Coling 2010

23rd International Conference on Computational Linguistics

Proceedings of the

4th Workshop on Cross Lingual Information Access

28 August 2010
Beijing International Convention Center

Produced by
Chinese Information Processing Society of China

To order the CD of Coling 2010 and its Workshop Proceedings, please contact:

Chinese Information Processing Society of China
No.4, Southern Fourth Street
Haidian District, Beijing, 100190
China
Tel: +86-010-62562916
Fax: +86-010-62562916
cips@iscas.ac.cn

Introduction

Welcome to the Coling Workshop on *Cross Lingual Information Access*.

Cross-lingual information access (CLIA) is concerned with technologies and applications that enable people to freely access information expressed in any language which may differ from the query language. As the web has grown to include rich contents in many different languages, and with rapid globalization, there is a growing demand for CLIA. Ordinary netizens who surf the Internet for special information and communicate in social networks, global companies which provide multilingual services to their multinational customers, governments who aim to lower the barriers to international commerce and collaboration and homeland security are in need of CLIA. This has triggered vigorous research and development activity in CLIA. This workshop is the fourth in a series of workshops and aims to address the need of CLIA. The previous three workshops were held during IJCAI 2007 in Hyderabad, IJCNLP 2008 in Hyderabad, and NAACL 2009 in Colorado.

In this workshop, in addition to Cross-lingual Information Retrieval (CLIR), the focus is on multi-lingual information extraction, information integration, summarization and other key technologies that are useful for CLIA. The workshop aims to bring together researchers from a variety of fields such as information retrieval, computational linguistics, machine translation, and practitioners from government and industry to address the issue of information need of multi-lingual societies. This workshop also aims to highlight and emphasize the contributions of Natural Language Processing (NLP) and Computational Linguistics to CLIA, in addition to the previously better represented viewpoint from Information Retrieval.

The workshop received a total of fourteen submissions, out of which the proceedings includes ten papers covering various aspects of this field. There are two papers on corpus acquisition. The papers by Pattabhi *et al.* and Lejune *et al.* focus on acquiring multilingual documents on various topics. There are three papers on bilingual lexicon acquisition. The papers by Okita *et al.* and Chatterjee *et al.* propose methods for word alignment and lexicon extraction from parallel and comparable corpora, while the paper by Rapp *et al.* proposes to learn dictionaries from monolingual corpora containing foreign words. Tang *et al.* do named entity translation for cross language question answering applications by combining a number of different sources, namely, machine translation, online encyclopedia and web documents. Falaise *et al.* use a light ontology to extraxt content from multilingual texts and user requests associated with images. Litvak *et al.* explore the performance of summarization methods across two languages. The paper by Vachchani *et al.* presents studies on pseudo relevance feedback utilizing multiple assisiting languages. Hajlaoui *et al.* discuss multilinguization and personalization in natural language based systems.

Besides these contributed papers, the workshop features two invited talks. Professor Pushpak Bhattacharya will speak on word sense disambiguation and information retrieval. Dr Tetsuya Sakai will speak on multilinguality at NTCIR.

With this gamut of topics, we look forward to a lively exchange of ideas in the workshop.

We take this opportunity to thank all the members of the Program Committee for their timely and insightful reviews, to the two invited speakers for kindly agreeing to speak at the workshop, the authors who submitted their work to this workshop and all the participants of this workshop.

Organizers, 4th Workshop on Cross Lingual Internet Access at COLING 2010, Beijing

Min Zhang, Sudeshna Sarkar, Raghavendra Udupa and Adam Lopez

Organizers:

Min Zhang, Institute for Infocomm Research(Singapore)
Sudeshna Sarkar, Indian Institute of Technology Kharagpur(India)
Raghavendra Udupa, Microsoft Research(India)
Adam Lopez, The University of Edinburgh(United Kingdom)

Program Committee:

Eneko Agirre, University of the Basque Country(Spain)
Ai Ti Aw, Institute for Infocomm Research(Singapore)
Sivaji Bandyopadhyay, Jadavpur University(India)
Pushpak Bhattacharyya, IIT Bombay(India)
Nicola Cancedda, Xerox Research Centre(France)
Patrick Saint Dizier, IRIT, Universite Paul Sabatier(France)
Nicola Ferro, University of Padua(Italy)
Guohong Fu, Heilongjiang University(China)
Cyril Goutte, National Research Council of Canada(Canada)
A Kumaran, Microsoft Research(India)
Gareth Jones, Dublin City University(Ireland)
Joemon Jose, University of Glasgow(United Kingdom)
Gina-Anne Levow, National Centre for Text Mining(United Kingdom)
Haizhou Li, Institute for Infocomm Research(Singapore)
Qun Liu, ICT/CAS(China)
Ting Liu, Harbin Institute of Technology(China)
Paul McNamee, Johns Hopkins University(USA)
Yao Meng, Fujitsu R&D Center Co. Ltd.(China)
Mandar Mitra, ISI Kolkata(India)
Doug Oard, University of Maryland, College Park(USA)
Carol Peters, Istituto di Scienza e Tecnologie dell'Informazione(Italy)
Maarten de Rijke, University of Amsterdam(Netherlands)
Paolo Rosso, Technical University of Valencia(Spain)
Hendra Setiawan, University of Maryland(USA)
L Sobha, AU-KBC, Chennai(India)
Rohini Srihari, University at Buffalo, SUNY(USA)
Ralf Steinberger, European Commission Joint Research Centre(Italy)
Le Sun, Institute of Software, CAS(China)
Chew Lim Tan, National University of Singapore(Singapore)
Vasudeva Varma, IIIT Hyderabad(India)
Thuy Vu, Institute for Infocomm Research(Singapore)
Haifeng Wang, Baidu(China)
Yunqing Xia, TsingHua University(China)
Deyi Xiong, Institute for Infocomm Research(Singapore)

Guodong Zhou, SooChow University(China)
Chengqing Zong, Institute of Automation, CAS(China)

Invited Speakers:

Pushpak Bhattacharya, Indian Institute of Technology Bombay(India)
Tsetsuya Sakai, Microsoft Research Asia(China)

Table of Contents

Conference Program

Saturday, August 28, 2010

8:35–8:45 Opening Remarks by Sudeshna Sarkar, Min Zhang, Adam Lopez and Raghavendra Udupa

8:45–9:40 Invited Talk 1:

Word Sense Disambiguation and IR
Pushpak Bhattacharyya

Invited Talk 2

11:00–11:55 *Multiliguality at NTCIR, and moving on ...*
Tetsuya Sakai

9:40–10:05 *Filtering news for epidemic surveillance: towards processing more languages with fewer resources*
Gael Lejeune, Antoine Doucet, Roman Yangarber and Nadine Lucas

10:05–10:30 *How to Get the Same News from Different Language News Papers*
T Pattabhi R K Rao and Sobha Lalitha Devi

10:30–11:00 Morning Break

11:55–12:20 *The Noisier the Better: Identifying Multilingual Word Translations Using a Single Monolingual Corpus*
Reinhard Rapp, Michael Zock, Andrew Trotman and Yue Xu

12:20–13:50 Lunch

13:50–14:15 *Multi-Word Expression-Sensitive Word Alignment*
Tsuyoshi Okita, Alfredo Maldonado Guerra, Yvette Graham and Andy Way

14:15–14:40 *Co-occurrence Graph Based Iterative Bilingual Lexicon Extraction From Comparable Corpora*
Diptesh Chatterjee, Sudeshna Sarkar and Arpit Mishra

14:40–15:05 *A Voting Mechanism for Named Entity Translation in EnglishChinese Question Answering*
Ling-Xiang Tang, Shlomo Geva, Andrew Trotman and Yue Xu

Saturday, August 28, 2010 (continued)

15:05–15:30 *Ontology driven content extraction using interlingual annotation of texts in the OMNIA project*
Achille Falaise, David Rouquet, Didier Schwab, Hervé Blanchon and Christian Boitet

15:30–16:00 Afternoon Break

16:00–16:25 *Towards multi-lingual summarization: A comparative analysis of sentence extraction methods on English and Hebrew corpora*
Marina Litvak, Mark Last, Slava Kisilevich, Daniel Keim, Hagay Lipman and Assaf Ben Gur

16:25–16:50 *More Languages, More MAP?: A Study of Multiple Assisting Languages in Multilingual PRF*
Vishal Vachhani, Manoj Chinnakotla, Mitesh Khapra and Pushpak Bhattacharyya

16:50–17:15 *Multilinguization and Personalization of NL-based Systems*
Najeh Hajlaoui and Christian Boitet

17:15–17:25 Closing

Word Sense Disambiguation and IR

Pushpak Bhattacharya
Department of Computer Science & Engineering,
Indian Institute of Technology Bombay,
Powai, Mumbai 400076,
India
pb@cse.iitb.ac.in

Multilinguality at NTCIR, and moving on...

Tetsuya Sakai
Microsoft Research Asia
Beijing
tesakai@microsoft.com

1 Abstract

NTCIR, often referred to as the Asian TREC, is eleven years old now. From NTCIR-1 (1999) to NTCIR-6 (2007), I was a task participant. From NTCIR-7 (2008), I started to serve as an organiser. From NTCIR-9 (2011), I will be serving as an NTCIR evaluation co-chair. In this talk, I will first look back on the past NTCIR rounds with a focus on crosslingual and multilingual tasks, e.g. Advanced Crosslingual Information Access (ACLIA). Then I will briefly discuss future plans for NTCIR which is currently going through drastic structural changes.

2 About the Speaker

Tetsuya Sakai received a Master's degree from Waseda University in 1993 and joined the Toshiba Corporate R&D Center in the same year. He received a Ph.D from Waseda University in 2000 for his work on information retrieval and filtering systems. From 2000 to 2001, he was a visiting researcher at the University of Cambridge Computer Laboratory. In 2007, he became Director of the Natural Language Processing Laboratory at NewsWatch, Inc. In 2009, he joined Microsoft Research Asia. He is Chair of IPSJ SIG-IFAT, Evaluation Co-chair of NTCIR, and Regional Representative to the ACM SIGIR Executive Committee (Asia/Pacific). He has served as a Senior PC member for ACM SIGIR, CIKM and AIRS. He is on the editorial board of Information Processing and Management and that of Information Retrieval the Journal. He has received several awards in Japan, mostly from IPSJ.

Filtering news for epidemic surveillance: towards processing more languages with fewer resources

Gaël Lejeune[1], **Antoine Doucet**[1], **Roman Yangarber**[2], **Nadine Lucas**[1]
[1]GREYC, University of Caen [2]CS department, University of Helsinki
first.last@info.unicaen.fr yangarbe@cs.helsinki.fi

Abstract

Processing content for security becomes more and more important since every local danger can have global consequences. Being able to collect and analyse information in different languages is a great issue. This paper addresses multilingual solutions for analysis of press articles for epidemiological surveillance. The system described here relies on pragmatics and stylistics, giving up "bag of sentences" approach in favour of discourse repetition patterns. It only needs light resources (compared to existing systems) in order to process new languages easily. In this paper we present here results in English, French and Chinese, three languages with quite different characteristics. These results show that simple rules allow selection of relevant documents in a specialized database improving the reliability of information extraction.

1 Multilingual techniques in information extraction

In natural language processing, information extraction is a task where, given raw text, a system is to give precise information fitting in a predefined semantic template.

1.1 Epidemic surveillance

Automated news surveillance is an important application of information extraction. The detection of terrorist events and economic surveillance were the first applications, in particular in the framework of the evaluation campaigns of the Message Understanding Conference (MUC) (MUC, 1992; MUC, 1993). In MUC-3 (1991) and MUC-4 (1992), about terrorism in Latin American countries, the task of participants was, given a collection of news feed data, to fill in a predetermined semantic template containing the name of the terrorist group that perpetrated a terrorist event, the name of the victim(s), the type of event, and the date and location where it occurred. In economic surveillance, one can for instance extract mergers or corporate management changes.

An application of information extraction that lately gained much importance is that of epidemiological surveillance, with a special emphasis on the detection of disease outbreaks. Given news data, the task is to detect epidemiological events, and extract the location where they occurred, the name of the disease, the number of victims, and the "case", that is, a text description of the event, that may be the "status" of victims (sick, injured, dead, hospitalised ...) or a written description of symptoms. Epidemiological surveillance has become a crucial tool with increasing world travel and the latest crises of SARS, avian flu, H1N1 ...

In this paper, we present an application to epidemic surveillance, but it may be equally applied to any subdomain of news surveillance.

1.2 Multilingual information extraction

As in many fields of NLP, most of the work in information extraction long focused on English data (Etzioni et al., 2008).Multilingual has often been understood as adding many monolingual systems, except in pioneer multilingual parsing (Vergne, 2002). Whereas English is nowadays the *lingua franca* in many fields (in particular, business), we will see that for several applications, this is not sufficient. Most news agencies are translating part of their feed into English (e.g., AFP[1] and Xinhua[2] for which the source languages are respectively French and Chinese), but a good deal of the data is never translated, while for the part that is, the translation process naturally incurs a delay that is, by essence, problematic in a field where exhaustivity and early detection are crucial aspects.

