences throughout the index (the GR `ncmod`, for example, appeared many times more than the GR `iobj`). For each candidate, all its GR frequency weights are summed, and the weights are normalized to produce a probability distribution on candidates.

4 n-grams

Approaches based on n-grams drawn from the Google Web1T corpus [Brants and Franz, 2006] have been shown to constitute a particularly good approach to lexical substitution with the best performing system in SEMEVAL-2007 being n-gram based [Hassan et al., 2007]. The basic algorithm for such an approach is very clear: an n-gram containing the chosen word is extracted from the context, the chosen word is then replaced with a candidate and the frequency of the newly formed n-gram is found. The candidate with the highest frequency wins.

For this work, we use the Google Web1T corpus, a publicly available resource, containing frequency information about 1, 2, 3, 4 and 5-grams drawn from one trillion (10^{12}) words of English Web text, subject to a minimum occurrence threshold. While such a corpus is obviously a very valuable resource, it has been found previously that it is difficult to use due to its sheer size (it is 25Gb in compressed form). In order to provide a reasonable access time (and multiple wildcard searches), we treated each 5-gram as a separate document and indexed the 5-gram corpus with the publicly available tool Lucene (available from `http://lucene.apache.org`).[4]

For a word w with possible candidate substitutions $s_1, s_2, \ldots, s_n$, we exploit a 5 word window W centered around w in the following way for each s_i:

- We search for the frequency ($f_{5-gm}(s_i)$) of the 5-gram W with w replaced with s_i.
- The replaced 5-gram is also searched in a stoplisted form ($f_{stop}(s_i)$). Note that this can result in a much smaller n-gram.
- The frequencies of all consecutive subset 4-grams (with the target word w replaced with s_i) are extracted ($f_{4-gm_j}(s_i)$ for $j = 1, \ldots, 4$).
- The absolute frequency of the unigram s_i is also retrieved ($f_{1-gm}(s_i)$). A more frequent unigram is more likely to be found as part of a 5-gram or 4-gram, purely due to the frequency of occurrence. This factor allows us to remove this bias.

The resulting weight of each s_i is then expressed as shown in Figure 1.[5]

5 Hidden Markov Models

5.1 Introduction

Hidden Markov Models (e.g., [Roark and Sproat, 2007]) and, in particular, Forward-Backward Hidden Markov Models (HMMs), have a strong history of applications in linguistics. The justification for the applicability of a Hidden Markov Model to the problem of lexical substitution lies in both the limited number of possible substitutions and the large training corpus.

When compared to the issue of speech processing, for which a HMM is known to work as a reasonable model, the issue of lexical substitution is highly similar and can be expected to produce results of similar quality.

Meanwhile, the presence of the large training corpus[6] means that the transition probabilities can be calculated with a high degree of certainty for transitions between possible lexical substitutions.

5.2 Motivation

Compared to n-gram and grammatical relation (GR) models, the HMM introduce a few key distinctions which should have significant contributions to the quality of the substitution results. While the n-gram and GR algorithms are capable of comparing the likelihood of a lexical substitution in their respective contexts, they do not allow the non-target words to take on other senses in order to generate a better fit.

That said, the HMM lacks the ability of the GR model to consider the impact of grammar on the sentence. Furthermore, it does not benefit from the relative speed of implementation and execution enjoyed by n-grams.

The forward-backward algorithm allows the model to take into account both the words that preceded and followed the target word that was being disambiguated. In comparison, a Viterbi Algorithm would

[4] Note that subject to a predictably regular repetition, the information contained in the 2, 3, and 4-grams can be extracted from the 5-gram corpus.

[5] As this module is not expected to be acting alone, we are not making any adjustments for data sparsness.

[6] In this case, Google Web1T data is used to generate the transition probabilities.

$$p(s_i) = \frac{f_{5-gm}(s_i) + f_{stop}(s_i) + \sum_{j=1}^{4} f_{4-gm_j}(s_i)}{\sum_{k=1}^{n}(f_{5-gm}(s_k) + f_{stop}(s_k) + \sum_{j=1}^{4} f_{4-gm_j}(s_k)) + f_{1-gm}(s_i)}$$

Fig. 1: *The weight of each candidate s_i*

have limited the effectiveness of the solution to take into consideration words that follow the target word. For example, returning to the previous example

> Brian is a bright boy.

the key word in determining the proper lexical substitution of *bright* is *boy*. In this case, the Viterbi Algorithm would not be able to determine the proper substitution as the determining word *boy* follows the target word *bright*.

5.3 Algorithm

The key inputs to the HMM implementation are:

- S_i is one specific possible lexical substitution within the set of all possible substitutions S
- $B_{w_t i}$, or $P(W_t \rightarrow S_i)$, is the substitution probability of a word W_t by a substitution S_i[7]
- A_{ij}, or $P(S_i \rightarrow S_j)$, is the transition probability between two possible substitutions S_i and S_j[8]
- π_i, or $P(\emptyset \rightarrow S_i)$, is the probability that the model begins simulation in a given state S_i[9]

In implementation, the Forward-Backward Algorithm maximizes the product of the forward-looking matrix, α_{it}, the backward-looking matrix, β_{it}, and the lexical substitution probability, $b_{w_t i}$. The forward-looking matrix α_{it} measures the likelihood that the sentence is at state S_i, at time t, when the word w_t is registered. Likewise, the backward-looking matrix β_{it} measures the likelihood, given that the sentence is at state S_i at time t with probability α_{it}, that there is a valid transition path that reaches the end of the sentence. The lexical substitution likelihood probability $b_{w_t i}$ represents the relative, context-free probability that a given word w_t uses the substitution S_i.

Thus, the product $\alpha_{it} \times \beta_{it} \times b_{w_t i}$ represents the relative probability that the lexical substitution S_i is the intended sense of the word W_t seen in location t of the sentence. By comparing this product for each $S_i \in S$ and dividing the resulting values by the summation of the probabilities for all $S_i \in S$, the relative probabilities represent the likelihood that a specific word is the expected lexical substitution. The candidate with the highest likelihood estimation wins, though any substitution with a probability within two orders of magnitude of the winner is included as a possible solution for evaluation purposes.

5.4 Solution-Space Generalizations

In an ideal model, each sentence would be broken down into its constituent words and every possible substitution of each word would be a possibility interpretation. This idealized model would allow for all possible interpretations of the sentence, providing all possible frames with which to consider a given lexical substitution. Such a model would feature upwards of twenty possible substitutions per word with each requiring processing for all possible preceding and following substitutions.

The complexity of the HMM was found to be proportional to the square of the average number of possible lexical substitutions per word in its input sentences (see Table 2). This idealized model, though loss-less, proved computationally inefficient when scaled to the demands of the application, given the large percentage of time spent looking up transition probabilities in the training corpus. In order to minimize the total number of senses being processed without subjecting the model to unnecessary generalizations, two methods were used to reduce the solution complexity: sliding-window and sense-reduction generalizations.

The sliding-window generalization assumed that terms further from the target word would be less likely to contain useful information to disambiguate the target word sense. As such, a sliding-window representing likely relevant words was formed around each of the target words; any word not within the sliding-window had its possible word senses (expressed by lexical substitutes) reduced to unity while those within the window retained multiple senses.

The sense-reduction generalization assumed that word-senses with low probabilities would not contribute significant information to disambiguating the word-sense of the target word. As such, the senses were reduced by limiting possible sense for words within the sliding-window to only those senses that were common to both Encarta and WordNet.

6 Results

We evaluated various combinations of the above systems on the English lexical substitution data [McCarthy and Navigli, 2007], which contains substitution information about 171 nouns, verbs, adjectives and adverbs manually constructed by 5 native English speaker annotators. Each of our modules is capable of producing a probability distribution which allows us to investigate a number of possible combination techniques. All systems are given identical candidate sets as input, yielding two experiments:

1. Candidate set created from the gold standard (GS)

[7] The candidate sets, S, and their substitution probabilities, $B_{w_t i}$, are shared with the other applications discussed in this paper.

[8] The transition probability, A_{ij}, is generated from the Google 2-gram data set using Lucene.

[9] The initial state probability, π_i, is generated from the Google 1-gram data set using Lucene.