Subsequently, the ability to simultaneously handle documents written in different languages is becoming a more and more important feature (Poibeau et al., 2008; Gey et al., 2009). Indeed, in the field of epidemiological surveillance, it is especially important to detect a new event the very first time it is mentioned, and this very first occurrence will almost always happen in the local language (except for countries like Iraq for instance). Therefore, it is not enough to be able to deal with several languages : It is necessary to handle many. For instance, the Medical Information System (Medisys) of the European Community gathers news data in 42 different languages (Atkinson and der Goot, 2009) (now 45[3]).

1.3 Current approaches

There are currently 2 main approaches to multilingual information extraction. The first approach relies on the prior translation of all the documents into one common language (usually English), for which a well-performing information extraction system has been developed (Linge et al., 2009). Whereas the simple design of this solution is attractive, the current state of the art in machine translation only allows for mediocre results. Most monolingual information extraction systems indeed rely on a combination of grammatical patterns and specialized lexicons (Grishman et al., 2002; Riloff, 1996).

The second main approach consists in leaving documents in their original language but to translate the lexicons and extraction patterns into that language (Efimenko et al., 2004; Linge et al., 2009). However, the same problems occur as in the first approach because the patterns are strongly language-related. Yet, to “translate the system” seems more realistic than to translate the documents, as it can be done manually, and offline (once and for all, and not as documents arrive). The bottleneck is then that the amount of work for each language is enormous: it naturally requires the complete translation of the lexicon (for all trigger words), but the more challenging issue is the translation of patterns, whose language-dependence might well mean that the amount of work needed to translate them comes close to that required for writing them from scratch. In addition, this task must necessarily be achieved by a domain expert, with excellent skills in the languages at hand. One could want to tackle this problem by using machine learning but she will need training data in many languages. In practice, this will often mean that only a few major languages will be dealt with, whilst all the others (amongst which all low-resource languages), will again be totally discarded. One can then only wish that epidemics will chose to occur in locations handled by surveillance systems. . .

Both approaches additionally require a number of linguistic processing tools, in a number comparable to the number of languages to be dealt with: tokenizer, stemmer, syntactic analyzer, . . . One might therefore conclude that such techniques are not properly multilingual but rather monolingual methods that may be adapted to other languages individually.

In this paper, we explore a third approach to multilingual information extraction. We restrain ourselves to the sole use of truly mul-

[1] http://www.afp.com/afpcom/en

[2] http://www.xinhuanet.com/english2010/

[3] http://medusa.jrc.it/medisys/aboutMediSys.html

tilingual elements, facts that are equally true for any language. The approach hence relies on universals, relying, e.g., on stylistics and rhetorics.

2 Rationale of the experiment

The objective of the system is to monitor news in a variety of languages to detect disease outbreaks which is an important issue for an alert system in epidemic surveillance. For this task a simple and clear framework is needed in order to limit the amount of work for new languages while keeping good reliability. The main idea of our work is using text granularity and discourse properties to write rules that may be language independent, fast and reliable (Vergne, 2002). For this study, regularities at text level are exploited. These phenomena can be related to stylistics and pragmatics. It has already been shown that news discourse has its own constraints reflected in press articles of different languages (Van Dijk, 1988; Lucas, 2004).

2.1 Stylistic rules

Journalists all over the world know how to hook their potential readers. These methods are described in journalism schools (Itule and Anderson, 2006). One very important rule for journalists seems to be the"5W rule" which emphasise on the fact that answering to the questions"What","Where","When", "Why" and "Who" is a priority at the start of a paper. Only after that can journalists develop and give secondary information. This phenomenon is genre dependent and is exploited for processing texts by searching for repetitions.

Example 1 shows a piece of news where the disease name is found in the beginning of the news article and developed later on. No local pattern is needed to detect what the article is about, repetition phenomena is sufficient.

Example 2 is a counter example, where a disease name is found but not repeated. This French document reports on a pop music band being the "coqueluche" of Hip-Hop, which can mean "pertussis", but here means "fashion" in a figurative sense (underlining the fast spread of the band's popularity). Usually, figurative meanings are not used twice in the same article (Itule and Anderson, 2006) and hence the repetition criteria allows one to rightfully ignore this article.

2.2 Pragmatic rules

As press articles are made for humans, strong effort is exerted to ensure that readers will understand the main information with as few inferences as possible (Sperber and Wilson, 1998). In fact, the more inferences the reader has to make, the more errors he is likely to make and the more probability he will get confused and not read the full article. Repetitions are there to relieve the memory effort. A point that journalists pay much attention to is leaving as few ambiguities on main facts as possible. It means that potentially unknown or complicated terms will be used quite rarely. Only one main story will be developed in an article, other facts that are important will be developed elsewhere as main stories.

3 Our system

The system is based on the comparison of repetitions in the article to find documents relevant for epidemic surveillance and extract where the disease occurs and how many people are concerned.

3.1 String repetitions: relevant content

A system is not a human reader, so objective discourse marks are used by the system. Repetitions are known since the ancient times as reflecting discourse structure. A press article is divided into two parts, roughly the head and the rest of the news. The title and the first two sentences form the head or thematic part and the rest of the text is considered to be a development in an expository discourse.

Measles outbreak spreads north in B.C.
Number of cases hits **44** provincewide B.C.'s **measles** outbreak appears to have spread to northeastern areas of the province, after doctors confirmed two new cases of the disease in the Fort St. John and Fort Nelson areas on Thursday.

The new cases bring the total number of confirmed cases in the province to **44**, not including suspected but unconfirmed cases, said the B.C. Centre for Disease Control. Northern Health spokeswoman Eryn Collins said the virus had not been detected in the north in more than six years and the two new cases involve people who weren't immunized. [...] "It is suspected that at least two out-of-country visitors brought **measles** into Vancouver sometime in February or early March, as two separate strains of the virus have been identified," said a statement from the B.C. Centre for Disease Control earlier this week. So far, 17 cases of the **measles** have been detected in the Fraser Valley, 17 in the Vancouver area, seven in the southern Interior, two in northern B.C. and one on Vancouver island.

Figure 1: Example in English: repetition of disease name and cases

Cameroun/Musique : X-Maleya nouvelle **coqueluche** du Hip-Hop camerounais !
Le trio Hip-Hop Cameounais X-Maleya, a le vent en poupe. Le groupe qui s'illustre dans la tendance Hip-Hop, est aujourd'hui l'une des valeurs sres musicales grâce son second opus Yelele.
Derrière ces trois prénoms : Roger, Auguste et Haïs, se cachent un trio camerounais qui s'illustre dans le monde du Hip-Hop. [etc.] C'est donc, une nouvelle valeur sûre qu'incarnent eux trois Roger, Auguste et Haïs. Le groupe rencontre en effet, une ascension fulgurante. Les trois faiseurs de Hip-Hop, ont une seule idée en tête, continuer de se produire pour ceux qui les apprécient, toujours composer de belles mélodies et, ne pas oublier d'où ils viennent.

Figure 2: Example in French: no repetition

Strings that are present in both parts will be referred to as"relevant content". They are found in the beginning of the news and repeated in the development. To process as many languages as possible, repeated character strings will be searched (not words because Chinese for instance does not use graphic words).

3.2 Defining epidemic event

Epidemic events are captured through these information slots:

- Disease (What)
- Location (Where)
- Case, i.e.,People concerned (Who)

3.3 Selecting potentially relevant documents

This discourse related heuristic rule limits resources needed by the system. Many character strings that are repeated in the text reflect important terms. However, repetition alone does not allow to fill IE templates with detailed information as required. Accordingly, a lexical filter is applied on the repeated strings. 200 common disease names are used to filter information and find disease names. The idea behind the restricted list is that a journalist will use a common name to help his readers understand the message. Similarly, for locations, a list of country names and capitals provided by UN is

WHO checks smallpox reports in **Uganda**
LONDON, Thursday
The World Health Organisation said today it was investigating reports of suspected cases of the previously eradicated disease smallpox in eastern **Uganda**.
Smallpox is an acute contagious disease and was one of the worlds most feared sicknesses until it was officially declared eradicated worldwide in 1979.
“WHO takes any report of smallpox seriously, Gregory Hartl, a spokesman for the Geneva-based United Nations health agency, told Reuters via email.
“WHO is aware of the reports coming out of **Uganda** and is taking all the necessary measures to investigate and verify.”[etc.]

Figure 3: Example in English: repetition and location

used (about 500 items). Finally, in order to comply with a specific demand of partners, blacklist terms were used to detect less relevant articles (vaccination campaign for instance).

When a disease name is found in the relevant content, the article is selected as potentially relevant and the system tries to extract location and cases.

3.4 Extracting location and cases

To extract the location, the following heuristic is applied: the relevant location corresponds to a string in the“relevant content”. For instance, Example 3 shows that it allows for the system to find that the main event concerns Uganda but not London.

If numerous locations match, the system compares frequencies in the whole document: if one location is more than twice as frequent as others, it is considered as the relevant one. If no location is found, the location of the source is selected by default. In fact according to pragmatic rules when one reads an article in the Washington Post, she will be sure that it is about the United States even if it is not explicitly mentioned. To the contrary if the article is about Argentina it will be clearly mentioned so the reader has less chances of misunderstanding.

Concerning the cases, they are related to the first numeric information found in the document, provided the figures are not related to money or date (this is checked by a blacklist and simple regular expressions).

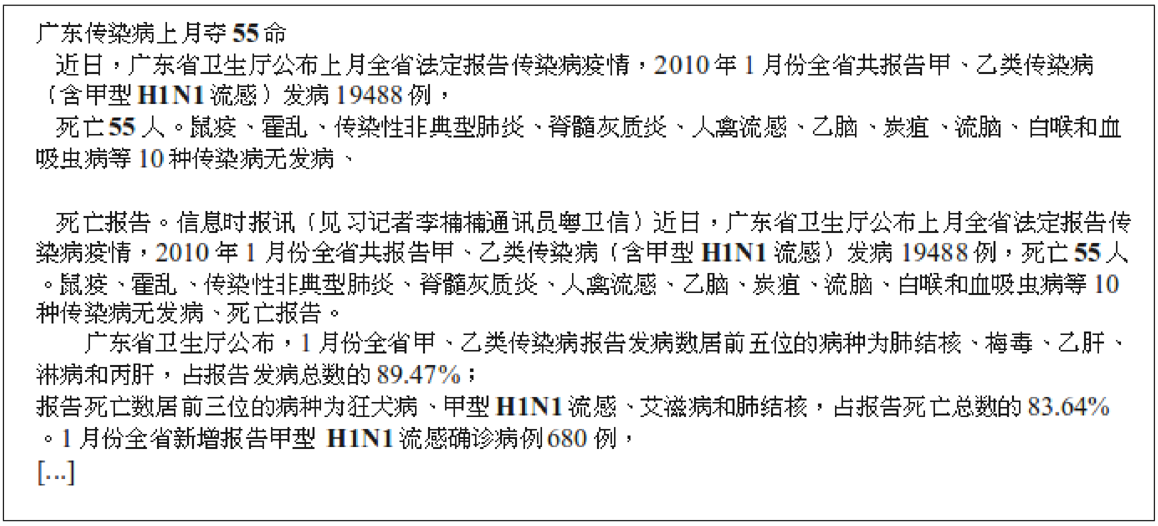
Furthermore the extracted cases are considered more relevant if they appear twice in the document, the system uses regular expressions to round up and compare them. See Example 4 where the number of dead people “55” is the first numeric information in the beginning and is repeated in the development (we chose an example where it is easy even for a non Chinese speaker to see the repetition). One can also note that the second repeated figure is “19488” which is the number of infected people.

4 Evaluation

It is important to insist on the fact that our system extracts the main event from one article, considering that secondary events have been or will be mentioned in another article. Often, the more topics are presented in one article, the less important each one is. In the case of epidemic surveillance, review articles or retrospectives are not first-hand, fresh and valuable information.

4.1 Corpus and Languages

For each language we randomly extracted documents from the Medisys website. Medisys documents are gathered using keywords: medical terms (including scientific disease names), but also weaker keywords such as casualties, hospital. . . This implies that some news document not related



广东传染病上月夺**55**命

近日，广东省卫生厅公布上月全省法定报告传染病疫情，2010年1月份全省共报告甲、乙类传染病（含甲型**H1N1**流感）发病19488例，

死亡**55**人。鼠疫、霍乱、传染性非典型肺炎、脊髓灰质炎、人禽流感、乙脑、炭疽、流脑、白喉和血吸虫病等10种传染病无发病、

死亡报告。信息时报讯（见习记者李楠楠通讯员粤卫信）近日，广东省卫生厅公布上月全省法定报告传染病疫情，2010年1月份全省共报告甲、乙类传染病（含甲型**H1N1**流感）发病19488例，死亡**55**人。鼠疫、霍乱、传染性非典型肺炎、脊髓灰质炎、人禽流感、乙脑、炭疽、流脑、白喉和血吸虫病等10种传染病无发病、死亡报告。

广东省卫生厅公布，1月份全省甲、乙类传染病报告发病数居前五位的病种为肺结核、梅毒、乙肝、淋病和丙肝，占报告发病总数的89.47%；

报告死亡数居前三位的病种为狂犬病、甲型**H1N1**流感、艾滋病和肺结核，占报告死亡总数的83.64%。1月份全省新增报告甲型 **H1N1**流感确诊病例680例，

[...]

Figure 4: Example in Chinese: 55 deaths from H1N1

to epidemic surveillance, but to accident reports for instance, are liable to be found in the database.

We must underline that in this framework, recall can only be estimated, notably because the news documents are keyword-filtered beforehand. However, our aim is not to provide an independent system, but to provide quick sorting of irrelevant news, prior to detailed analysis, which is the key issue of a surveillance and alert system. 200 documents were extracted for each language and manually tagged by native speakers with the following instructions:

- Is this article about an epidemic?
- If it is, please give when possible:

  Disease(s)

  Country (or Worldwide)

  Number of cases

100 of these annotated documents were used for fine-tuning the system, 100 others for evaluating. We chose for this study 3 fairly different languages for checking the genericity of the approach

- French, with its rather rich morphology,
- English,a rather isolating language with poor morphology,
- Chinese, a strict isolating language with poor morphology.

4.2 Results

These results were computed from a set of 100 annotated documents, as described in section 4. Table 1 shows recall, precision and F-measure for document selection(more examples are available online [4]) Table 2 compares automatically extracted slots and human annotated slots, therefore if an event is not detected by the system it will count as an error for each slot.

Table 1 shows that selection of documents is quite satisfactory and that recall is better than precision. This is mostly due to the fact that the system still extracts documents with low relevance. We found it impossible to predict if this is a general bias and whether it can be improved. The result analysis showed that many false negatives are due to cases when the piece of news is quite small, see for instance Example 5 where "Swine flu" is only found in the first two sentences, which implies the repetition criteria does not apply (and the system misses the document).

Table 2 shows the accuracy of the information entered into semantic slots, respec-

[4] http://sites.google.com/site/iesystemcoling2010

China has 100 cases of swine flu: state media
China has 100 confirmed cases of swine flu, state media said Tuesday, as data from the World Health Organization showed the disease had spread to 73 countries.
“The health ministry has reported that so far, China has 100 confirmed cases of A(H1N1) flu,” said a news report on state television CCTV. The report said the 100 cases were in mainland China, which does not include Hong Kong or Macau.

Figure 5: Example in English: Disease name not in “relevant content”

Language	Recall	Precision	F-measure
French	93%	88%	90%
English	88%	84%	86%
Chinese	92%	85%	88%

Table 1: Selecting documents

Language	Diseases	Locations	Cases
French	88%	87%	81%
English	81%	81%	78%
Chinese	82%	79%	77%

Table 2: Accuracy in filling slots

tively name of disease, location and number of cases. It is important to say that the descriptors extracted are really reliable in spite of the fact that the annotated set used for evaluation is fairly small: 100 documents per language, 30 to 40 of which were marked as relevant. The extraction of cases performs a bit worse than that of locations but the location is the most important to our end-users.

5 Discussion and Conclusion

Most research in Information Extraction (IE) focuses on building independent systems for each language, which is time and resource consuming. To the contrary, using common features of news discourse saves time. The system is not quite independent, but it allows filtering news feeds and it provides reasonable information even when no resources at all are available. Our results on English are worse than some existing systems (about 93% precision for Global health Monitor for instance) but these systems need strong resources and are not multilingual. We then really need a multilingual baseline to compare both approaches.

Recall is important for an alert system, but is very difficult to assess in the case of epidemiological surveillance. This measure is always problematic for web based documents, due to the fact that any randomly checked sample would only by sheer luck contain all the positive documents. The assumption here is that no important news has been missed by Medisys, and that no important news filtered from Medisys has been rejected.

One explanation for missed articles lies in the definition of the article header: it is too rigid. While this is fine for standard size news, it is inappropriate for short news, hence meaningful repetitions are missed in the short news. This is a flaw, because first alerts are often short news. In the future, we may wish to define a discourse wise detection rule to improve the location slot filling. The extraction of locations is currently plagued by a very long list of countries and capitals, most of which is not useful. Locations are actually mentioned in data according to states, provinces, prefectures, etc. The country list might be abandoned, since we do not favour external resources.

The methods that are presented here maintain good reliability in different languages, and the assumption that genre laws are useful has not been challenged yet. Light resources, about 750 items (to be compared to tens of thousands in classical IE systems), make it possible to strongly divide the amount of work needed for processing new languages. It might be attempted to refine the simple hypotheses underlying the program and build a better system for filtering relevant news. This approach is best suited when combined with elaborate pattern-based IE modules when available. Repetition can be checked for selecting documents prior to resource intensive semantic processing. It can also provide a few, easily fixable and efficient preliminary results where language resources are scarce or not available at all.

References

Atkinson, Martin and Erik Van der Goot. 2009. Near real time information mining in multilingual news. In *18th International World Wide Web Conference (WWW2009).*

Efimenko, Irina, Vladimir Khoroshevsky, and Victor Klintsov. 2004. Ontosminer family: Multilingual ie systems. In *SPECOM 2004: 9th Conference Speech and Computer.*

Etzioni, Oren, Michele Banko, Stephen Soderland, and Daniel S. Weld. 2008. Open information extraction from the web. *Commun. ACM*, 51(12):68–74.

Gey, Fredric, Jussi Karlgren, and Noriko Kando. 2009. Information access in a multilingual world: transitioning from research to real-world applications. *SIGIR Forum*, 43(2):24–28.

Grishman, Ralph, Silja Huttunen, and Roman Yangarber. 2002. Information extraction for enhanced access to disease outbreak reports. *Journal of Biomedical Informatics*, 35(4):236–246.

Itule, Bruce and Douglas Anderson. 2006. *News Writing and Reporting for Today's Media.* McGraw-Hill Humanities.

Linge, JP, R Steinberger, T P Weber, R Yangarber, E van der Goot, D H Al Khudhairy, and N I Stilianakis. 2009. Internet surveillance systems for early alerting of threats. *Eurosurveillance*, 14.

Lucas, Nadine. 2004. The enunciative structure of news dispatches, a contrastive rhetorical approach. *Language, culture, rhetoric*, pages 154–164.

MUC. 1992. *Proceedings of the 4th Conference on Message Understanding, MUC 1992, McLean, Virginia, USA, June 16-18, 1992.*

MUC. 1993. *Proceedings of the 5th Conference on Message Understanding, MUC 1993, Baltimore, Maryland, USA, August 25-27, 1993.*

Poibeau, Thierry, Horacio Saggion, and Roman Yangarber, editors. 2008. *MMIES '08: Proceedings of the Workshop on Multi-source Multilingual Information Extraction and Summarization*, Morristown, NJ, USA. Association for Computational Linguistics.

Riloff, Ellen. 1996. Automatically generating extraction patterns from untagged text. In *AAAI/IAAI, Vol. 2*, pages 1044–1049.

Sperber, Dan and Deirdre Wilson. 1998. *Relevance: Communication and cognition.* Blackwell press, Oxford U.K.

Van Dijk, T.A. 1988. *News as discourse.* Lawrence Erlbaum Associates, Hillsdale N.J.

Vergne, Jacques. 2002. Une méthode pour l'analyse descendante et calculatoire de corpus multilingues: application au calcul des relations sujet-verbe. In *TALN 2002*, pages 63–74.

How to Get the Same News from Different Language News Papers

T. Pattabhi R. K Rao
AU-KBC Research Centre
Anna University Chennai

Sobha Lalitha Devi
AU-KBC Research Centre
Anna University Chennai
sobha@au-kbc.org

Abstract

This paper presents an ongoing work on identifying similarity between documents across News papers in different languages. Our aim is to identify similar documents for a given News or event as a query, across languages and make cross lingual search more accurate and easy. For example given an event or News in English, all the English news documents related to the query are retrieved as well as in other languages such as Hindi, Bengali, Tamil, Telugu, Malayalam, Spanish. We use Vector Space Model, a known method for similarity calculation, but the novelty is in identification of terms for VSM calculation. Here a robust translation system is not used for translating the documents. The system is working with good recall and precision.

1 Introduction

In this paper we present a novel method for identifying similar News documents from various language families such as Indo-European, Indo- Aryan and Dravidian. The languages considered from the above language families are English, Hindi, Bengali, Tamil, Telugu, Malayalam and Spanish. The News documents in various languages are obtained using a crawler. The documents are represented as vector of terms.

Given a query in any of the language mentioned above, the documents relevant to the query are retrieved. The first two document retrieved in the language of the query is taken as base for the identification of similar documents. The documents are converted into terms and the terms are translated to other languages using bilingual dictionaries. The terms thus obtained is used for similarity calculation. The paper is further organized as follows. In the following section 2, related work is described. In section 3, the algorithm is discussed. Section 4 describes experiments and results. The paper concludes with section 5.

2 Related Work

In the past decade there has been significant amount of work done on finding similarity of documents and organizing the documents according to their content. Similarity of documents are identified using different methods such as Self-Organizing Maps (SOMs) (Kohonen et al, 2000; Rauber, 1999), based on Ontologies and taxanomy (Gruber, 1993; Resnik, 1995), Vector Space Model (VSM) with similarity measures like Dice similarity, Jaccard's similarity, cosine similarity (Salton, 1989).

Many similarity measures were developed, such as information content (Resnik, 1995) mutual information (Hindle, 1990), Dice coefficient (Frakes and Baeza-Yates, 1992), cosine coefficient (Frakes and Baeza-Yates, 1992), distance-based measurements (Lee et al., 1989; Rada et al., 1989), and feature contrast model (Tversky, 1977). McGill etc. surveyed and compared 67 similarity measures used in information retrieval (McGill et al., 1979).

Proceedings of the 4th International Workshop on Cross Lingual Information Access at COLING 2010, pages 11–15,
Beijing, August 2010

3 Methodology

Similarity is a fundamental concept. Two documents can be said to be similar if both the documents have same content, describing a topic or an event or an entity. Similarity is a measure of degree of resemblance, or commonality between the documents.

In this work we have used Vector Space Model (VSM) for document representation. In VSM the documents are represented as vectors of unique terms. Here we have performed experiments by creating three types of document vector space models. In the first case we have taken all unique words in the document collection for vector of terms. In the second case we take the terms after removing all stop words. In the third case we have taken a sequence of words as terms. After the document model is built we use cosine similarity measure to identify the degree of similarity between documents.

In this work we have taken documents from the languages mentioned in the previous section. For the purpose of identifying similar documents across the languages we use map of term vectors of documents from English to other languages. Using the term vector map we can identify similar documents for various languages.

3.1 Similarity analyser

The main modules are i) Document vector creator ii) Translator and iii) Similarity identifier.

a) Document Vector Creator: Each document is represented as vector of terms. Here we take three types of term vectors. In the first type a single word is taken as a term which is the standard implementation of VSM. In the second type single words are taken but the stop words are removed.

In the third type each term is a sequence of words, where we define the number of words in the sequence as 4. This moving window of 4 is obtained by performing many experiments using different combinations of words. So our term of vector is defined as a set of four consecutive words, where the last three words in the preceding sequence is considered as the first three words in the following sequence. For example if a sentence has 10 words (w), the vector of terms for this sentence is w1w2w3w4, w2w3w4w5, w3w4w5w6, w4w5w6w7, w5w6w7w8, w6w7w8w9, w7w8w9w10. The weights of the terms in the vector are the term frequency and inverse document frequency (tf-idf). While creating document vectors, for Indian languages which are highly agglutinative and morphologically rich we use morphological analyzer to reduce the word into its root and it is used for document vector creation.

The first two experiments are the standard VSM implementation. The third experiment differs in the way the terms are taken for building the VSM. For building the VSM model which is common for all language document texts, it is essential that there should be translation/transliteration tool. First the terms are collected from individual language documents and a unique list is formed. The unique list of words is then translated using the translator module.

b) Word by Word Translator: In this module, the terms from English documents are taken and are translated to different languages. The translation is done word by word with the use of bilingual and multilingual synset dictionaries. This translation creates a map of terms from English to different languages. We have used bilingual dictionaries from English to Spanish, Hindi, Tamil, Telugu, and Malayalam dictionaries. Also we have used multilingual synset dictionaries for English, Tamil, Telugu, Hindi, and Malayalam. For each pair of bilingual dictionaries there are more than 100K root words. Since in this work we do not require syntactically and semantically correct translation of the sentences we adopted word to word translation. Hence we did not use any other system such as SMT for English to Indian languages. Named entities require transliteration. Here we have used a transliteration tool. This tool uses rule based approach, based on the phoneme match. The transliteration tool produces all possible transliteration outputs. Here we take into consideration the top five best possible outputs. For example the name "Lal Krishna Advani" would get transliterations in Indian languages as "laala krishna athvaani", "laala krishna advaani".

c) Similarity Identifier: The similarity identifier module takes the query in the form document as input and identifies all relevant documents. The similarity identifier uses cosine similarity measure over documents vector creator. The cosine similarity measure is the dot product of two vectors and is between 0 and 1 value. The more it is closer to 1, the similarity is more. The formula of cosine similarity is as follows:

$$\mathrm{Sim}(S1,S2)_{tj} = \Sigma\ (W1j \times W2j\ )\ \text{-- (1)}$$

Where,

tj is a term present in both vectors S1and S2.
W1j is the weight of term tj in S1 and
W2j is the weight of term tj in S2.

The weight of term tj in the vector S1 is calculated by the formula given by equation (2), below.

Wij=(tf*log(N/df))/[sqrt(Si12+Si22+…+Sin2)] --(2)

Where,

tf = term frequency of term tj
N=total number of documents in the collection
df = number of documents in the collection that the term tj occurs in.
sqrt represents square root

The denominator [sqrt(Si12+Si22+……+Sin2)] is the cosine normalization factor. This cosine normalization factor is the Euclidean length of the vector Si, where 'i' is the document number in the collection and Sin2 is the square of the product of (tf*log(N/df)) for term in the vector Si.