Without Sense-Reduction	545 lexical substitution candidates/sentence
Non-Target Sense-Reduction	88 lexical substitution candidates/sentence
Full Sense-Reduction	45 lexical substitution candidates/sentence
Without Rolling-Window	83 Lucene queries/sentence
With Rolling-Window	24 Lucene queries/sentence

Table 2: *Hidden Markov Model Computational Complexity*

Eval	System	Candidates	Precision	Recall	Mode precision	Mode recall
OOT	GRs	GS	63.49	7.23	71.05	8.78
Best	GRs	GS	5.58	0.64	6.58	0.81
OOT	HMMs	GS	52.74	43.41	63.41	52.28
Best	HMMs	GS	13.64	11.23	18.34	15.12
OOT	n-grams	GS	65.06	65.02	73.80	73.74
Best	n-grams	GS	12.31	12.30	17.33	17.32
OOT	Voting	GS	68.67	68.67	77.80	77.80
Best	Voting	GS	13.90	13.90	19.59	19.59
OOT	GRs	WNE	13.68	0.09	12.50	0.08
Best	GRs	WNE	1.82	0.01	0.00	0.00
OOT	HMMs	WNE	16.52	0.25	20.00	0.33
Best	HMMs	WNE	2.24	0.03	0.00	0.00
OOT	n-grams	WNE	35.79	8.90	48.11	12.44
Best	n-grams	WNE	6.92	1.72	11.01	2.85
OOT	Voting	WNE	36.07	8.98	48.43	12.52
Best	Voting	WNE	7.02	1.75	11.01	2.85

Table 3: *Results of each module on the English lexical substitution task*

2. Candidate set created from WordNet and Encarta as described in Section 2 (WNE).

The results of these evaluation can be found in Table 3. Two evaluations are presented:

1. **best**: Only the top candidate is evaluated against the gold standard (this corresponds to the highest probability candidate).

2. **oot**: The top ten candidates are collected and evaluated against the gold standard.

The results can be compared to the highest performing system in Semeval-2007 which achieved an oot precision / recall of 69.03 / 68.90, and mode precision / recall of 58.54, while the highest performing best system had a precision / recall of 12.90, and mode precision / recall of 20.65. (Note that the results for the WNE experiment are partial, as discussed in Section 6.1 representing only 10% of the data.)

6.1 Discussion

The single largest factor in the effectiveness of an approach to the problem space appears to be the proper determination of the scope of its candidate list. If an under-generated candidate set was used, the lexical substitutions suggested would be technically sound but incorrect insofar as they were only the best from the subset, not from the set of all possible substitutions. Omission of candidates could also reduce the number of valid substitutions to zero, creating a model where no candidate that remained would fit within the constraints imposed by the system evaluating its candidacy.

While under-generation was a concern, the candidate sets more directly suffered from over-generation. In over-generated candidate sets, the inclusion of rarely used substitutions, including hypernyms and hyponyms, served only to dramatically increase solution time without a corresponding increase in solution accuracy. As the complexity of the systems frequently increased proportional to the square of the average number of lexical substitution possibilities, these candidate sets quickly became disproportionately large when compared to the gold standard candidate sets. For such candidate sets that were fully evaluated, no noticable improvement was found in the ability to correctly identify the proper lexical substitution over the gold standard candidates.

These issues served as the motivations for proceding using the gold standard candidates (GS results) instead of the locally generated sets (WNE results). The gold standard candidates avoided the potential shortfall of under-generation as they were guaranteed to contain the anticipated substitution of the target word within their candidate sets; thus, protecting them from failing to produce a candidate selection. At the same time, the candidate list was also small enough to avoid the growth issues experienced in the over-generated candidate lists. Since the gold standard candidates do not overlap within their set, they are significantly more likely to feature a broad selection of possible candidates within the OOT, boosting the accuracy of the results. As we are merely interested in the performance of our modules (to demonstrate their suitability

Czech	PoS	Senses	English
cesta	n	5	way, path
číslo	n	6	number, performance
funkce	n	8	function, event
zůstat	v	6	stay, remain
těžký	a	6	hard, difficult
nechat	v	16	leave
důkaz	n	9	proof
povrch	n	7	surface
partie	n	12	part, partner
věc	n	6	thing
akce	n	7	action, event

Table 4: *Words selected for Czech lexical substitution including (some) English translations*

for the task and to enable their evaluation on the Czech lexical sample task), the use of the gold standard candidate sets is justifiable. Also, a properly generated candidate list would exhibit similar characteristics to this set.

7 Evaluation on Czech data

7.1 Creating the evaluation corpus

Unfortunately a lexical substitution corpus is not available for other languages. In an effort to investigate the applicability of our methods to other languages, we selected an extreme example: Czech, a highly morphologically rich, free order language, which should therefore produce a valuable comparison.

Ten words were selected at random from the online, publicly available, Czech Wiktionary[10] subject to the constraint that they had at least 5 senses listed (note that this step is completely automated, and could be executed with any language). The words chosen, along with the number of senses and their parts of speech in Wiktionary can be found in Table 4. The most frequent English translations are also provided. Ten sample sentences for each of these words (where the target word is to be substituted) were extracted from the Prague Dependency Treebank 2.0 [Hajičová, 1998], which contains markup of lemmatized form and thus allows various instances of use to be extracted. The annotation was done by a single native Czech speaker.

Due to the absence of a freely available parser providing GRs for Czech, it was only possible to run the n-gram and HMM modules in this experiment. Also, after initial experiments with using the Czech Wikipedia as training data, a further inflection problem came to light: should the candidate substitute be of a different gender to the original target word, the sentence stopped being grammatically correct when the candidate was substituted due to agreement. Thus a same animacy / type candidate would always be preferred. Consider the example:

> ... vstoupit do chrámu za účelem policejní <head>akce</head>

PoS	Average
Noun	7
Verb	8
Adjective	7

Table 5: *Average numbers of candidates for Czech*

Evaluation	Precision	Recall
Best	18.86	18.86
OOT	92.11	92.11

Table 6: *Czech lexical substitution*

if the correct substitute for the word *akce*, *čin* is used, the sentence needs to change to:

> ... vstoupit do chrámu za účelem policejního činu

The test data, and the training data, therefore required lemmatization: in the absence of a freely available lemmatizer for Czech, the PDT was used for both training and testing (with the test sentences being withheld from training). Thus n-grams (for $n = 1, 2, 5$) were acquired from this data, and indexed as carried out for English.

The candidates for each word were acquired from the Czech online synonyms resource (`http://www.synonyma-online.cz`), but the candidates for target words were also augmented by semi-automatically extracted synonyms from Wikipedia. The average numbers of candidates are presented in Table 5, and the combined results for the Czech lexical sample are presented in Table 6.

8 Conclusion and future work

We have presented a modular lexical substitution system which incorporates a number of novel approaches to the task. The approaches were shown to have good performance on the English lexical substitution data, while also being highly portable to other, potentially very different, languages (with a very good performance on the Czech data). We highlight the importance of a comprehensive, yet not over-generated candidate set, an issue which we fell has not been addressed enough in the past.

8.1 Future work

The GR module did not deal with issues of sparsness – the motivation being that the other modules will fill in. However, an alternative method for future work could be in grouping GRs together in meaningful ways [Pereira et al., 1993].

The HMM implemented a 1^{st}-Order Forward-Backward Algorithm. This introduces certain limitations to the transition probability matrices. If our running example had been

> Brian is a bright and lively boy.

[10] Available from `http://cs.wiktionary.org/`

instead, the separation of *bright* and *boy* by the intervening words *and lively* would have the effect of neutralizing the impact of the determining word on the target word. In this case, the words that would have the greatest impact on *bright* would be *a* and *and*, neither of which would contribute a significant amount of information that could lead to a proper lexical substitution.

References

[Brants and Franz, 2006] Brants, T. and Franz, A. (2006). Web 1T 5-gram Version 1.

[Briscoe et al., 2002] Briscoe, E. J., Carroll, J., Graham, J., and Copestake, A. (2002). Relational evaluation schemes. In *Proceedings of the beyond PARSEVAL Workshop at the 3rd International Conference on Language Resources and Evaluation*, pages 4–8.

[Briscoe et al., 2006] Briscoe, E. J., Carroll, J., and Watson, R. (2006). The second release of the RASP system. In *Proceedings of the COLING/ACL 2006, Interactive Poster Session*.

[Dagan et al., 2006] Dagan, I., Glickman, O., Gliozzo, A., Marmorshtein, E., and Strapparava, C. (2006). Direct word sense matching for lexical substitution. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 449–456.

[Elworthy, 1994] Elworthy, D. (1994). Does Baum-Welch re-estimation help taggers? In *Proceedings of the 4th Conference on Applied NLP*, pages 53–58.

[Graff, 2003] Graff, D. (2003). English gigaword. Technical report, Linguistic Data Consortium.

[Hajičová, 1998] Hajičová, E. (1998). Prague dependency treebank: From analytic to tectogrammatical annotations. In *Proceedings of 2nd TST*, pages 45–50.

[Hassan et al., 2007] Hassan, S., Csomai, A., Banea, C., and Mihalcea, R. (2007). UNT: SubFinder: combining knowledge sources for automatic lexical substitution. In *Proceedings of the 4th International Workshop on the Semantic Evaluations*.

[McCarthy, 2002] McCarthy, D. (2002). Lexical substitution as a task for WSD evaluation. In *Proceedings of the Workshop on Word Sense Disambiguation: Recent Successes and Future Directions*, pages 109–115.

[McCarthy and Navigli, 2007] McCarthy, D. and Navigli, R. (2007). Task 10: English lexical substitution task. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, pages 48–53.