4 Experiments and Results

We have performed three experiments with two different data sets. The first data set was collected by crawling the web for a single day's news articles and obtained 1000 documents from various online news magazines in various languages. The test set was taken from Times of India, The Hindu for English, BBC, Dinamani, Dinamalar for Tamil, Yahoo for Telugu, Matrubhumi for Malayalam, BBC and Dainik Jagran for Hindi and BBC for Spanish. The distribution of documents in the first set for various languages is as follows: 300 English, 200 Tamil, 150 Telugu, 125 Hindi, 125 Malayalam, 50 Spanish. The figure 1 given below shows the language distribution in this first set. The number of similar documents were 600 in this set.

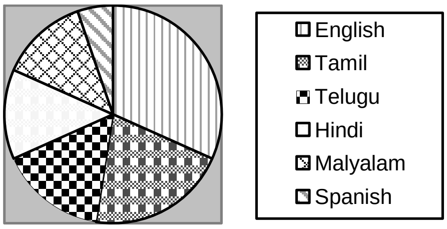


Figure 1. Data Distribution of Set 1

In the second data set we have taken news documents of one week time duration. This consisted of 9750 documents. The language distribution for this data set is shown in figure 2. This second data set consisted of 5350 similar documents.

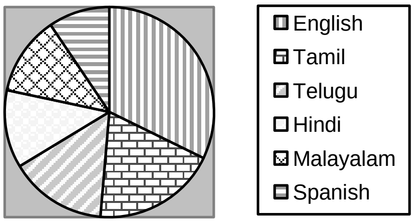


Figure 2. Data Distribution of Set 2

In the first experiment we took all the unique words (separated by white space) as terms for building the document vector. In the second experiment the terms taken were same as the first experiment, except that all the stop words were removed. In the third experiment, the terms taken for document vector creation were four consecutive words. The results obtained for three experiments for data set 1 is shown in Table 1. And results for data set 2 are shown in Table 2. Table 3 shows the similarity identification for various languages.

Here we take a news story document as a query and perform similarity analysis across all documents in the document collection to identify similarly occurring news stories. In the first data set in the gold standard there are 600 similar pairs of documents. And in the second data set there are 5350 similar pairs of documents in the gold standard.

It is observed that even though there were more similar documents which could have been identified, but the system could not identify those documents. The cosine measure for those unidentified documents was found to be lower than 0.8. We have taken 0.8 as the threshold for documents to be considered similar. In the documents which were not identified by the system, the content described consisted of less number of words. These were mostly two paragraph documents; hence the similarity score obtained was less than the threshold. In experiment three, we find that the number of false positives is decreased and also the number of documents identified similar is increased. This is because, in this case the system sees for terms of four words and hence single word matches are reduced. This reduces false positives. The other advantage of this is the words get the context, in a sense that the words in each sequence are not independent. The words get an order and are sensitive to that order. This solves sense disambiguation. Hence we find that it is solving the polysemy problem to some extent. The system can be further improved by creating robust map files between terms in different languages. The bilingual dictionaries also need to be improved.

In our work, since we are using a sequence of words as terms for document vectors, we do not require proper, sophisticated translation systems. A word by word translation would suffice to get the desired results.

Exp No	Gold std Similari ty	System Identified Correct	System Identified Wrong	Pre c %	Rec %
1	600	534	50	91.4	89.0
2	600	547	44	92.5	91.2
3	600	565	10	98.3	94.2

Table 1. Similarity Results on Data Set 1

Exp No	Gold Standard Similarity	System Identified Correct	System Identifi ed Wrong	Prec %	Rec %
1	5350	4820	476	91.0	90.0
2	5350	4903	410	92.3	91.6
3	5350	5043	114	97.8	94.3

Table 2. Similarity Results on Data Set 2

Lang	Gold Std similar docs	System Identifi ed correct	System Identifi ed wrong	Prec %	Rec %
Eng	1461	1377	30	97.86	94.25
Span	732	690	15	97.87	94.26
Hin	588	554	11	98.05	94.22
Mal	892	839	19	97.78	94.05
Tam	932	880	22	97.56	94.42
Tel	745	703	17	97.63	94.36
AVG				**97.79**	**94.26**

Table 3.Similarity Results Data Set with Ex:3

5 Conclusion

Here we have shown how we can identify similar News document in various languages. The results obtained are encouraging; we obtain an average precision of 97.8% and recall of 94.3%. This work differs from previous works in two aspects: 1) no language preprocessing of the documents is required and 2) terms taken for VSM are a sequence of four words.

References

Frakes, W. B. and Baeza-Yates, R., editors 1992. *Information Retrieval, Data Structure and Algorithms*. Prentice Hall.

T. R. Gruber. 1993. *A translation approach to portable ontologies,* Knowledge Acquisition, 5(2):199–220.

Hindle, D. 1990. *Noun classification from predicate-argument structures*. In Proceedings of ACL-90, pages 268–275, Pittsburg, Pennsylvania.

Kohonen, Teuvo Kaski, Samuel Lagus, Krista Salojarvi, Jarkko Honkela, Jukka Paatero,Vesa Saarela, Anti. 2000. *Self organisation of a massive document collection*, IEEE Transactions on Neural Networks, 11(3): 574-585.

Lee, J. H., Kim, M. H., and Lee, Y. J. 1989. *Information retrieval based on conceptual distance in is-a hierarchies.* Journal of Documentation, 49(2):188–207.

McGill et al., M. 1979. *An evaluation of factors affecting document ranking by information retrieval systems.* Project report, Syracuse University School of Information Studies.

Rauber, Andreas Merkl, Dieter. 1999. *The SOMLib digital library system,* In the Proceedings of the 3rd European Conference on Research and Advanced Technology for Digital Libraries (ECDL'99), Paris, France. Berlin: 323-341.

Rada, R., Mili, H., Bicknell, E., and Blettner, M. 1989. *Development and application of a metric on semantic nets*. IEEE Transaction on Systems, Man, and Cybernetics, 19(1):17–30.

P. Resnik. 1995. *Using information content to evaluate semantic similarity in taxonomy,* Proceedings of IJCAI: 448–453.

Salton, Gerald. 1989. *Automatic Text Processing: The Transformation, Analysis and Retrieval of Information by Computer,* Reading, MA: Addison Wesley

Tversky, A. 1977. *Features of similarity*. Pychological Review, 84:327–352.

The Noisier the Better: Identifying Multilingual Word Translations Using a Single Monolingual Corpus

Reinhard Rapp
University of Tarragona
GRLMC
reinhardrapp@gmx.de

Michael Zock
Laboratoire d'Informatique Fondamentale
CNRS Marseille
Michael.Zock@lif.univ-mrs.fr

Abstract

The automatic generation of dictionaries from raw text has previously been based on parallel or comparable corpora. Here we describe an approach requiring only a single monolingual corpus to generate bilingual dictionaries for several language pairs. A constraint is that all language pairs have their target language in common, which needs to be the language of the underlying corpus. Our approach is based on the observation that monolingual corpora usually contain a considerable number of foreign words. As these are often explained via translations typically occurring close by, we can identify these translations by looking at the contexts of a foreign word and by computing its strongest associations from these. In this work we focus on the question what results can be expected for 20 language pairs involving five major European languages. We also compare the results for two different types of corpora, namely *newsticker texts* and *web corpora*. Our findings show that results are best if English is the source language, and that noisy web corpora are better suited for this task than well edited newsticker texts.

1 Introduction

Established methods for the identification of word translations are based on *parallel* (Brown et al., 1990) or *comparable corpora* (Fung & McKeown, 1997; Fung & Yee, 1998; Rapp, 1995; Rapp 1999; Chiao et al., 2004). The work using parallel corpora such as Europarl (Koehn, 2005; Armstrong et al., 1998) or JRC Acquis (Steinberger et al., 2006) typically performs a length-based sentence alignment of the translated texts, and then tries to conduct a word alignment within sentence pairs by determining word correspondences that get support from as many sentence pairs as possible. This approach works very well and can easily be put into practice using a number of freely available open source tools such as Moses (Koehn et al., 2007) and Giza++ (Och & Ney, 2003).

However, parallel texts are a scarce resource for many language pairs (Rapp & Martín Vide, 2007), which is why methods based on comparable corpora have come into focus. One approach is to extract parallel sentences from comparable corpora (Munteanu & Marcu, 2005; Wu & Fung, 2005). Another approach relates co-occurrence patterns between languages. Hereby the underlying assumption is that across languages there is a correlation between the co-occurrences of words which are translations of each other. If, for example, in a text of one language two words *A* and *B* co-occur more often than expected by chance, then in a text of another language those words which are the translations of *A* and *B* should also co-occur more frequently than expected.

However, to exploit this observation some bridge needs to be built between the two languages. This can be done via a basic dictionary comprising some essential vocabulary. To put it simply, this kind of dictionary allows a (partial) word-by-word translation from the source to the target language,[1] so that the result can be considered as a pair of monolingual corpora. Deal-

[1] Note that this translation can also be conducted at the level of co-occurrence vectors rather than at the text level.

Proceedings of the 4th International Workshop on Cross Lingual Information Access at COLING 2010, pages 16–25,
Beijing, August 2010

ing only with monolingual corpora means that the established methodology for computing similar words (see e.g. Pantel & Lin, 2002), which is based on Harris' (1954) *distributional hypothesis*, can be applied. It turns out that the most similar words between the two corpora effectively identify the translations of words.

This approach based on comparable corpora considerably relieves the data acquisition bottleneck, but has the disadvantage that the results tend to lack accuracy in practice.

As an alternative, there is also the approach of identifying orthographically similar words (Koehn & Knight, 2002) which has the advantage that it does not even require a corpus. A simple word list will suffice. However, this approach works only for closely related languages, and has limited potential otherwise.

We propose here to generate dictionaries on the basis of foreign word occurrences in texts. As far as we know, this is a method which has not been tried before. When doing so, a single monolingual corpus can be used for all source languages for which it contains a sufficient number of foreign words. A constraint is that the target language must always be the language of the monolingual corpus,[2] which therefore all dictionaries have in common.

2 Approach and Language Resources

Starting from the observation that monolingual dictionaries typically include a considerable number of foreign words, the basic idea is to consider the most significant co-occurrences of a foreign word as potential translation candidates. This implies that the language of the underlying corpus must correspond to the target language, and that this corpus can be utilized for any source language for which word citations are well represented.

As the use of foreign language words in texts depends on many parameters, including writer, text type, status of language and cultural background, it is interesting to compare results when varying some of these parameters. However, due to the general scarceness of foreign word citations our approach requires very large corpora. For this reason, we were only able to vary two parameters, namely language and text type.

Some large enough corpora that we had at our disposal were the *Gigaword Corpora* from the Linguistic Data Consortium (Mendonça et al., 2009a; Mendonça et al., 2009b) and the *WaCky Corpora* described in Sharoff (2006), Baroni et al. (2009), and Ferraresi et al. (2010). From these, we selected the following for this study:

- French WaCky Corpus (8.2 GB)
- German WaCky Corpus (9.9 GB)
- Italian WaCky Corpus (10.4 GB)
- French Gigaword 2nd edition (5.0 GB)
- Spanish Gigaword 2nd edition (6.8 GB)

The memory requirements shown for each corpus relate to ANSI coded text only versions. We derived these from the original corpora by removing linguistic annotation (for the WaCky corpora) and XML markup, and by converting the coding from UTF8 to ANSI.

Both Gigaword corpora consist of newsticker texts from several press agencies. Newsticker text is a text type closely related to newspaper text. It is usually carefully edited, and the vocabulary is geared towards easy understanding for the intended readership. This implies that foreign word citations are kept to a minimum.

In contrast, the WaCky Corpora have been downloaded from the web and represent a great variety of text types and styles. Hence, not all texts can be expected to have been carefully edited, and mixes between languages are probably more frequent than with newsticker text.

As in this work English is the main source language, and as we have dealt with it as a target language already in Rapp & Zock (2010), we do not use the respective English versions of these corpora here. We also do not use the *Wikipedia XML* C*orpora* (Denoyer et al., 2006) as these greatly vary in size for different languages which makes comparisons across languages somewhat problematic. In contrast, the sizes of the above corpora are within the same order of magnitude (1 billion words each), which is why we do not control for corpus size here.

[2] Although in principle it would also be possible to determine relations between foreign words from different languages within a corpus, this seems not promising as the problem of data sparsity is likely to be prohibitive.

Concerning the number of foreign words within these corpora, we might expect that, given the status of English as the world's premiere language, English foreign words should be the most frequent ones in our corpora. As French and Spanish are also prominent languages, foreign words borrowed from them may be less frequent but should still be common, whereas borrowings from German and Italian are expected to be the least likely ones. From this point of view the quality of the results should vary accordingly. But of course there are many other aspects that are important, for example, relations between countries, cultural background, relatedness between languages, etc. As these are complex influences with intricate interactions, it is impossible to accurately anticipate the actual outcome. In other words, experimental work is needed. Let us therefore describe our approach.

For identifying word translations within a corpus, we assume that the strongest association to a foreign word is likely to be its translation. This can be justified by typical usage patterns of foreign words often involving, for example, an explanation right after their first occurrence in a text.

Associations between words can be computed in a straightforward manner by counting word co-occurrences followed by the application of an association measure on the co-occurrence counts. Co-occurrence counts are based on a text window comprising the 20 words on either side of a given foreign word. On the resulting counts we apply the log-likelihood ratio (Dunning, 1993). As explained by Dunning, this measure has the advantage to be applicable also on low counts, which is an important characteristic in our setting where the problem of data sparseness is particularly severe. This is also the reason why we chose a window size somewhat larger than the ones used in most other studies.

Despite its simplicity this procedure of computing associations to foreign words is well suited for identifying word translations. As mentioned above, we assume that the strongest association to a foreign word is its best translation.

We did this for words from five languages (English, French, German, Italian, and Spanish). The results are shown in the next section. In order to be able to quantitatively evaluate the quality of our results, we counted for all source words of a language the number of times the expected target word obtained the strongest association score.

Our expectations on what should count as a correct translation had been fixed before running the experiments by creating a gold standard for evaluation. We started from the list of 100 English words (nouns, adjectives and verbs) which had been introduced by Kent & Rosanoff (1910) in a psychological context.

We translated these English words into each of the four target languages, namely French, German, Italian, and Spanish. As we are at least to some extent familiar with these languages, and as the Kent/Rosanoff vocabulary is fairly straightforward, we did this manually. In cases where we were aware of ambiguities, we tried to come up with a translation relating to what we assumed to be the most frequent of a word's possible senses. In case of doubt we consulted a number of written bilingual dictionaries, the *dict.leo.org* dictionary website, and the translation services provided by Google and Yahoo. For each word, we always produced only a single translation. In an attempt to provide a common test set, the appendix shows the resulting list of *word equations* in full length for reference by interested researchers.

It should be noted that the concept of *word equations* is a simplification, as it does not take into account the fact that words tend to be ambiguous, and that ambiguities typically do not match across languages. Despite these shortcomings we nevertheless use this concept. Let us give some justification.

Word ambiguities are omnipresent in any language. For example, the English word *palm* has two meanings (*tree* and *hand*) which are usually expressed by different words in other languages. However, for our gold standard we must make a choice. We can not include two or more translations in one word equation as this would contradict the principle that all words in a word equation should share their main sense.

Another problem is that, unless we work with dictionaries derived from parallel corpora, it is difficult to estimate how common a translation is. But if we included less common translations in our list, we would have to give their matches a smaller weight during evaluation.

This, however, is difficult to accomplish accurately. This is why, despite their shortcomings, we use word equations in this work.

Evaluation of our results involves comparing a predicted translation to the corresponding word in the gold standard. We consider the predicted translation to be correct if there is a match, otherwise we consider it as false. While in principle possible, we do not make any finer distinctions concerning the quality of a match.

A problem that we face in our approach is what we call the *homograph trap*. What we mean by this term is that a foreign word occurring in a corpus of a particular language may also be a valid word in this language, yet possibly with a different meaning. For example, if the German word *rot* (meaning *red*) occurs in an English corpus, its occurrences can not easily be distinguished from occurrences of the English word *rot*, which is a verb describing the process of decay.

Having dealt with this problem in Rapp & Zock (2010) we will not elaborate on it here, rather we will suggest a workaround. The idea is to look only at a very restricted vocabulary, namely the words defined in our gold standard. There we have 100 words in each of the five languages, i.e. 500 words altogether. The question is how many of these words occur more often than once. Note, however, that apart from English (which was the starting point for the gold standard), repetitions can occur not only across languages but also within a language. For example, the Spanish word *sueño* means both *sleep* and *dream*, which are distinct entries in the list.

The following is a complete list of words showing either of these two types of repetitions, i.e. exact string matches (taking into account capitalization and accents): alto (4), bambino (2), Bible (2), bitter (2), casa (2), commando (2), corto (2), doux (2), duro (2), fruit (2), justice (2), lento (2), lion (2), long (2), luna (2), mano (2), memoria (2), mouton (2), religion (2), sacerdote (2), sueño (2), table (2), whisky (4).

However, as is obvious from this list, these repetitions are due to common vocabulary of the languages, with *whisky* being a typical example. They are not due to incidental string identity of completely different words. So the latter is not a problem (i.e. causing the identification of wrong translations) as long as we do not go beyond the vocabulary defined in our gold standard.

For this reason and because dealing with the full vocabulary of our (very large) corpora would be computationally expensive, we decided to replace in our corpora all words absent from the gold standard by a common designator for unknown words. Also, in our evaluations, for the target language vocabulary we only use the words occurring in the respective column of the gold standard.

So far, we always computed translations to single source words. However, if we assume, for example, that we already have word equations for four languages, and all we want is to compute the translations into a fifth language, then we can simply extend our approach to what we call the *product-of-ranks algorithm*. As suggested in Rapp & Zock (2010) this can be done by looking up the ranks of each of the four given words (i.e. the words occurring in a particular word equation) within the association vector of a translation candidate, and by multiplying these ranks. So for each candidate we obtain a product of ranks. We then assume that the candidate with the smallest product will be the best translation.[3]

Let us illustrate this by an example: If the given words are the variants of the word *nervous* in English, French, German, and Spanish, i.e. *nervous*, *nerveux*, *nervös*, and *nervioso*, and if we want to find out their translation into Italian, we would look at the association vectors of each word in our Italian target vocabulary. The association strengths in these vectors need to be inversely sorted, and in each of them we will look up the positions of our four given words. Then for each vector we compute the product of the four ranks, and finally sort the Italian vocabulary according to these products. We would then expect that the correct Italian translation, namely *nervoso*, ends up in the first position, i.e. has the smallest value for its product of ranks.

[3] Note that, especially in the frequent case of zero-co-occurrences, many words may have the same association strength, and rankings within such a group of words may be arbitrary within a wide range. To avoid such arbitrariness, it is advisable to assign all words within such a group the same rank, which is chosen to be the average rank within the group.

In the next section, we will show the results for this algorithm in addition to those for single source language words.

As a different matter, let us mention that for our above algorithm we do not need an explicit identification of what should count as a foreign word. We only need a list of words to be translated, and a list of target language words containing the translation candidates from which to choose. Overlapping vocabulary is permitted. If the overlapping words have the same meaning in both languages, then there is no problem and the identification of the correct translation is rather trivial as co-occurrences of a word with itself tend to be frequent. However, if the overlapping words have different meanings, then we have what we previously called a *homogaph trap*. In such (for small vocabularies very rare) cases, it would be helpful to be able to distinguish the occurrences of the foreign words from those of the homograph. However, this problem essentially boils down to a word sense disambiguation task (actually a hard case of it as the foreign word occurrences, and with them the respective senses, tend to be rare) which is beyond the scope of this paper.

3 Experimental Results and Evaluation

We applied the following procedure on each of the five corpora: The language of the respective corpus was considered the target language, and the vocabulary of the respective column in the gold standard was taken to be the target language vocabulary. The other languages are referred to as the source languages, and the corresponding columns of the gold standard contain the respective vocabularies. Using the algorithm described in the previous section, for each source vocabulary the following procedure was conducted: For every source language word the target vocabulary was sorted according to the respective scores. The word obtaining the first rank was considered to be the predicted translation. This predicted translation was compared to the translation listed in the gold standard. If it matched, the prediction was counted as correct, otherwise as wrong.

	Source Languages					
	DE	EN	FR	ES	IT	all
DE WaCky	–	54	22	18	20	48
ES Giga	9	42	37	–	29	56
FR Giga	15	45	–	20	14	49
FR WaCky	27	59	–	16	21	50
IT WaCky	17	53	29	27	–	56
Average	17.0	50.6	29.3	20.3	21.0	51.8

Table 1: Number of correctly predicted translations for various corpora and source languages. Column *all* refers to the parallel use of all four source languages using the product-of-ranks algorithm.

Table 1 lists the number of correct predictions for each corpus and for each source language. These results lead us to the following three conclusions:

1) The noisier the better

We have only for one language (French) both a Gigaword and a WaCky corpus. The results based on the WaCky corpus are clearly better for all languages except Spanish. Alternatively, we can also look at the average performance for the five source languages among the three WaCky corpora, which is 30.3, and the analogous performance for the two Gigaword corpora, which is 26.4. These findings lend some support to our hypothesis that noisy web corpora are better suited for our purpose than carefully edited newsticker corpora, which are probably more successful in avoiding foreign language citations

2) English words are cited more often

In the bottom row, Table 1 shows for each of the five languages the scores averaged over all corpora. As hypothesized previously, we can take citation frequency as an indicator (among others) of the "importance" of a language. And citation frequency can be expected to correlate with our scores. With 50.6, the average score for English is far better than for any other language, thereby underlining its special status among world languages. With an average score of 29.3 French comes next which confirms the hypothesis that it is another world language receiving considerable attention elsewhere. Somewhat surprising is the finding that Spanish can not keep up with French and obtains an average score of 20.3 which is even lower than the 21.0 for Italian. A possible explanation is the fact that we are only dealing with European languages here, and that the cultural influence of the Roman Empire and Italy has been so considerable in Europe that it may well account for this. So the status of Spanish in the world may not be well reflected in our selection of corpora. Finally, the average score of 17.0 for German shows that it is the least cited language in our selection of languages. Bear in mind, though, that German is the only clearly Germanic language here, and that its vocabulary is very different from that of the other languages. These are mostly Romanic in type, with English somewhere in between. Therefore, the little overlap in vocabulary might make it hard for French, Italian, and Spanish writers to understand and use German foreign words.

3) Little improvement for several source words

The right column in Table 1 shows the scores if (using the product-of-ranks algorithm) four source languages are taken into account in parallel. As can be seen, with an average score of 51.8 the improvement over the English only variant (50.6) is minimal. This contrasts with the findings described in Rapp & Zock (2010) where significant improvements could be achieved by increasing the number of source languages. So this casts some doubt on these. However, as English was not considered as a source language there, the performance levels were mostly between 10 and 20, leaving much room for improvement. This is not the case here, where we try to improve on a score of around 50 for English. Remember that this is a somewhat conservative score as we count correct but alternative translations, as errors. As this is already a performance much closer to the optimum, making further performance gains is more difficult. Therefore, perhaps we should take it as a success that the product-of-ranks algorithm could achieve a minimal performance gain despite the fact that the influence of the non-English languages was probably mostly detrimental.

Having analyzed the quantitative results, to give a better impression of the strengths and weaknesses of our algorithm, for the (according to Table 1) best performing combination of corpus and language pair, namely the French WaCky corpus, English as the source language and French as the target language, Table 2 shows some actual source words and their computed translations.

ESW	CF	ET	RE	CT
cabbage	9	chou	1	chou
blossom	25	fleur	73	commande
carpet	39	tapis	1	tapis
bitter	59	amer	1	amer
hammer	67	marteau	1	marteau
bread	82	pain	1	pain
citizen	115	citoyen	1	citoyen
bath	178	bain	1	bain
butterfly	201	papillon	1	papillon
eat	208	manger	1	manger
butter	220	beurre	59	terre
eagle	282	aigle	1	aigle
cheese	527	fromage	1	fromage
cold	539	froid	1	froid
deep	585	profond	1	profond
cottage	624	cabanon	1	cabanon
earth	702	terre	53	tabac
child	735	enfant	1	enfant
bed	806	lit	2	table
beautiful	923	beau	1	beau
care	1267	soin	1	soin
hand	1810	main	2	main
city	2610	ville	1	ville
girl	2673	fille	1	fille
green	2861	vert	1	vert
blue	2914	bleu	1	bleu
hard	3615	dur	1	dur
black	9626	noir	1	noir
Bible	17791	Bible	1	Bible
foot	23548	pied	8	siffler
chair	24027	chaise	1	chaise
fruit	38544	fruit	1	fruit

Table 2: Results for the language pair English → French. The meaning of the columns is as follows: ESW = English source word; CF = corpus frequency of English source word; ET = expected translation according to gold standard; RE = computed rank of expected translation; CT = computed translation.

4 Summary and Future Work

In this paper we made an attempt to solve the difficult problem of identifying word translations on the basis of a single monolingual corpus, whereby the same corpus is used for several language pairs. The basic idea underlying our work is to look at foreign words, to compute their co-occurrence-based associations, and to consider these as translations of the respective words.

Whereas Rapp & Zock (2010) dealt only with an English corpus, the current work shows that this methodology is applicable to a wide range of languages and corpora. We were able to shed some light on criteria influencing performance, such as the selection of text type and the direction of a language pair. For example, it is more promising to look at occurrences of English words in a German corpus rather than the other way around. Because of the special status of English it is also advisable to use it as a pivot wherever possible.

Perhaps surprisingly, the work may have implications regarding cognitive models of second language acquisition. The reason is that it describes how to acquire the vocabulary of a new language from a mixed corpus. This is relevant as traditional foreign language teaching (involving explanations in the native tongue and vocabulary learning using bilingual word lists) can be considered as providing such a mixed corpus.

Regarding future work, let us outline a plan for the construction of a universal dictionary of all languages which are well enough represented on the web.[4] There might be some chance for it, because the algorithm can be extended to work with standard search engines and is also suitable for a bootstrapping approach.

Let us start by assuming that we have a large matrix where the rows correspond to the union of the vocabularies of a considerable number of languages, and the columns correspond to these languages themselves. We presuppose no prior translation knowledge, so that the matrix is completely empty at the beginning (although prior knowledge could be useful for the iterative algorithm to converge).