[Miller et al., 1990] Miller, G., Beckwith, R., Felbaum, C., Gross, D., and Miller, K. (1990). Introduction to WordNet: An on-line lexical database. *Journal of Lexicography*, 3(4):235–244.

[Pereira et al., 1993] Pereira, F., Tishby, F., and Lee, L. (1993). Distributional clustering of English words. In *Proceedings of the Association for Computational Linguistics*, pages 183–190.

[Preiss and Yarowsky, 2002] Preiss, J. and Yarowsky, D., editors (2002). *Proceedings of SENSEVAL-2: Second International Workshop on Evaluating Word Sense Disambiguating Systems*.

[Roark and Sproat, 2007] Roark, B. and Sproat, R. W. (2007). *Computational Approaches to Morphology and Syntax*. Oxford University Press.

[Sinha et al., 2009] Sinha, R., McCarthy, D., and Mihalcea, R. (2009). Semeval-2010 task 2: Cross-lingual lexical substitution. In *Proceedings of the NAACL-HLT 2009 Workshop: SEW-2009 - Semantic Evaluations*.

[Yuret, 2007] Yuret, D. (2007). KU: Word sense disambiguation by substitution. In *Workshop of SemEval*.

Evidence-Based Word Alignment

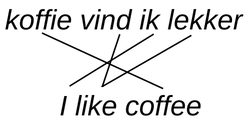
Jörg Tiedemann
Alpha-Informatica, Rijksuniversiteit Groningen,
Groningen, The Netherland,
Department of Linguistics and Philology
Uppsala University, Uppsala/Sweden
j.tiedemann@rug.nl

Abstract

In this paper we describe word alignment experiments using an approach based on a disjunctive combination of alignment evidence. A wide range of statistical, orthographic and positional clues can be combined in this way. Their weights can easily be learned from small amounts of hand-aligned training data. We can show that this "evidence-based" approach can be used to improve the baseline of statistical alignment and also outperforms a discriminative approach based on a maximum entropy classifier.

1 Introduction

Automatic word alignment has received a lot of attention mainly due to the intensive research on statistical machine translation. However, parallel corpora and word alignment are not only useful in that field but may be applied to various tasks such as computer aided language learning (see for example [15]) and bilingual terminology extraction (for example [8, 10]). The automatic alignment of corresponding words in translated sentences is a challenging task even for small translation units as the following Dutch-English example tries to illustrate.



Word alignment approaches have to consider crossing links and multiple links per word in both directions. Discontinuous units may also be aligned to corresponding parts in the other language as shown in the example above (*vind...lekker - like*). Various other issues due to translation divergency make word alignment a much more challenging task than, for instance, sentence alignment. Generative statistical models for word alignment usually have problems with non-monotonic alignments and many-to-many links. In the literature several attempts are described in which additional features are integrated besides the distribution of surface words to overcome these difficulties. In recent years various discriminative approaches have been proposed for this task [18, 9, 13, 14, 11, 1]. They require word-aligned training data for estimating model parameters in contrast to the traditional IBM alignment models that work on raw parallel (sentence aligned) corpora [2, 16]. However, previous studies have shown that only a small number of training examples (around 100 word-aligned sentence pairs) are sufficient to train discriminative models that outperform the traditional generative models.

In this paper we present another supervised alignment approach based on association clues trained on small amounts of word-aligned data. This approach differs from previous discriminative ones in the way the evidence for alignment is combined as we will explain in the following section.

2 Evidence-based alignment

The evidence-based alignment approach is based on the techniques proposed by [19]. This approach applies the notion of link evidence derived from word alignment clues. An *alignment clue* $C(r_k|s_i,t_j)$ is used as a probabilistic score indicating a (positive) relation r_k between two items s_i, t_j in their contexts. *Link evidence* $E(a, r_k|s_i,t_j)$ is then defined as the product of this score and the likelihood of establishing a link given the relation indicated by that clue:

$$E(a, r_k|s_i, t_j) = C(r_k|s_i, t_j)P(a|r_k)$$

Various types of alignment clues can be found in parallel data. Association scores and similarity measures can be used to assign their values. For example, the relation of "cognateness" may be indicated by string similarity measures. Translational equivalence relations can be indicated by co-occurrence measures. For the estimation of these scores, no word-aligned training data is required. However, for the estimation of the likelihood values we need training data as we will explain below. They can be seen as weights that correspond to the quality of clues in predicting links properly. Note that we can also use binary clues. Their influence on alignment decisions is determined by the alignment likelihood values only.

So far, this model is not so much different from previous discriminative alignment approaches in which weighted features are used in a classification approach (see, for example, [18], [13]). However, we use our weighted features as individual pieces of evidence that are combined in a disjunctive way, i.e. the overall alignment evidence for two given items is defined as the union of individual evidence scores:

$$E(a|s_i, t_j) = E(a, r_1 \vee r_2 \vee .. \vee r_k|s_i, t_j)$$

Note that alignment clues are not mutually exclusive and, therefore, we need to subtract the overlapping parts when computing the union. Using the addition rule of probabilities we obtain, for example, for two clues:

$$\begin{aligned} E(a, r_1 \vee r_2|s_i, t_j) &= E(a, r_1|s_i, t_j) + E(a, r_2|s_i, t_j) - \\ &\quad E(a, r_1 \wedge r_2|s_i, t_j) \end{aligned}$$

Hence we combine individual pieces of evidence in a non-linear way. Figure 1 tries to illustrate such a combination for two given cases.

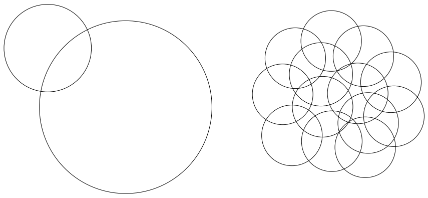


Fig. 1: *Combining alignment evidence. The size of the circles refers to the strength of the evidence given.*

The intuition behind this way of combining features is to give stronger pieces of evidence a larger influence on alignment decisions. As illustrated in figure 1 strong evidence is hard to overrule even by many other weaker clues. A few solid clues are sufficient just like a reliable witness in a murder case overrules all kinds of other weaker pieces of evidence indicating a different scenario. A consequence of our model is that alignment evidence with a value of 1.0 can not be outranked by any other combination of evidence. However, this is not as strange as it sounds if we consider that evidence giving 100% certainty should always be trusted. These cases should be very exceptional, though.

One difficulty arises in our approach: We need to estimate the overlapping parts of our collected evidence. For simplicity, we assume that all relation types are independent of each other (but not mutually exclusive) and, therefore, we can define the joint probability score of the overlapping part as $E(a, r_1 \wedge r_2|s_i, t_j) = E(a, r_1|s_i, t_j)E(a, r_2|s_i, t_j)$. The combination of independent evidence is illustrated in figure 2.

Altogether this model is similar to noisy OR-gates frequently used in belief networks in which causes are modeled to act independently of others to produce a determined effect [17]. Certainly, the independence assumption is violated in most cases. However, we will see in our experiments that this simplification still works well for alignment purposes. Note, that complex features can easily be constructed in order to reduce the impact of this violation on alignment performance.

2.1 Parameter estimation

As we have said earlier, the only parameters that need to be estimated from word-aligned training data are the alignment likelihoods used as weights for individual clues. Due to our independence assumption, we can do this by evaluating each individual clue on the training data. For this, we need to find out to what extent the indicated relations can be used to establish links in our data. Hence, we use each observed clue as a binary classifier and simply count the number of correctly predicted links using that clue (as usual a value above 0.5 is used to predict a positive example). This means that we use the precision of each individual clue on some training data to estimate alignment likelihoods. Intuitively, this seems to fit our approach in which we prefer high precision features as described earlier.

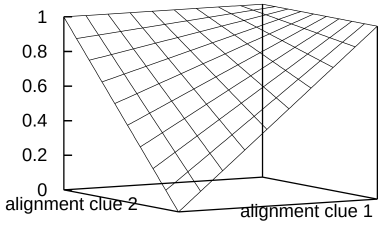


Fig. 2: *The combination of two independent alignment clues.*

Thus, training is extremely simple. The most expensive computation is actually the extraction of features used as alignment clues (see section 3.1 for details). The overhead of training is tiny and can be done in linear time. Note that this model only covers the classification of individual items. For the actual word alignment we need to apply a search algorithm that optimizes the alignment of all words according to the evidence found for individual pairs. This will briefly be discussed in the following section.

2.2 Link dependencies & alignment search

The problem of alignment search has been discussed in related studies on discriminative word alignment. The problem is that the dependency between links has to be considered when creating word alignments. Several approaches have been proposed that either include link dependencies directly in the underlying model [14, 1] or that include contextual features that implicitly add these dependencies [18]. Depending on the model optimal alignments can be found [18, 9, 1] or greedy search heuristics are applied [11, 14].