STEP 1: For each word in the vocabulary we perform a search via a search engine such as Google, preferably in an automated fashion via an application programming interface (API). Next, we retrieve as many documents as possible, and separate them according to language.[5] Then, for each language for which we have obtained the critical mass of documents, we apply our algorithm and compute the respective translations. These are entered into the matrix. As we are interested in word equations, we assume that translations are symmetric. This means that each translation identified can be entered at two positions in the matrix. So at the end of step 1 we have for each word the translations into a number of other languages, but this number may still be small at this stage.

STEP 2: We now look at each row of the matrix and feed the words found within the same row into the product-of-ranks algorithm. We do not have to repeat the Google search, as step 1 already provided all documents needed. Because when looking at several source words we have a better chance to find occurrences in our documents, this should give us translations for some more languages in the same row. But we also need to recompute the translations resulting from the previous step as some of them will be erroneous e.g. for reasons of data sparseness or due to the homograph trap.

STEP 3: Repeat step 2 until as many matrix cells as possible are filled with translations. We hope that with each iteration completeness and correctness improve, and that the process converges in such a way that the (multilingual) words in each row disambiguate each other, so that ultimately each row corresponds to an unambiguous concept.

Acknowledgments

Part of this research was supported by a Marie Curie Intra European Fellowship within the 7th European Community Framework Programme. We thank the WaCky group for making available their excellent Web corpora, and the Linguistic Data Consortium for providing the Gigaword Corpora. We also wish to thank Lourdes Callau, Maria Dolores Jimenez Lopez, and Lilica Voicu for their support in acquiring the LDC corpora.

[4] Note that this plan could also be adapted to other methodologies (such as Rapp, 1999), and may be more promising with these.

[5] If the language identification markup within the retrieved documents turns out to be unreliable (which is unfortunately often the case in practice), standard language identification software can be used.

References

Armstrong, Susan; Kempen, Masja; McKelvie, David; Petitpierre, Dominique; Rapp, Reinhard; Thompson, Henry (1998). Multilingual Corpora for Cooperation. *Proceedings of the 1st International Conference on Linguistic Resources and Evaluation (LREC), Granada,* Vol. 2, 975–980.

Baroni, Marco; Bernardini, Silvia; Ferraresi, Adriano, Zanchetta, Eros (2009). *The WaCky Wide Web: A collection of very large linguistically processed Web-crawled corpora*. Journal of Language Resources and Evaluation 43 (3): 209-226.

Brown, Peter; Cocke, John; Della Pietra, Stephen A.; Della Pietra, Vincent J.; Jelinek, Frederick; Lafferty, John D.; Mercer, Robert L.; Rossin, Paul S. (1990). A statistical approach to machine translation. *Computational Linguistics*, 16(2), 79–85.

Chiao, Yun-Chuang; Sta, Jean-David; Zweigenbaum, Pierre (2004). A novel approach to improve word translations extraction from non-parallel, comparable corpora. In: *Proceedings of the International Joint Conference on Natural Language Processing*, Hainan, China. AFNLP.

Denoyer, Ludovic; Gallinari, Pattrick (2006). The Wikipedia XML Corpus. *SIGIR Forum*, 40(1), 64–69.

Dunning, T. (1993). Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics*, 19(1), 61–74.

Ferraresi, Adriano; Bernardini, Silvia; Picci, Giovanni; Baroni, Marco (2010). Web corpora for bilingual lexicography: a pilot study of English/ French collocation extraction and translation. In Xiao, Richard (ed.): *Using Corpora in Contrastive and Translation Studies*. Newcastle: Cambridge Scholars Publishing.

Fung, Pascale; McKeown, Kathy (1997). Finding terminology translations from non-parallel corpora. *Proceedings of the 5th Annual Workshop on Very Large Corpora,* Hong Kong, 192-202.

Fung, Pascale; Yee, Lo Yuen (1998). An IR approach for translating new words from nonparallel, comparable texts. In: *Proceedings of COLING-ACL 1998,* Montreal, Vol. 1, 414-420.

Harris, Zelig S. (1954). Distributional structure. *WORD*, 10:146–162.

Kent, Grace Helen; Rosanoff , A.J. (1910). A study of association in insanity. American Journal of Insanity 67:317–390.

Koehn, Philipp (2005). Europarl: A parallel corpus for statistical machine translation. *Proceedings of MT Summit*, Phuket, Thailand, 79–86.

Koehn, Philipp; Hoang, Hieu; Birch, Alexandra; Callison-Burch, Chris; Federico, Marcello; Bertoldi, Nicola; Cowan, Brooke; Shen, Wade; Moran, Christine; Zens, Richard; Dyer, Chris; Bojar, Ondřej; Constantin, Alexandra; Herbst, Evan (2007). Moses: Open source toolkit for statistical machine translation. In: *Proceedings of ACL*, Prague, demonstration session, 177–180.

Koehn, Philipp; Knight, Kevin (2002). Learning a translation lexicon from monolingual corpora. In: *Unsupervised Lexical Acquisition. Proceedings of the ACL SIGLEX Workshop*, 9–16.

Mendonça, Angelo, Graff, David, DiPersio, Denise (2009a). *French Gigaword Second Edition.* Linguistic Data Consortium, Philadelphia.

Mendonça, Angelo, Graff, David, DiPersio, Denise (2009b). *Spanish Gigaword Second Edition.* Linguistic Data Consortium, Philadelphia.

Munteanu, Dragos Stefan; Marcu, Daniel (2005). Improving machine translation performance by exploiting non-parallel corpora. *Computational Linguistics* 31(4), 477–504.

Och, Franz Josef; Ney, Hermann (2003). A Systematic Comparison of Various Statistical Alignment Models. *Computational Linguistics*, 29(1), 19–51.

Pantel, Patrick; Lin, Dekang (2002). Discovering word senses from text. In: *Proceedings of ACM SIGKDD*, Edmonton, 613–619

Rapp, Reinhard (1995). Identifying word translations in non-parallel texts. In: *Proceedings of the 33rd Meeting of the Association for Computational Linguistics.* Cambridge, Massachusetts, 320-322.

Rapp, Reinhard. (1999). Automatic identification of word translations from unrelated English and German corpora. In: *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics 1999,* College Park, Maryland. 519–526.

Rapp, Reinhard; Martín Vide, Carlos (2007). Statistical machine translation without parallel corpora. In: Georg Rehm, Andreas Witt, Lothar Lemnitzer (eds.): *Data Structures for Linguistic Resources and Applications. Proceedings of the Biennial GLDV Conference 2007*. Tübingen: Gunter Narr Verlag. 231–240.

Rapp, Reinhard; Zock, Michael (2010). Utilizing Citations of Foreign Words in Corpus-Based Dictionary Generation. *Proceedings of NLPIX 2010.*

Sharoff, Serge (2006). Creating general-purpose corpora using automated search engine queries. In Marco Baroni and Silvia Bernardini (eds.): WaCky! *Working papers on the Web as Corpus*. Gedit, Bologna, http://wackybook.sslmit.unibo.it/

Steinberger, Ralf; Pouliquen, Bruno; Widiger, Anna; Ignat, Camelia; Erjavec, Tomaž; Tufiş, Dan; Varga, Dániel (2006). The JRC-Acquis: A multilingual aligned parallel corpus with 20+ languages. *Proceedings of the 5th International Conference on Language Resources and Evaluation* **(***LREC 2006***)***.* Genoa, Italy.

Wu, Dekai; Fung, Pascale (2005). Inversion transduction grammar constraints for mining parallel sentences from quasi-comparable corpora. Proceedings of the *Second International Joint Conference on Natural Language Processing (IJCNLP-2005)*. Jeju, Korea.

Appendix: Gold Standard of 100 Word Equations

	ENGLISH	GERMAN	FRENCH	SPANISH	ITALIAN
1	anger	Wut	colère	furia	rabbia
2	baby	Baby	bébé	bebé	bambino
3	bath	Bad	bain	baño	bagno
4	beautiful	schön	beau	hermoso	bello
5	bed	Bett	lit	cama	letto
6	Bible	Bibel	Bible	Biblia	Bibbia
7	bitter	bitter	amer	amargo	amaro
8	black	schwarz	noir	negro	nero
9	blossom	Blüte	fleur	flor	fiore
10	blue	blau	bleu	azul	blu
11	boy	Junge	garçon	chico	ragazzo
12	bread	Brot	pain	pan	pane
13	butter	Butter	beurre	mantequilla	burro
14	butterfly	Schmetterling	papillon	mariposa	farfalla
15	cabbage	Kohl	chou	col	cavolo
16	care	Pflege	soin	cuidado	cura
17	carpet	Teppich	tapis	alfombra	tappeto
18	chair	Stuhl	chaise	silla	sedia
19	cheese	Käse	fromage	queso	formaggio
20	child	Kind	enfant	niño	bambino
21	citizen	Bürger	citoyen	ciudadano	cittadino
22	city	Stadt	ville	ciudad	città
23	cold	kalt	froid	frío	freddo
24	command	Kommando	commande	comando	comando
25	convenience	Bequemlichkeit	commodité	conveniencia	convenienza
26	cottage	Häuschen	cabanon	casita	casetta
27	dark	dunkel	foncé	oscuro	buio
28	deep	tief	profond	profundo	profondo
29	doctor	Arzt	médecin	médico	medico
30	dream	Traum	rêve	sueño	sogno
31	eagle	Adler	aigle	águila	aquila
32	earth	Erde	terre	tierra	terra
33	eat	essen	manger	comer	mangiare
34	foot	Fuß	pied	pie	piede
35	fruit	Frucht	fruit	fruta	frutta
36	girl	Mädchen	fille	chica	ragazza
37	green	grün	vert	verde	verde
38	hammer	Hammer	marteau	martillo	martello
39	hand	Hand	main	mano	mano
40	handle	Griff	poignée	manejar	maniglia
41	hard	hart	dur	duro	duro
42	head	Kopf	tête	cabeza	testa
43	health	Gesundheit	santé	salud	salute
44	heavy	schwer	lourd	pesado	pesante

45	high	hoch	élevé	alto	alto
46	house	Haus	maison	casa	casa
47	hungry	hungrig	affamé	hambriento	affamato
48	joy	Freude	joie	alegría	gioia
49	justice	Gerechtigkeit	justice	justicia	giustizia
50	King	König	roi	rey	re
51	lamp	Lampe	lampe	lámpara	lampada
52	light	Licht	lumière	luz	luce
53	lion	Löwe	lion	león	leone
54	long	lang	long	largo	lungo
55	loud	laut	fort	alto	alto
56	man	Mann	homme	hombre	uomo
57	memory	Gedächtnis	mémoire	memoria	memoria
58	moon	Mond	lune	luna	luna
59	mountain	Berg	montagne	montaña	montagna
60	music	Musik	musique	música	musica
61	mutton	Hammel	mouton	cordero	montone
62	needle	Nadel	aiguille	aguja	ago
63	nervous	nervös	nerveux	nervioso	nervoso
64	ocean	Ozean	océan	océano	oceano
65	oven	Backofen	four	horno	forno
66	priest	Priester	prêtre	sacerdote	sacerdote
67	quick	schnell	rapide	rápido	rapido
68	quiet	still	tranquille	tranquilo	tranquillo
69	red	rot	rouge	rojo	rosso
70	religion	Religion	religion	religión	religione
71	river	Fluss	rivière	río	fiume
72	rough	rau	rugueux	áspero	ruvido
73	salt	Salz	sel	sal	sale
74	scissors	Schere	ciseaux	tijeras	forbici
75	sheep	Schaf	mouton	oveja	pecora
76	short	kurz	courte	corto	corto
77	sickness	Krankheit	maladie	enfermedad	malattia
78	sleep	schlafen	sommeil	sueño	dormire
79	slow	langsam	lent	lento	lento
80	smooth	glatt	lisse	liso	liscio
81	soft	weich	doux	suave	morbido
82	soldier	Soldat	soldat	soldado	soldato
83	sour	sauer	acide	agrio	acido
84	spider	Spinne	araignée	araña	ragno
85	square	Quadrat	carré	cuadrado	quadrato
86	stomach	Magen	estomac	estómago	stomaco
87	street	Straße	rue	calle	strada
88	sweet	süß	doux	dulce	dolce
89	table	Tisch	table	mesa	tavolo
90	thief	Dieb	voleur	ladrón	ladro
91	thirsty	durstig	soif	sediento	assetato
92	tobacco	Tabak	tabac	tabaco	tabacco
93	whisky	Whisky	whisky	whisky	whisky
94	whistle	pfeifen	siffler	silbar	fischiare
95	white	weiß	blanc	blanco	bianco
96	window	Fenster	fenêtre	ventana	finestra
97	wish	Wunsch	désir	deseo	desiderio
98	woman	Frau	femme	mujer	donna
99	work	arbeiten	travail	trabajo	lavoro
100	yellow	gelb	jaune	amarillo	giallo

Multi-Word Expression-Sensitive Word Alignment

Tsuyoshi Okita[1], **Alfredo Maldonado Guerra**[2], **Yvette Graham**[3], **Andy Way**[1]
{CNGL[1], NCLT[3]} / School of Computing / Dublin City University,
CNGL / School of Computer Science and Statistics / Trinity College Dublin[2]
{tokita,ygraham,away}@computing.dcu.ie, maldonaa@scss.tcd.ie

Abstract

This paper presents a new word alignment method which incorporates knowledge about Bilingual Multi-Word Expressions (BMWEs). Our method of word alignment first extracts such BMWEs in a bidirectional way for a given corpus and then starts conventional word alignment, considering the properties of BMWEs in their grouping as well as their alignment links. We give partial annotation of alignment links as prior knowledge to the word alignment process; by replacing the maximum likelihood estimate in the M-step of the IBM Models with the Maximum A Posteriori (MAP) estimate, prior knowledge about BMWEs is embedded in the prior in this MAP estimate. In our experiments, we saw an improvement of 0.77 Bleu points absolute in JP–EN. Except for one case, our method gave better results than the method using only BMWEs grouping. Even though this paper does not directly address the issues in Cross-Lingual Information Retrieval (CLIR), it discusses an approach of direct relevance to the field. This approach could be viewed as the opposite of current trends in CLIR on semantic space that incorporate a notion of order in the bag-of-words model (e.g. co-occurences).