We will use the second approach and model link dependencies in terms of contextual features. We believe that this gives us more flexibility when defining contextual dependencies and also keeps the model very simple with regards to training. For the alignment search problem we could still apply a model that allows optimal decoding, for example, the approach proposed in [18]. However, we will stick to a simple greedy search heuristics, similar to the "refined" heuristics defined in [16], that is known to produce good results for example for the symmetrization of directional statistical word alignment. The advantages of this approach is that it is fast and easy to apply, it allows n:m alignments, and it makes our results comparable to the statistical alignments that include symmetrization.

3 Experiments

For our experiments we will use well-known data sets that have been used before for word alignment experiments. Most related work on supervised alignment models reports results on the French-English data set from the shared task at WPT03 [12] derived from the parallel Canadian Hansards corpus. This data set caused a lot of discussion especially because of the flaws in evaluation measures used for word alignment experiments [5]. Therefore, we will apply this set for training purposes only (447 aligned sentences with 4,038 sure (S) links and 13,400 (P) possible links) and stick to another set for evaluation [4]. This set includes English-French word alignment data for 100 sentences from the Europarl corpus [6] with a much smaller number of possible links (437 compared to 1,009 sure links) which hopefully leads to more reliable results.

Some of the alignment clues require large parallel corpora for estimating reliable feature values (for example co-occurrence measures). For training we use the Canadian Hansards as provided for the WPT03 workshop and for evaluation these values are taken from the Europarl corpus.

For evaluation we use the standard measures used in related research:

$$\begin{aligned}
Prec(A,P) &= \frac{|P \cap A|}{|A|} \\
Rec(A,S) &= \frac{|S \cap A|}{|S|} \\
AER(A,P,S) &= 1 - \frac{|P \cap A| + |S \cap A|}{|S| + |A|} \\
F(A,P,S,\alpha) &= 1/\left(\frac{\alpha}{Prec(A,P)} + \frac{(1-\alpha)}{Rec(A,S)}\right)
\end{aligned}$$

For the F-measure we give balanced values and also unbalanced F-values with $\alpha = 0.4$. The latter is supposed to show a better correlation with BLEU scores. However, we did not perform any tests with statistical MT using our alignment techniques to verify this for the data we have used.

For comparison we use the IBM model 4 alignments and the intersection and grow-diag-final-and symmetrizaton heuristics as implemented in the Moses toolkit [7]. We also compare our results with a discriminative alignment approach using the same alignment search algorithm, the same features and a global maximum entropy classifier [3] trained on the same training data (using default settings of the megam toolkit).

3.1 Alignment features

A wide variety of features can be used to collect alignment evidence. We use, among others, similar features as described in [18]. In particular, we use the Dice coefficient for measuring co-occurrence, the longest common subsequence ratio (LCSR) for string similarity, and other orthographic features such as identical string matching, prefix matching and suffix matching. We use the positional distance measures as described in [18] but turn them into similarity measures. We also model contextual dependencies by including Dice values for the next and the previous words. We use rank similarity derived from word type frequency tables and we use POS labels for the current words and their contexts. Furthermore, we also use evidence derived from the IBM models for statistical alignment. We use lexical probabilities, the directional alignment predictions of Model 4 and the links from the intersection heuristics of Model 4 alignments (produced by Moses/GIZA++; henceforth referred to as *Moses features*). As expected, these features are very powerful as we will see in our experimental results. A small sample from a feature file extracted from a sentence aligned parallel corpus is shown in figure 3.

```
possim 1 mosessrc2trg 1 mosestrg2src 1 pos_NN_VER:pper 1
possim 0.75 pos_NN_PRP 1 lcsr 0.05
possim 0.5 pos_NN_DET:ART 1
possim 0.25 pos_NN_NOM 1 lcsr 0.0714285714285714
possim 0.75 pos_IN_VER:pper 1
possim 1 mosessrc2trg 1 mosestrg2src 1 pos_IN_PRP 1
possim 0.75 pos_IN_DET:ART 1
possim 0.5 lcsr 0.142857142857143 pos_IN_NOM 1
possim 0.5 pos_DT_VER:pper 1 lcsr 0.142857142857143
....
```

Fig. 3: *A short example of link features extracted for each possible word combination in aligned sentences.* possim = *relative position similarity,* lcsr = *string similarity measure,* pos_* = *POS label pairs*

As we can see, some features are in fact binary (as discussed earlier) even though we use them in the same way as the real-valued features. For example, statistical alignment features derived from GIZA++/Moses (mosessrc2trg, mosestrg2src) are set to 1 if the corresponding word pair has been linked in the statistical Viterbi alignment. Other feature types are used as templates and will be instantiated by various values. For example, the POS label feature template adds a feature to each word pair made out of the labels attached to the corresponding words. Again, these features are used as binary flags as we can see in the example in figure 3.

Note that complex features can easily be created. We consider several combinations, for example the product of Dice scores and positional similarity scores. Contextual features can also be combined with any other feature. Complex features are especially useful in cases where the independence assumption is heavily violated. They are also useful to improve linear classification in cases where the correlation between certain features is non-linear.

3.2 Results

Our results are summarized in table 1.

As we can see, we cannot outperform the strong baselines without the features derived from statistical word

baselines	Rec	Prec	$F_{0.5}$	$F_{0.4}$	AER
intersection	72.1	**95.2**	82.0	79.8	17.5
grow-diag-final	**84.5**	78.7	81.5	82.1	18.8
best setting without Moses features					
MaxEnt	71.5	73.0	72.2	72.1	27.7
Clues	68.9	70.1	69.5	69.3	30.5
best setting with all features					
MaxEnt	82.3	84.4	83.3	83.1	16.6
Clues	82.6	85.4	**84.0**	**83.7**	**15.9**

Table 1: *Overview of results: Statistical word alignment derived from GIZA++/Moses (intersection/grow-diag-final), discriminative word alignment using a maximum entropy classifier (MaxEnt), and the evidence-based alignment (Clues).*

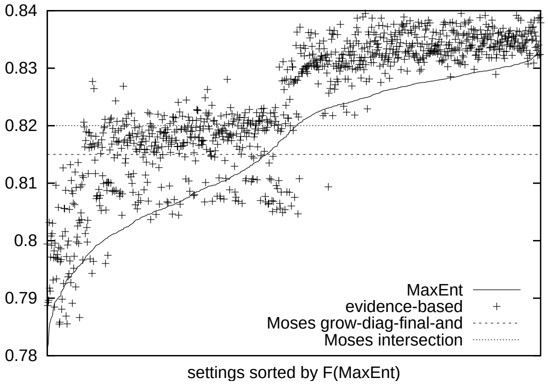


Fig. 4: *A comparison of $F_{0.5}$ scores obtained with settings that include statistical word alignment features.*

alignment. However, adding these features makes it possible to improve alignment results according to AER and F-scores. We can also observe that the MaxEnt classifier is better in handling the dependencies between non-Moses features. The scores are in general slightly above the corresponding clue-based scores. However, including the strong Moses features, our approach outperforms the maximum entropy classifier and yields the overall best result. As expected, our approach seems to handle the combination of strong evidence and weak clues well. It learns to trust these strong clues and still includes additional evidence from other alignment clues. Figure 4 illustrates this by plotting the results ($F_{0.5}$ scores) for settings that include Moses features for both, the MaxEnt classifier approach and the evidence-based approach. The settings are sorted by the F-scores obtained by the MaxEnt classifier approach (solid line) along the x-axis. Corresponding F-scores obtained by the evidence-based approach using the same feature set and alignment search algorithm are plotted as points in the graph. As we can see in most cases, our simple evidence-based approach yields similar or better results than the MaxEnt approach. We can also see that both discriminative approaches improve the baseline scores obtained by the generative statistical word alignment after symmetrization (dashed and dotted lines in the graph).

The best result is obtained with the following features: Dice for the current word pair and the previous one, positional similarity, POS labels, rank similarity, lexical probabilities and link predictions of the two IBM 4 Viterbi alignments. Surprisingly, the orthographic features (LCSR etc) do not perform well at all. Some example weights learned from the training data using the alignment prediction precision are shown in table 2.

feature	**prediction precision**
dice	0.8120830711139080
prevdice	0.8228682170542640
possim	0.2656349270994540
ranksim	0.4259383603034980
lexe2f	0.9634980007738940
lexf2e	0.9348459880846750
lexe2f*lexf2e	0.9900965585540980
mosessrc2trg	0.9601313748745550
mosestrg2src	0.9514683153013910
pos_VBZ_VER:pres	0.7395577395577400
pos_NNS_NOM	0.5319049836981840
pos_)_PUN	0.7142857142857140
pos_VV_ADJ	0.0393013100436681
pos_NNS_VER:pper	0.0593607305936073

Table 2: *Examples of weights learned from prediction precision of individual clues.*

We can see that features derived from statistical word alignment have a high precision and, therefore, the evidence-based alignment approach trusts them a lot. This includes the lexical probabilities taken from the translation model as estimated by Moses. Especially their product is very accurate which is maybe not so surprising considering that this score will be very low for most word pairs and, therefore, only a few links will be predicted by this feature. Co-occurrence measures score also very high. Note that the Dice score of the previous words (*prevdice*) also seems to be very useful for alignment prediction. On the other hand, positional similarity (*possim*) is a rather weak clue according to the precision computed. However, it is still very useful to make alignment decisions in cases where other evidence is missing or not discriminative enough. Frequency rank similarity (*ranksim*) is also surprisingly strong. This is probably due to the similarity between English and French especially in terms of inflectional complexity. Finally, we can see examples of the weights estimated for binary features such as POS label pairs. Here, we use a threshold of a minimum of five occurrences to obtain reliable estimates. We can see that some of them are very useful in predicting links whereas others are very low. Probably, negative clues could be useful as well, for example, using POS labels that indicate a preference for not linking the corresponding items. However, for this the alignment model has to be adjusted to account for such clues as well.