1 Introduction

Word alignment (Brown et al., 1993; Vogel et al., 1996; Och and Ney, 2003a; Graca et al., 2007) remains key to providing high-quality translations as all subsequent training stages rely on its performance. It alone does not effectively capture many-to-many word correspondences, but instead relies on the ability of subsequent heuristic phrase extraction algorithms, such as grow-diag-final (Koehn et al., 2003), to resolve them.

Some aligned corpora include implicit partial alignment annotation, while for other corpora a partial alignment can be extracted by state-of-the-art techniques. For example, implicit tags such as reference number within the patent corpus of Fujii et al. (2010) provide (often many-to-many) correspondences between source and target words, while statistical methods for extracting a partial annotation, like Kupiec et al. (1993), extract terminology pairs using linguistically predefined POS patterns. Gale and Church (1991) extract pairs of anchor words, such as numbers, proper nouns (organization, person, title), dates, and monetary information. Resnik and Melamed (1997) automatically extract domain-specific lexica. Moore (2003) extracts named-entities. In Machine Translation, Lambert and Banchs (2006) extract BMWEs from a phrase table, which is an outcome of word alignment followed by phrase extraction; this method does not alter the word alignment process.

This paper introduces a new method of incorporating previously known many-to-many word correspondences into word alignment. A well-known method of incorporating such prior knowledge in Machine Learning is to replace the likelihood maximization in the M-step of the EM algorithm with either the MAP estimate or the Maximum Penalized Likelihood (MPL) estimate (McLach-

Proceedings of the 4th International Workshop on Cross Lingual Information Access at COLING 2010, pages 26–34,
Beijing, August 2010

lan and Krishnan, 1997; Bishop, 2006). Then, the MAP estimate allows us to incorporate the *prior*, a probability used to reflect the degree of prior belief about the occurrences of the events.

A small number of studies have been carried out that use partial alignment annotation for word alignment. Firstly, Graca et al. (2007) introduce a posterior regularization to employ the prior that cannot be easily expressed over model parameters such as stochastic constraints and agreement constraints. These constraints are set in the E-step to discard intractable alignments contradicting these constraints. This mechanism in the E-step is in a similar spirit to that in GIZA++ for IBM Model 3 and 4 which only searches around neighbouring alignments around the Viterbi alignment. For this reason, this algorithm is not intended to be used combined with IBM Models 3 and 4. Although theoretically it is possible to incorporate partial annotation with a small change in its code, Graca et al. do not mention it. Secondly, Talbot (2005) introduces a constrained EM method which constrains the E-step to incorporate partial alignment into word alignment,[1] which is in a similar manner to Graca et al. (2007). He conducted experiments using partial alignment annotation based on cognate relations, a bilingual dictionary, domain-specific bilingual semantic annotation, and numerical pattern matching. He did not incorporate BMWEs. Thirdly, Callison-Burch et al. (2004) replace the likelihood maximization in the M-step with mixed likelihood maximization, which is a convex combination of negative log likelihood of known links and unknown links.

The remainder of this paper is organized as follows: in Section 2 we define the anchor word alignment problem. In Section 3 we include a review of the EM algorithm with IBM Models 1-5, and the HMM Model. Section 4 describes our own algorithm based on the combination of BMWE extraction and the modified word alignment which incorporates the groupings of BMWEs and enforces their alignment links; we explain the EM algorithm with MAP estimation with three kinds of priors. In Section 5 our experimental results are presented, and we conclude in Section 6.

pair	GIZA++(no prior)			Ours(with prior)		
EN-FR	fin	ini	prior	fin	ini	prior
is *NULL*	1	.25	0	0	.25	.25
rosy *en*	1	.5	0	0	.5	.2
that .	1	.25	0	0	.25	.25
life *la*	1	.25	0	0	.25	0
. *c‘*	1	.25	0	0	.25	.25
that *c‘*	0	.25	0	1	.25	.25
is *est*	0	.25	0	1	.25	.25
life *vie*	0	.5	0	1	.5	1
rosy *rose*	0	.25	0	1	.25	.2

Table 1: The benefit of prior knowledge of anchor words.

2 Anchor Word Alignment Problem

The input to standard methods of word alignment is simply the sentence-aligned corpus, whereas our alignment method takes in additionally a partial alignment. We assume, therefore, the availability of a partial alignment, for example via a MWE extraction tool. Let $\breve{e}$ denote an English sentence, and e denote an English word, throughout this paper. The anchor word alignment problem is defined as follows:

Definition 1 (Anchor Word Alignment Problem) Let $(\breve{e}, \breve{f}) = \{(\breve{e}_1, \breve{f}_1), \ldots, (\breve{e}_n, \breve{f}_n)\}$ be a parallel corpus. By prior knowledge we additionally have knowledge of anchor words $(\hat{e}, \hat{f}) = \{(sent_i, t_{e_1}, t_{f_1}, pos_{e_1}, pos_{f_1}, length_e, length_f),$ $\ldots,$ $(sent_k, t_{e_n}, t_{f_n}, pos_{e_n}, pos_{f_n}, length_e, length_f)\}$ where $sent_i$ denotes sentence ID, pos_{e_i} denotes the position of t_{e_i} in a sentence $\breve{e}_i$, and $length_e$ (and $length_f$) denotes the sentence length of the original sentence which includes e_i. Under a given $(\breve{e}, \breve{f})$ and $(\hat{e}, \hat{f})$, our objective is to obtain word alignments. It is noted that an anchor word may include a phrase pair which forms n-to-m mapping objects.

Table 1 shows two example phrase pairs for French to English *c’est la vie* and *that is life*, and *la vie en rose* and *rosy life* with the initial value for the EM algorithm, the prior value and the fi-

[1] Although the code may be similar in practice to our Prior Model I, his explanation to modify the E-step will not be applied to IBM Models 3 and 4. Our view is to modify the M-step due to the same reason above, i.e. GIZA++ searches only over the alignment space around the Viterbi alignment.

<table>
<tr><th>Statistical MWE extraction method</th></tr>
<tr><td>97|||groupe_socialiste|||socialist_group|||26|||26
101|||monsieur_poettering|||mr_poettering|||1|||4
103|||monsieur_poettering|||mr_poettering|||1|||11
110|||monsieur_poettering|||mr_poettering|||1|||9
117|||explication_de_vote|||explanation_of_vote|||28|||26</td></tr>
<tr><td>Heuristic-based MWE extraction method</td></tr>
<tr><td>28|||the_wheel_2|||車輪_２ ||| 25||| 5
28|||the_primary-side_fixed_armature_13|||１_次_側_固定_電機_子_１_３ ||| 13||| 9
28|||the_secondary-side_rotary_magnet_7|||２_次_側_回転_マグネット_７ ||| 15||| 11</td></tr>
</table>

Table 2: Example of MWE pairs in Europarl corpus (FR-EN) and NTCIR patent corpus (JP-EN). There are 5 columns for each term: sentence number, source term, target term, source position, and target position. The number appended to each term from the patent corpus (lower half) is a reference number. In this corpus, all the important technical terms have been identified and annotated with reference numbers.

nal lexical translation probability for Giza++ IBM Model 4 and that of our modified Giza++. Our modified Giza++ achieves the correct result when anchor words 'life' and '*vie*' are used to assign a value to the prior in our model.

3 Word Alignment

We review two models which address the problem of word alignment. The aim of word alignment is to obtain the model parameter t among English and French words, e_i and f_j respectively. We search for this model parameter under some model $\mathcal{M}$ where $\mathcal{M}$ is chosen by IBM Models 1-5 and the HMM model. We introduce the latent variable a, which is an alignment function with the hypothesis that each e and f correspond to this latent variable. (e, f, a) is a complete data set, and (e, f) is an incomplete data set.

3.1 EM Algorithm

We follow the description of the EM algorithm for IBM Models of Brown et al. (1993) but introduce the parameter t explicitly. In this model, the parameter t represents the lexical translation probabilities $t(e_i|f_j)$. It is noted that we use $e|f$ rather than $f|e$ following the notation of Koehn (2010). One important remark is that the Viterbi alignment of the sentence pair $(\breve{e}, \breve{f}) = (e_1^J, f_1^I)$, which is obtained as in (1):

$$\mathbf{E}^{\mathbf{viterbi}}: \quad \hat{a_1^J} = \arg\max_{a_1^J} p_{\hat{\theta}}(f, a|e) \qquad (1)$$

provides the best alignment for a given log-likelihood distribution $p_{\hat{\theta}}(f, a|e)$. Instead of summing, this step simplifies the E-step. However, under our modification of maximum likelihood estimate with MAP estimate, this simplification is not a correct approximation of the summation since our surface in the E-step is greatly perturbed by the prior. There is no guarantee that the Viterbi alignment is within the proximity of the target alignment (cf. Table 1).

Let z be the latent variable, t be the parameters, and x be the observations. The EM algorithm is an iterative procedure repeating the E-step and the M-step as in (2):

$$\begin{aligned} \mathbf{E}^{\mathbf{EXH}}: \quad & q(z; x) = p(z|x; \theta) & (2) \\ \mathbf{M}^{\mathbf{MLE}}: \quad & t' = \arg\max_t Q(t, t^{old}) \\ & \quad = \arg\max_t \sum_{x,z} q(z|x) \log p(x, z; t) \end{aligned}$$

In the E-step, our knowledge of the values of the latent variables in a is given only by the posterior distribution $p(a|e, f, t)$. Hence, the (negative log)-likelihood of complete data (e, f, a), which we denote by $-\log p(t|e, f, a)$, is obtained over all possible alignments a. We use the current parameter values t^{old} to find the posterior distribution of the latent variables given by $p(a|e, f, t^{old})$. We then use this posterior distribution to find the expectation of the complete data log-likelihood evaluated for parameter value t. This expectation is given by $\sum_a p(a|e, f, t^{old}) \log p(e, f, a|t)$.

In the M-step, we use a maximal likelihood estimation to minimize negative log-likelihood in order to determine the parameter t; note that t is a lexical translation probability. Instead of using the log-likelihood $\log p(a, e, f|t)$, we use the expected complete data log-likelihood over all the possible alignments a that we obtained in the E-step, as in (3):

$$\mathrm{M}^{\mathbf{MLE}}: \quad t' = \arg\max_t Q(t, t^{old}) \qquad (3)$$
$$= \frac{c(f|e;f,e)}{\sum_e c(f|e;f,e)}$$

where an auxiliary function $c(e|f;e,f)$ for IBM Model 1 introduced by Brown et al. is defined as

$$c(f|e;f,e) = \sum_a p(a|e,f) \sum_{j=1}^{m} \delta(f, f_j)\delta(e, e_{a_j})$$

and where the Kronecker-Delta function $\delta(x, y)$ is 1 if $x = y$ and 0 otherwise. This auxiliary function is convenient since the normalization factor of this count is also required. We note that if we use the MAP estimate, the E-step remains the same as in the maximum likelihood case, whereas in the M-step the quantity to be minimized is given by $Q(t, t^{old}) + \log p(t)$. Hence, we search for the value of t which maximizes the following equation:

$$\mathrm{M}^{\mathbf{MAP}}: \quad t' = \arg\max_t Q(t, t^{old}) + \log p(t)$$

3.2 HMM

A first-order Hidden Markov Model (Vogel et al., 1996) uses the sentence length probability $p(J|I)$, the mixture alignment probability $p(i|j, I)$, and the translation probability, as in (4):

$$p(f|e) = p(J|I) \prod_{j=1}^{J} p(f_j|e_i) \qquad (4)$$

Suppose we have a training set of R observation sequences X_r, where $r = 1, \cdots, R$, each of which is labelled according to its class m, where $m = 1, \cdots, M$, as in (5):

$$p(i|j, I) = \frac{r(i - j\frac{I}{J})}{\sum_{i'=1}^{I} r(i' - j\frac{I}{J})} \qquad (5)$$

The HMM alignment probabilities $p(i|i', I)$ depend only on the jump width $(i - i')$. Using a set of non-negative parameters $s(i - i')$, we have (6):

$$p(i|i', I) = \frac{s(i - i')}{\sum_{l=1}^{I} s(l - i')} \qquad (6)$$

4 Our Approach

Algorithm 1 Overall Algorithm

```
Given: a parallel corpus,
1. Extract MWEs by Algorithm 2.
2. Based on the results of Step 1, specify a set of anchor word alignment links in the format of anchor word alignment problem (cf. Definition 1 and Table 2).
3. Group MWEs in source and target text.
4. Calculate the prior in order to embed knowledge about anchor words.
5. Calculate lexical translation probabilities with the prior.
6. Obtain alignment probabilities.
7. Ungroup of MWEs in source and target text.
```

Algorithm 1 consists of seven steps. We use the Model I prior for the case where our prior knowledge is sparse and evenly distributed throughout the corpus, whereas we use the Model II prior when our prior knowledge is dense in a partial corpus. A typical example of the former case is when we use partial alignment annotation extracted throughout a corpus for bilingual terminology. A typical example of the latter case is when a sample of only a few hundred lines from the corpus have been hand-annotated.