Finally, we also include the plot of alignment error rates for settings that include Moses features (see figure 5).

We can see that the curve follows the same trend as we have seen for the F-scores in figure 4. Most of the evidence-based alignment results are below the corresponding runs with a linear classifier. Again, we also outperform the generative alignment approach,

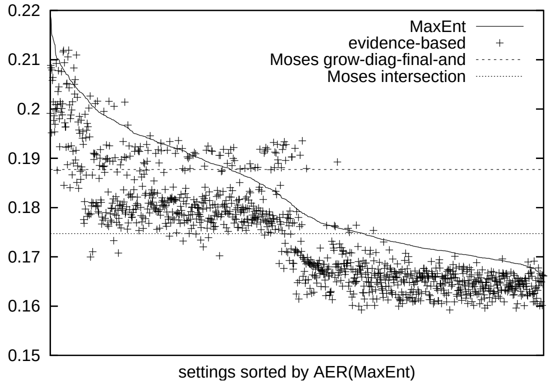


Fig. 5: *A comparison of AER scores obtained with settings that include statistical word alignment features.*

however, only when using features derived from these alignments.

4 Conclusions

In this paper we describe our experiments with evidence-based word alignment. Features (alignment clues) in this approach are combined in a non-linear way in contrast to related discriminative word alignment approaches that usually apply linear classification techniques in the underlying model. We have shown that this kind of combination can be beneficial when comparing to a straightforward linear classification approach especially when high precision features are applied. Another advantage is the simplicity of training feature weights using individual link prediction precision. However, this requires the assumption that each feature can be used as an independent base classifier. This assumption is often violated which can be seen in the degrading performance of the evidence-based approach when applying it in connection with weaker clues. However, the approach seems to work well in terms of picking up strong clues and learns to trust them appropriately. It remains to be investigated to what extend this approach can be used to improve subsequent applications such as machine translation or bilingual terminology extraction. Furthermore, it should be embedded in a proper structural prediction framework in which output space dependencies (between predicted links in a sentence pair) are modeled explicitly. This will boost the performance even further as it has been shown for other discriminative word alignment approaches.

References

[1] P. Blunsom and T. Cohn. Discriminative word alignment with conditional random fields. In *Proceedings of ACL*, Sydney, Australia, 2006.

[2] P. F. Brown, S. A. D. Pietra, V. J. D. Pietra, and R. L. Mercer. The Mathematics of Statistcal Machine Translation: Parameter Estimation. *Computational Linguistics*, 19(2):263–311, June 1993.

[3] H. Daumé III. Notes on CG and LM-BFGS optimization of logistic regression. Paper available at `http://pub.hal3.name#daume04cg-bfgs`, implementation available at `http://hal3.name/megam/`, 2004.

[4] J. de Almeida Varelas Graa, J. P. Pardal, L. Coheur, and D. A. Caseiro. Building a golden collection of parallel multi-language word alignment. In *Proceedings of LREC*, 2008.

[5] A. Fraser and D. Marcu. Measuring word alignment quality for statistical machine translation. *Computational Linguistics*, 33(3):293–303, 2007.

[6] P. Koehn. Europarl: A parallel corpus for statistical machine translation. In *Proceedings of MT Summit*, 2005.

[7] P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, C. Dyer, O. Bojar, A. Constantin, and E. Herbst. Moses: Open source toolkit for statistical machine translation. In *Proceedings of ACL*, Prague, Czech Republic, 2007.

[8] J. Kupiec. An algorithm for finding noun phrase correspondences in bilingual corpora. In *Proceedings of the 31st annual meeting on Association for Computational Linguistics*, pages 17–22, Morristown, NJ, USA, 1993. Association for Computational Linguistics.

[9] S. Lacoste-Julien, B. Taskar, D. Klein, and M. Jordan. Word alignment via quadratic assignment. In *Proceedings of HLT-NAACL*, New York, 2006.

[10] E. Lefever, L. Macken, and V. Hoste. Language-independent bilingual terminology extraction from a multilingual parallel corpus. In *EACL*, pages 496–504. The Association for Computer Linguistics, 2009.

[11] Y. Liu, Q. Liu, and S. Lin. Log-linear models for word alignment. In *Proceedings of ACL*, Ann Arbor, Michigan, 2005.

[12] R. Mihalcea and T. Pedersen. An evaluation exercise for word alignment. In *HLT-NAACL 2003 Workshop: Building and Using Parallel Texts: Data Driven Machine Translation and Beyond*, 2003.

[13] R. C. Moore. A discriminative framework for bilingual word alignment. In *Proceedings of HLT/EMNLP*, Vancouver, Canada, 2005.

[14] R. C. Moore, W. tau Yih, and A. Bode. Improved discriminative bilingual word alignment. In *Proceedings of ACL*, 2006.

[15] J. Nerbonne. Parallel texts in computer-assisted language learning. In J. Veronis, editor, *Parallel Text Processing*, pages 354–369. Kluwer, Dordrecht and Boston, 2000.

[16] F. J. Och and H. Ney. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51, 2003.

[17] J. Pearl. Fusion, propagation and structuring in belief networks. *Artificial Intelligence*, 29(3):241–288, 1986.

[18] B. Taskar, S. Lacoste-Julien, and D. Klein. A discriminative matching approach to word alignment. In *Proceedings of HLT/EMNLP*, Vancouver, Canada, 2005.

[19] J. Tiedemann. Combining clues for word alignment. In *Proceedings of the 10th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 339–346, Budapest, Hungary, April 2003.

A Discriminative Approach to Tree Alignment

Jörg Tiedemann
Alpha-Informatica, Rijksuniversiteit Groningen,
Groningen, The Netherland,
Department of Linguistics and Philology
Uppsala University, Uppsala/Sweden
j.tiedemann@rug.nl

Gideon Kotzé
Alpha-Informatica
Rijksuniversiteit Groningen
Groningen, The Netherlands
g.j.kotze@rug.nl

Abstract

In this paper we propose a discriminative framework for automatic tree alignment. We use a rich feature set and a log-linear model trained on small amounts of hand-aligned training data. We include contextual features and link dependencies to improve the results even further. We achieve an overall F-score of almost 80% which is significantly better than other scores reported for this task.

1 Introduction

A parallel treebank consists of a collection of sentence pairs that have been grammatically tagged, syntactically annotated and aligned on sub-sentential level [12]. Large parallel treebanks are much sought after in present-day NLP applications but have been, until recently, only been built by hand and therefore tended to be small and expensive to create. Some areas of application for parallel treebanks are:

- knowledge source for transfer-rule induction
- training for data-driven machine translation
- reference for phrase-alignment
- knowledge source for corpus-based translation studies
- knowledge source for studies in contrastive linguistics

As for ourselves, we are interested in applying tree alignment in the context of a syntax-based machine translation (MT) approach. Since well-aligned treebanks will play a substantial role in our MT model, finding an optimal solution to the problem of tree alignment is very important. In the next section, we provide a brief background of recent findings on the topic before presenting our own approach thereafter.

2 Related Work

Most related work on tree alignment is done in the context of machine translation research. Several variants of syntax-based MT approaches have been proposed in recent years involving the alignment of syntactic structures. In general we can distinguish between tree-to-string (or vice versa) and tree-to-tree alignment approaches. [15] describe some recent attempts at sub-sentential alignment on the phrase level:

[9] use a stochastic inversion transduction grammar to parse a source sentence and use the output to build up a target language parse, while also inducing alignments. The latter are extracted and converted into translation templates. [16] use a method they call “bilingual chunking”, where the words of a tree pair are aligned and during the process, chunks are extracted by using the tree structure, after which the chunks are POS tagged. However, the original tree structures are lost in the process. [3] proposes a method which alters the structure of non-isomorphic phrase-structure trees to impose isomorphism in order to align the trees using a stochastic tree substitution grammar (STSG). This, however, restricts its portability to other domains, according to [15]. [4] present a rule-based aligner which makes use of previously determined word alignments. However, the algorithm performed poorly when applied to other language pairs [15].

According to [12] there are two general approaches to tree alignment: finding correspondences between phrases through parsing or chunking (eg. [13]), or deriving phrase alignment through previous word alignment, the latter of which they have adopted themselves, where the best configuration yields an $F_{0.5}$ score of 65.84%. Lately, in [15] a better and faster method was proposed using 1:1 word alignment probabilities and parse trees. Trees can also be constructed automatically in the absence of a parser. In a more recent update [17] taking all links into account, a highest precision of 61,79% and a highest recall of 78,49% in the tree alignment task were achieved. Zhechev and Way define a set of principles (2008:1106) to be followed in their alignment method:

- independence with respect to language pair and constituent labelling schema
- preservation of the given tree structures
- minimal external resources required
- word-level alignments are guided by links higher up the trees, which provide more context information

NLP Methods and Corpora in Translation, Lexicography, and Language Learning 2009 – Borovets, Bulgaria, pages 33–39

In addition, the authors quote [6] in defining a set of well-formedness criteria and explaining that this should result in producing “enough information to allow the inference of complex translational patterns from a parallel treebank, including some idiosyncratic translational divergences” (2008:1106): (i) A node in a tree may only be linked once. (ii) Descendants/ancestors of a source linked node may only be linked to descendants/ancestors of its target linked counterpart. In short the alignment algorithm consists of the following steps:

- Each source node s can link to any target node t and vice versa. Initially all these links are hypothesized.

- Every one of these hypotheses is assigned a score $\gamma(\langle s,t\rangle) = \alpha(s_l|t_l)\alpha(t_l|s_l)\alpha(\overline{s_l}|\overline{t_l})\alpha(\overline{t_l}|\overline{s_l})$ based on the word-alignment probabilities of the words that are governed by the current nodes (s_l and t_l), as well as the probabilities of the words outside the span ($\overline{s_l}$ and $\overline{t_l}$):

$$\alpha(x|y) = \prod_{i=1}^{|x|} \frac{1}{|y|} \sum_{j=1}^{|y|} P(x_i|y_j)$$

- Using these scores a set of links is selected applying a greedy search algorithm that also satisfies the well-formedness criteria.

Since the system described here has produced promising results and has been released publicly, we have decided to use it as a baseline, as well as a source of input material, upon which we hope to improve. For this we apply a discriminative alignment approach that is presented below.

3 Tree Alignment

In our approach we only look at tree-to-tree alignment using phrase-structure trees on both sides. In the following we first introduce the general link prediction model. Thereafter, we give a detailed description of features applied in our experiments and the alignment search strategy applied.

3.1 Link Prediction

Similar to related work on discriminative word alignment we base our model on association features extracted for each possible alignment candidate. For tree alignment, each pair of nodes $\langle s_i, t_j\rangle$ from the source and the target language parse tree is considered and a score x_{ij} is computed that represents the degree to which both nodes should be aligned according to their features $f_k(s_i, t_j, a_{ij})$ and corresponding weights λ_k derived from training data. In our approach we use conditional likelihood using a log-linear model for estimating these values:

$$P(a_{ij}|s_i,t_j) = \frac{1}{Z(s_i,t_j)} exp\left(\sum_k \lambda_k f_k(s_i,t_j,a_{ij})\right)$$

Here, the mapping of data points to features is user provided (see section 3.2) and the corresponding weights are learned from aligned training data. We simplify the problem by predicting individual alignment points for each candidate pair instead of aiming at structured approaches. Hence, we can train our conditional model as a standard binary classification problem. Note that contextual features can easily be integrated even though first-order dependencies on surrounding alignments are not explicitly part of the model. More details will be given below in sections 3.2.5 and 3.2.7.

In our experiments we will use a maximum entropy classifier using the log-linear model as stated above. One of the advantages of maximum entropy classifiers is the flexibility of choosing features. No independence assumptions have to be made and state-of-the art toolboxes are available with efficient learning strategies. Here, we apply the freely available toolbox Megam [1] and train a global binary classification model predicting links between given node pairs.

3.2 Alignment Features

The selection of appropriate features for classification is crucial in our approach. The input to the tree aligner is sentence aligned parse tree pairs from which various features can be extracted. Another important source is word alignment and information derived from statistical word alignment models. In the following we describe the different feature types that we apply.

3.2.1 Lexical Equivalence Features

Lexical probabilities are used in unsupervised tree alignment approaches as explained earlier in section 2. We will also use the same *inside/outside scores* defined in [17] as our basic features as they have proven to be useful for tree alignment. However, we define additional features and feature combinations derived from automatic word alignment in order to enrich the alignment model. First of all, we use inside and outside scores as individual features besides their product. We also use individual $\alpha(x|y)$ scores as separate features. Furthermore, we define a variant of inside and outside scores using a slightly modified definition of the equivalence score α:

$$\alpha_{max}(x|y) = \prod_{i=1}^{|x|} max_j P(x_i|y_j)$$

We believe that this definition better reflects the relations between words in sentences than the original definition in which an average of the conditional lexical probabilities is used. We assume that most words are linked to only one target word (hence we look for the maximum) whereas averaging over all combinations punishes long phrases too much.

Another variant can be defined by replacing the product above by a sum and taking the average per source token of this score:

$$\alpha_{avgmax}(x|y) = \frac{1}{|x|} \sum_{i=1}^{|x|} max_j P(x_i|y_j)$$

In this way, the impact of source tokens for which no links can be found with any of the target language tokens is reduced. In the original formulation scores will be zero if there is such a token even if all the other ones show a strong relation. This is avoided with the new definition. Using the modified scores the same combinations of inside and outside scores can be defined as explained earlier.

3.2.2 Word Alignment Features

Important features can be derived from the Viterbi alignments produced by statistical word alignment. Apart from the lexical probabilities used in the previous section, the actual statistical word alignment takes additional parameters into account, for example, positional similarity and first-order dependencies between links. Using Viterbi alignments we can implicitly take advantage of these additional parameters. We define *word alignment features* as the proportion of consistent links $cons(l_{xy}, s_i, t_j)$ among all links l_{xy} involving either source (s_x) or target language words (t_y) dominated by the current tree nodes (which we will call relevant links $relev(l_{w_s,w_t}, s_i, t_j)$). Consistent links are links between words which are both dominated by the nodes under consideration (dominance is denoted as $s_x \leq s_i$).

$$
\begin{aligned}
align(s_i, t_j) &= \frac{\sum_{l_{xy}} cons(l_{xy}, s_i, t_j)}{\sum_{l_{xy}} relev(l_{xy}, s_i, t_j)} \\
cons(l_{xy}, s_i, t_j) &= \begin{cases} 1 & \text{if } s_x \leq s_i \wedge t_y \leq t_j \\ 0 & \text{otherwise} \end{cases} \\
relev(l_{xy}, s_i, t_j) &= \begin{cases} 1 & \text{if } s_x \leq s_i \vee t_y \leq t_j \\ 0 & \text{otherwise} \end{cases}
\end{aligned}
$$

Note that the definition above is not restricted to word alignment. Other types of existing links between nodes dominated by the current subtree pair could be used in the same way. However, using the results of automatic word alignment we can compute these features from the links between terminal nodes. We can use various types of automatic word alignments. In our experiments we apply the Viterbi alignments produced by Giza++ [11] using the IBM 4 model in both directions, the union of these links and the intersection. For the latter we use Moses [8] which is also used for the estimation of lexical probabilities applied for lexical features described in the previous section.

Yet another feature derived from word alignment can be used to improve the alignment of terminal nodes. This feature is set to one if and only if both nodes are terminal nodes and are linked in the underlying word alignment.

3.2.3 Sub-tree Features

Features can also be derived directly from the parse trees. Similar to statistical word alignment, positional similarity can be used to make alignment decisions. However, in tree alignment we look at hierarchical structures and therefore a second dimension has to be considered. Therefore, we define the following two *tree position features*: tree-level similarity (tls) and tree span similarity (tss). For the former we use the distances $d(s_i, s_{root})$, $d(t_i, t_{root})$ from the current candidate nodes to the root nodes of source and target language tree, respectively. Furthermore, we use the size of a tree (defined as the maximum distance of any terminal node in the tree to the root) to compute the relative tree level of a given node. Finally, the tree-level similarity is then defined as the one minus the absolute value of the difference between relative tree levels:

$$
tll(s_i, t_j) = 1- \quad abs \quad \left(\frac{d(s_i, s_{root})}{max_x d(s_x, s_{root})} - \frac{d(t_i, t_{root})}{max_x d(t_x, t_{root})} \right)
$$

The second measure, the source span similarity is defined as one minus the absolute value of the difference between the relative positions of the subtrees under consideration. The relative positions are computed from the subtree spans using the surface positions $pos(s_x)$, $pos(t_y)$ of words dominated by the root nodes of these subtrees divided by the lengths of source and target language sentence, respectively.

$$
tss(s_i, t_j) = 1- \quad abs \quad \left(\frac{min\ pos(s_x) + max\ pos(s_x)}{2 * length(S)} - \frac{min\ pos(t_y) + max\ pos(t_y)}{2 * length(T)} \right)
$$

Another tree feature that we will use refers to the number of terminal nodes dominated by the candidate nodes. We define the ratio of leaf nodes as follows:

$$
leafratio(s_i, t_j) = \frac{min(|s_x \leq s_i|, |t_y \leq t_i|)}{max(|s_x \leq s_i|, |t_y \leq t_i|)}
$$

The intuition behind this feature is the assumption that nodes dominating a large number of terminal nodes are less likely to be aligned to nodes dominating a small number of terminal nodes.

3.2.4 Annotation Features

Finally, we can also define binary features describing the presence of certain annotations. For example, we can define pairs of category labels (for non-terminal nodes) or part-of-speech labels (for terminal nodes) as binary features. Each observed combination of labels in the training data is then used as a possible feature and the weights learned in training will determine if they influence alignment decisions in a positive or negative way.

3.2.5 Contextual Features

Using the tree structure, we can extract similar features from the context of candidate nodes. In this way, first order dependencies can implicitly be included in the model. For example, including inside/outside scores from the parent nodes partially includes the likelihood of these nodes being aligned. This may then increase the likelihood of the current nodes to be aligned as well. Contextual features can be very flexible and may also show a negative correlation. For example, a positive feature extracted for the current source language node together with the parent node of the target language node may decrease the likelihood of the alignment between the two current nodes.

In our implementation we allow various kinds of contextual features. Any feature as defined in the previous section can also be extracted from the parent nodes (either both parents or just one of them together with the current node in the other language). Furthermore, we also allow to extract these features from sister nodes (nodes with the same parent) and child nodes. For these nodes we only use the feature that provides the largest value.

We also allow multiple steps in our feature definition, allowing for, for example, grandparent features to be included. Naturally, contextual features are only extracted if the specified contexts actually exist (i.e. if there is a grandparent node).

3.2.6 Complex Features

A drawback of log-linear models is that features are combined in a linear way only. However, the correlation between some features might be non-linear and a typical strategy to reduce the negative effects of such interactions is to combine features and to build complex ones[1]. We define two operations for the combination of features:

Multiplication: The values of two or more features are multiplied with each other. This is only used for non-binary features.

Concatenation: Binary *feature types* can be combined with other features in the following way: Each *instantiation* of that type (for example a category label pair) is concatenated with the name of the other feature and the average of feature values is used.

We do not attempt to perform an exhaustive search among all possible combinations. Many of them will fail anyway due to data sparseness. However, complex features provide valuable contributions as we will see in our experiments.

3.2.7 Link Dependency Features

The last category of features refers to link dependency features. As we explained earlier, first-order dependencies are not explicitly modeled in our classification-based approach. However, features may include such dependencies, for example link information of connected nodes. Such features can easily be included in training where the complete link information is given. However, we have to adjust the link strategy in order to use these features in the alignment phase.

In our experiments, we define first-order features in the following way. The *children_links* feature is the number of links between child nodes of the current node pair normalized by the maximum of the number of source language children and the number of target language children. Similarly, the *subtree_links* feature is the number of links between nodes in the entire subtrees dominated by the current nodes. This score is then normalized by the larger number of nodes in either the source subtree or the target subtree.

In the alignment phase corresponding link information is not available. However, from the classifier we obtain probabilities for creating links between given nodes. We will use these conditional probabilities as soft counts for computing the first-order features as defined above, i.e. we sum over the link probabilities and normalize again in the same way. Our features are defined in terms of descendents of the current nodes. Hence, we perform classification in a bottom-up breadth-first fashion starting at the terminal nodes that do not include any children.

We also tried a top-down classification strategy together with parent link dependencies. However, this did not give us any significant improvements. Therefore, we will not report these results here.

3.3 Alignment Search

Our tree alignment approach is based on a global binary classifier. This means that we actually classify individual node pairs even though we include contextual and first-order features as described above. Despite the fact that individual classification is possible in this way, the important notion of *alignment competition* is not explored in this way. That this is a strong drawback has already been pointed out in related research on word alignment [14]. However, similar to discriminative word alignment, competition can easily be integrated in the system by applying appropriate search strategies. Naturally, the best strategy would be to include competition explicitly in the alignment model and train parameters for a structural alignment approach. We will leave this for future research and concentrate our current work on feature selection in combination with simple greedy search heuristics. In particular, we will use a greedy best-first search similar to competitive linking used in early work on word alignment. One of the drawbacks in this technique is that it only allows one-to-one links. However, in tree alignment this is not necessarily a drawback and often even defined as a well-formedness criterion [15]. Another drawback is, of course, that we are not guaranteed to find the optimal solution. However, it should be rather straightforward to implement a graph-theoretic search approach as described by [14] defining tree alignment as a weighted bipartite graph matching problem. We will leave even this for future research.

Finally, we will also introduce additional constraints that may help to improve alignment accuracy. First of all, a threshold can be defined in order to stop the greedy link strategy if link probabilities obtained by the classifier are too low. Secondly, a number of well-formedness criteria can be added to avoid unusual link combinations. We will use the criteria as defined in [17], as already mentioned in section 2: De-

[1] Another possibility would be to switch to kernel-based methods and to apply, for example, support vector machines with non-linear kernels. This will be tested more thoroughly in future work. Our first experiments with SVMs were discouraging mainly due to the largely increased time necessary for training.

scendents/ancestors of a source linked node may only be linked to descendents/ancestors of its target linked counterparts. Furthermore, we will use another constraint which is similar to the collapsing strategy of unary productions used by the same authors. However, we do not collapse trees at these points but we simply do not align nodes with single children. Note that this still allows links between terminal nodes as they do not have any children at all. Node type specific constraints can also be applied. For example, we may restrict links to be assigned to nodes of the same type only (non-terminals to non-terminals and terminals to terminals). We may also restrict ourselves to non-terminal nodes only. Note that these restrictions change the behavior of the unary-production constraint in the following way: If these restrictions are applied the unary-production constraint is relaxed in such a way that these nodes are only skipped if the one and only child is not a terminal node. This relaxation is necessary to include valuable links near the leafs that otherwise would be skipped.

Our implementation allows to switch on and off any of the constraints described above. Search heuristics can also easily be altered within the framework described above. In the following section we will describe experiments using various settings and models trained on a given treebank.

4 Experiments

We ran a number of experiments using a pre-aligned treebank and various settings including features as described above. In the following, we will first briefly describe the data used for training and testing. Thereafter evaluation measures are defined and results of our experiments are summarized.

4.1 Data

Aligned parallel treebanks are rare and, hence, training material for a supervised tree alignment approach is hard to find. However, a number of parallel treebank projects have been initiated recently and their data and tools become available. For our experiments, we will use the Smultron treebank [5] that includes two trilingual parallel treebanks in English, Swedish and German. The corpus contains the alignment of English-Swedish and German-Swedish phrase structure trees from the first two chapters of the novel "Sophie's World" by Jostein Gaarder and from economical texts taken from three different sources. We will use the English-Swedish treebank of Sophie's World which includes roughly 500 sentences per language. The first 100 aligned parse trees are used for training and the remaining part for testing. The alignment has been done manually using the Stockholm Tree Aligner [10] which we also intend to use later on when working on our own corpora and language pairs. The alignment includes *good* links and *fuzzy* links. We will use both but give them different weights in training (good alignments get three times the weight of fuzzy and negative examples). Altogether, there are 6,671 good links and 1,141 fuzzy links in the corpus.

4.2 Evaluation

For evaluation we use the standard measures of precision, recall and F-scores. Due to the distinction between good and fuzzy alignments we compute values similar to word alignment evaluation scores in which "sure" and "possible" links are considered:

$$\begin{aligned}
Prec(A,P) &= |P \cap A|/|A| \\
Rec(A,S) &= |S \cap A|/|S| \\
F(A,P,S,\alpha) &= 1/\left(\frac{\alpha}{Prec(A,P)} + \frac{(1-\alpha)}{Rec(A,S)}\right)
\end{aligned}$$

S refers here to the good alignments in the gold standard and P refers to the possible alignments which includes both, good and fuzzy. A are the links proposed by the system and α is used to define the balance between precision and recall in the F-score. We will only use a balanced F-score with $\alpha = 0.5$. We also omit alignment error rates due to the discussion about this measure in the word alignment literature. Note that the proportion of fuzzy links seems reasonable and we do not expect severe consequences on our evaluation as discussed in [2] for word alignment experiments with unbalanced gold standards.

4.3 Results

The selection of appropriate features is very important in our approach. We tested a number of feature sets and combinations in order to see the impact of features on alignment results. Table 1 summarizes our experiments with various sets. The upper part represents the performance of separate feature types on their own. The lower part shows results of combined feature types. Link dependency features are added in the right-hand side columns – either child link dependencies or dependencies on all subtree nodes.

As we can see in table 1, adding features consistently improves the scores even if their standalone performance is rather low. Especially the addition of label features improves the scores significantly. Contextual features are also very useful as we can see on the example of label features. Note, that we also use complex features such as combined inside/outside scores and alignment features. Also the concatenation of label features with alignment features is very successful.

For comparison we also ran the subtree aligner by [17] on the same data set. It yields a balanced F-score on our test set of 57.57% which is significantly lower than our best results. However, this comparison is not entirely fair as our training data is very small and the unsupervised subtree aligner relies on good estimates of lexical probabilities. Therefore, we also ran the aligner on our data with a lexical model extracted from a much larger data set. For this, we used the combination of the entire Swedish-English Europarl corpus [7] and the Smultron data. However, the scores improve only slightly to an F-score of 58.64%. The reason for this is probably that the Europarl data represents a very different type than the novel used in our test. However, it indicates the possibilities of discriminative tree alignment when trained on small amounts of aligned data.

	no link dependencies			+ child link dependencies			+ subtree link dependencies		
features	Prec	Rec	$F_{0.5}$	Prec	Rec	$F_{0.5}$	Prec	Rec	$F_{0.5}$
lexical	65.54	35.72	46.24	62.92	41.26	49.84	59.64	41.07	48.64
lexical$_{max}$	66.07	36.77	47.24	63.17	41.74	50.26	59.76	41.81	49.20
lexical$_{avgmax}$	63.76	43.04	51.39	60.95	41.96	49.70	60.92	41.94	49.68
tree	30.46	34.50	32.36	33.10	38.61	35.64	33.37	38.81	35.84
alignment	61.36	54.52	57.74	64.91	58.72	61.66	59.24	54.68	56.87
label	36.14	35.12	35.62	45.00	41.38	43.11	48.77	44.03	46.28
context-label	56.53	44.64	49.88	59.17	50.79	53.72	60.47	53.44	56.74
lexical$_{max}$ + tree	48.32	55.15	51.51	54.86	57.51	52.95	49.40	57.25	53.03
+ alignment	55.65	57.94	56.77	57.09	60.31	58.65	57.18	60.58	58.83
+ label	73.43	74.76	74.09	74.39	75.86	75.12	74.68	76.67	75.17
+ context-label	**76.65**	75.45	76.05	76.99	75.85	76.42	77.17	76.10	76.63
+ align-context	76.23	**77.43**	**76.83**	**77.07**	**78.16**	**77.61**	**78.12**	**78.42**	**78.27**

Table 1: *Results for different feature sets.*

Furthermore, we want to see the performance of our tree aligner on different node types. For this we computed separate evaluation scores for the different types using a run with all features (see table 2).

	Rec			Prec	$F_{0.5}$
type	*good*	*fuzzy*	*all*	*all*	*all*
non-terminals	**84.29**	**70.23**	**81.28**	78.04	**79.63**
terminals	75.08	59.11	73.80	**78.18**	75.93

Table 2: *Results for different node types (all features) including recall scores for different link types.*

From the table we can see that the aligner has more difficulties in finding links between terminal nodes than between non-terminals. This is especially true for fuzzy links. However, the precision is as high as for non-terminal nodes[2]. The reason for the drop in recall is probably due to the search algorithm which restricts our results to one-to-one links only. This constraint might be reasonable for non-terminal nodes but not for the alignment of words. A conclusion from this result is that we should either keep the external word alignment for establishing terminal links in our tree alignment or that we should use a separate model and search strategy for aligning terminal nodes.

Finally, we also want to look at the generality of our approach. A drawback of supervised methods is the risk of over-training especially if a rich feature set and small amounts of training data are used. Certainly, our approach is not language independent especially when label features are applied. However, we would like to know if the models learned can be applied to different text types without significant loss in performance. Therefore, we carried out an experiment training on one text type (novel or economy) and aligning the other one from the Smultron corpus. For reasons of fair comparison we also trained on the first 100 sentence pairs only but applied the model learned to the entire test corpus of the other type. Table 3 summarizes the results when applying the full-featured model in this way.

As we can see in the table, performance drops, especially in terms of recall. Precision is still comparable to the model trained on the same corpus (see line one in table 3). However, the drop is not dramatical and the models seem to capture enough general associations to make reasonable predictions. This is certainly encouraging especially considering the effort of human annotation necessary when preparing appropriate training data.

setting	Prec	Rec	$F_{0.5}$
train=novel, test=novel	78.12	78.42	78.27
train=novel, test=economy	77.39	73.50	75.39
train=economy, test=novel	76.66	74.62	75.62

Table 3: *Training on different text types*

5 Conclusions & Future Work

In this paper we describe a discriminative framework for automatic tree alignment. A log-linear model is learned from small amounts of pre-aligned training data. We use a rich set of features coming from the annotation and from automatic word alignment. We include contextual features and link dependency information for further improvements. Our model performs significantly better than previous methods on the same task and we believe that our results can be further improved in various ways. Some ideas for future work include the optimization of the search algorithm (using a graph-theoretic matching approach), the exploration of automatic methods for feature selection and combination (using, for example, a genetic algorithm) and a better integration of link dependencies (using a structural model instead of a single binary classifier). We will also look at additional features and the application of this approach to other data sets and language pairs. Finally, we will also investigate the impact of alignment quality on machine translation models based on parallel treebanks.

References

[1] H. Daumé III. Notes on CG and LM-BFGS optimization of logistic regression. Paper available at http://pub.hal3.name#daume04cg-bfgs, implementation available at http://hal3.name/megam/, August 2004.

[2] Note that the aligner does not assign link types and therefore, precision cannot be measured for different types.

[2] A. Fraser and D. Marcu. Measuring word alignment quality for statistical machine translation. *Computational Linguistics*, 33(3):293–303, 2007.

[3] D. Gildea. Loosely tree-based alignment for machine translation. In *Proceedings of the 41st Annual Meeting of the Association of Computational Linguistics (ACL-03)*, pages 80–87, Sapporo, Japan, 2003.

[4] D. Groves, M. Hearne, and A. Way. Robust sub-sentential alignment of phrase-structure trees. In *Proceedings of the 20th International Conference on Computational Linguistics (CoLing 2004)*, pages 1072–1078, Geneva, Switzerland, 2004.

[5] S. Gustafson-Čapková, Y. Samuelsson, and M. Volk. SMULTRON (version 1.0) - The Stockholm MULtilingual parallel TReebank. http://www.ling.su.se/dali/research/smultron/index.htm, 2007. An English-German-Swedish parallel Treebank with sub-sentential alignments.

[6] M. Hearne, J. Tinsley, V. Zhechev, and A. Way. Capturing translational divergences with a statistical tree-to-tree aligner. In *Proceedings of the 11th International Conference on Theoretical and Methodological Issues in Machine Translation (TMI '07)*, pages 85–94, Skövde, Sweden, 2007.

[7] P. Koehn. Europarl: A parallel corpus for statistical machine translation. In *Proceedings of MT Summit*, 2005.

[8] P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, C. Dyer, O. Bojar, A. Constantin, and E. Herbst. Moses: Open source toolkit for statistical machine translation. In *Proceedings of ACL*, Prague, Czech Republic, 2007.

[9] Y. Lü, M. Zhou, and S. Li. Automatic translation template acquisition based on bilingual structure alignment. *Computational Linguistics and Chinese Language Processing*, 6(1):83–108, 2001.

[10] J. Lundborg, T. Marek, M. Mettler, and M. Volk. Using the stockholm treealigner. In *Proceedings of the 6th Workshop on Treebanks and Linguistic Theories*, pages 73–78, Bergen, Norway, 2007.

[11] F. J. Och and H. Ney. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51, 2003.

[12] Y. Samuelsson and M. Volk. Automatic phrase alignment: Using statistical n-gram alignment for syntactic phrase alignment. In *Proceedings of the Sixth International Workshop on Treebanks and Linguistic Theories*, pages 139–150, Geneva, Switzerland, 2007.

[13] B. Schrader. *Exploiting Linguistic and Statistical Knowledge in a Text Alignment System.* PhD thesis, Universität Osnabrück, 2007.

[14] B. Taskar, S. Lacoste-Julien, and D. Klein. A discriminative matching approach to word alignment. In *Proceedings of HLT/EMNLP*, Vancouver, Canada, 2005.

[15] J. Tinsley, V. Zhechev, M. Hearne, and A. Way. Robust language pair-independent sub-tree alignment. In *Proceedings of Machine Translation Summit XI*, pages 467–474, Copenhagen, Denmark, 2007.

[16] W. Wang, J.-X. Huang, M. Zhou, and C.-N. Huang. Structure alignment using bilingual chunking. In *Proceedings of the 19th Conference on Computational Linguistics*, pages 1–7, Taipei, Taiwan, 2002.

[17] V. Zhechev and A. Way. Automatic generation of parallel treebanks. In *Proceedings of the 22nd International Conference on Computational Linguistics (CoLing)*, pages 1105–1112, 2008.

Author Index