4.1 MWE Extraction

Our algorithm of extracting MWEs is a statistical method which is a bidirectional version of Kupiec (1993). Firstly, Kupiec presents a method to extract bilingual MWE pairs in a unidirectional manner based on the knowledge about typical POS patterns of noun phrases, which is language-dependent but can be written down with some ease by a linguistic expert. For example in French they are N N, N prep N, and N Adj. Secondly, we take the intersection (or union) of extracted bilingual MWE pairs.[2]

[2] In word alignment, bidirectional word alignment by taking the intersection or union is a standard method which improves its quality compared to unidirectional word alignment.

Algorithm 2 MWE Extraction Algorithm

Given: a parallel corpus and a set of anchor word alignment links:

1. We use a POS tagger (Part-Of-Speech Tagger) to tag a sentence on the SL side.

2. Based on the typical POS patterns for the SL, extract noun phrases on the SL side.

3. Count n-gram statistics (typically $n = 1, \cdots, 5$ are used) on the TL side which jointly occur with each source noun phrase extracted in Step 2.

4. Obtain the maximum likelihood counts of joint phrases, i.e. noun phrases on the SL side and n-gram phrases on the TL side.

5. Repeat the same procedure from Step 1 to 4 reversing the SL and TL.

6. Intersect (or union) the results in both directions.

Let SL be the source language side and TL be the target language side. The procedure is shown in Algorithm 2. We informally evaluated the MWE extraction tool following Kupiec (1993) by manually inspecting the mapping of the 100 most frequent terms. For example, we found that 93 of the 100 most frequent English terms in the patent corpus were correctly mapped to their Japanese translation.

Depending on the corpus, we can use more prior knowledge about implicit alignment links. For example in some categories of patent and technical documents corpora,[3] we can use heuristics to extract the "noun phrase" + "reference number" from both sides. This is due to the fact that terminology is often labelled with a unique reference number, which is labelled on both the SL and TL sides.

4.2 Prior Model I

Prior for Exhaustive Alignment Space IBM Models 1 and 2 implement a prior for all possible alignments exhaustively. Such a prior requires the following two conditions. Firstly, partial knowledge about the prior that we use in our context is defined as follows. Let us denote a bilingual term list $T = \{(s_1, t_1), \ldots, (s_m, t_m)\}$. For example with IBM Model 1: Let us define the following prior $p(e|f, e, f; T)$ from Equation (4):

Algorithm 3 Prior Model I for IBM Model 1

```
Given: parallel corpus $\breve{e}$, $\breve{f}$,
    anchor words $biTerm$
initialize t($e|f$) uniformly
do until convergence
 set count($e|f$) to 0 for all e,f
 set total(f) to 0 for all f
 for all sentence pairs ($\breve{e}_s$,$\breve{f}_s$)
    prior($e|f$)$_s$ = getPriorModelI($\breve{e}, \breve{f}, biTerm$)
 for all words e in $\breve{e}_s$
    $total_s$(e) = 0
    for all words f in $\breve{f}_s$
       $total_s$(e) += t($e|f$)
 for all words e in $\breve{e}_s$
    for all words f in $\breve{f}_s$
       count($e|f$)+=$t(e|f)/total_s(e)\times prior(e|f)_s$
       total(f) += $t(e|f)/total_s(e) \times prior(e|f)_s$
 for all f
    for all e
       t($e|f$) = $count(e|f)/total(f)$
```

$$p(e|f, e, f; T) = \begin{cases} 1 & (e_i = s_i, f_j = t_j) \\ 0 & (e_i = s_i, f_j \neq t_j) \\ 0 & (e_i \neq s_i, f_j = t_j) \\ \text{uniform} & (e_i \neq s_i, f_j \neq t_j) \end{cases}$$

Secondly, this prior should be proper for the exhaustive case and non-proper for the sampled alignment space where by proper we mean that the probability is normalized to 1. Algorithm 3 shows the pseudo-code for Prior Model I. Note that if the prior is uniform in the MAP estimation, this is equivalent to maximum likelihood estimation.

Prior for Sampled Alignment (Function) Space Due to the exponential costs introduced by fertility, null token insertion, and distortion probability, IBM Models 3 and 4 do not consider all $(I + 1)^J$ alignments exhaustively, but rather a small subset in the E-step. Each iteration only uses the subset of all the alignment functions: this sampling

[3] Unlike other language pairs, the availability of Japanese–English parallel corpora is quite limited: the NTCIR patent corpus (Fujii et al., 2010) of 3 million sentence pairs (the latest NTCIR-8 version) for the patent domain and JENAAD corpus (Utiyama and Isahara, 2003) of 150k sentence pairs for the news domain. In this regard, the patent domain is particularly important for this particular language pair.

is not uniform, as it only includes the best possible alignment with all its neighbouring alignments which differ from the best alignment by one word (this can be corrected by a move operation) or two words (this can be corrected by a swap operation).

If we consider the neighbouring alignment via a move or a swap operation, two issues arise. Firstly, the fact that these two neighbouring alignments are drawn from different underlying distributions needs to be taken into account, and secondly, that the application of a move and a swap operation alters a row or column of a prior matrix (or indices of the prior) since either operation involves the manipulation of links.

Algorithm 4 Pseudo-code for Prior Model II Exhaustive Alignment Space

```
def getPriorModelII($\breve{e}$,$\breve{f}$,biTerm):
for i in sentence:
    for e in $\breve{e}_i$:
        allWords$_i$ = length of sentence $\breve{e}$
        for f in $\breve{f}_i$:
            if $(e, f)$ in biTerm:
                n= num of anchor words in $i$
                uni$(e|f)_i$ = $\frac{\text{allWords}_i - \text{n}}{\text{allWords}_i}$
                expSum$(e|f)$ += uni$(e|f)_i$ $\times$ n
            else:
                countSum$(e|f)_i$ += n
countSum$(e|f)$ += count$(e|f)_i$
for e in $all_e$:
    for f in $all_f$:
        $prior(e|f)$ = expSum$(e|f)$ + countSum$(e|f)$
return $prior(e|f)$
```

Prior for Jump Width i' One implementation of HMM is to use the forward-backward algorithm. A prior should be embedded within the forward-backward algorithm. From Equation (6), there are three cases which depend on whether a_i and its neighbouring alignment a_{i-1} are determined by our prior knowledge about anchor words or not. When both a_i and a_j are determined, this probability is expressed as in (7):

$$p(i-i';I) = \begin{cases} 0 & (else) \\ 1 & (e_i = s_i, f_j = t_j \text{ for } a_i) \text{ and } \\ & (e'_i = s'_i, f'_j = t'_j \text{ for } a_j) \end{cases} \tag{7}$$

When either a_i or a_j is determined, this probability is expressed as in (8):[4]

$$p(i-i';I) = \begin{cases} 0 & (\text{condition } 1) \\ 1 & (\text{condition } 2) \\ \frac{1}{(m-\#e_{a_i}-\cdots-\#e_{a_i+m})} & (else) \\ \text{(uniform distribution)} & \end{cases} \tag{8}$$

When neither a_i nor a_j is determined, this probability is expressed as in (9): [5]

$$p(i-i';I) = \begin{cases} 0 & (\text{condition } 3) \\ 1 & (\text{condition } 4) \\ \frac{m-i'}{(m-\#e_{a_i}-\cdots-\#e_{a_i+m})^2} & (else) \\ \text{(Pascal's triangle distribution)} & \end{cases} \tag{9}$$

4.3 Prior Model II

Prior Model II assumes that we have prior knowledge only in some part of the training corpus. A typical example is when a small part of the corpus has a hand-crafted 'gold standard' annotation.

Prior for Exhaustive Alignment Space Prior Model II is used to obtain the prior probability $p(e|f)$ over all possible combinations of e and f. In contrast to Prior Model I, which computes the prior probability $p(e|f)$ for each sentence, Prior Model II computes the prior probability globally for all sentences in the corpus. Algorithm 4 shows the pseudo-code for Prior Model II Exhaustive Alignment Space.

[4] condition 1 is as follows:

$((e_i \neq s_i, f_j \neq t_j$ for $a_i)$ and $(e'_i = s'_i, f'_j = t'_j$ for $a_j))$ or
$((e_i \neq s_i, f_j \neq t_j$ for $a_i)$ and $(e'_i = s'_i, f'_j = t'_j$ for $a_j))$ or
$((e_i = s_i, f_j = t_j$ for $a_i)$ and $(e'_i \neq s'_i, f'_j \neq t'_j$ for $a_j))$ or
$((e_i = s_i, f_j = t_j$ for $a_i)$ and $(e'_i \neq s'_i, f'_j \neq t'_j$ for $a_j))$

'condition 2' is as follows:

$((e_i = s_i, f_j \neq t_j$ for $a_i)$ and $(e'_i = s'_i, f'_j = t'_j$ for $a_j))$ or
$((e_i \neq s_i, f_j = t_j$ for $a_i)$ and $(e'_i = s'_i, f'_j = t'_j$ for $a_j))$ or
$((e_i = s_i, f_j = t_j$ for $a_i)$ and $(e'_i \neq s'_i, f'_j = t'_j$ for $a_j))$ or
$((e_i = s_i, f_j = t_j$ for $a_i)$ and $(e'_i = s'_i, f'_j \neq t'_j$ for $a_j))$

[5] 'condition 3' is as follows:

$((e_i \neq s_i, f_j \neq t_j$ for $a_i)$ and $(e'_i \neq s'_i, f'_j \neq t'_j$ for $a_j))$

'condition 4' is as follows:

$((e_i \neq s_i, f_j \neq t_j$ for $a_i)$ and $(e'_i \neq s'_i, f'_j = t'_j$ for $a_j))$ or
$((e_i \neq s_i, f_j \neq t_j$ for $a_i)$ and $(e'_i = s'_i, f'_j \neq t'_j$ for $a_j))$ or
$((e_i \neq s_i, f_j = t_j$ for $a_i)$ and $(e'_i \neq s'_i, f'_j \neq t'_j$ for $a_j))$ or
$((e_i = s_i, f_j \neq t_j$ for $a_i)$ and $(e'_i \neq s'_i, f'_j \neq t'_j$ for $a_j))$

Prior for Sampled Alignment (Function) Space This is identical to that of the Prior Model II exhaustive alignment space with only a difference in the normalization process.

Prior for Jump Width i' This categorization of Prior Model II is the same as that of Prior Model I for for Jump Width i' (see Section 4.2). Note that Prior Model II requires more memory compared to the Prior Model I.[6]

5 Experimental Settings

The baseline in our experiments is a standard log-linear phrase-based MT system based on Moses. The GIZA++ implementation (Och and Ney, 2003a) of IBM Model 4 is used as the baseline for word alignment, which we compare to our modified GIZA++. Model 4 is incrementally trained by performing 5 iterations of Model 1, 5 iterations of HMM, 5 iterations of Model 3, and 5 iterations of Model 4. For phrase extraction the grow-diag-final heuristics are used to derive the refined alignment from bidirectional alignments. We then perform MERT while a 5-gram language model is trained with SRILM. Our implementation is based on a modified version of GIZA++ (Och and Ney, 2003a). This modification is on the function that reads a bilingual terminology file, the function that calculates priors, the M-step in IBM Models 1-5, and the forward-backward algorithm in the HMM Model. Other related software tools are written in Python and Perl: terminology concatenation, terminology numbering, and so forth.

6 Experimental Results

We conduct an experimental evaluation on the NTCIR-8 corpus (Fujii et al., 2010) and on Europarl (Koehn, 2005). Firstly, MWEs are extracted from both corpora, as shown in Table 3. In the second step, we apply our modified version of GIZA++ in which we incorporate the results of MWE extraction. Secondly, in order to incorporate the extracted MWEs, they are reformatted as shown in Table 2. Thirdly, we convert all MWEs into a single token, i.e. we concatenate them with an underscore character. We then run the modified version of GIZA++ and obtain a phrase and reordering table. In the fourth step, we split the concatenated MWEs embedded in the third step. Finally, in the fifth step, we run MERT, and proceed with decoding before automatically evaluating the translations.

corpus	language	size	#unique MWEs	#all MWEs
statistical method				
NTCIR	EN-JP	200k	1,121	120,070
europarl	EN-FR	200k	312	22,001
europarl	EN-ES	200k	406	16,350
heuristic method				
NTCIR	EN-JP	200k	50,613	114,373

Table 3: Statistics of our MWE extraction method. The numbers of MWEs are from 0.08 to 0.6 MWE / sentence pair in our statistical MWE extraction methods.

Table 4 shows the results where 'baseline' indicates no BMWE grouping nor prior, and 'baseline2' represents a BMWE grouping but without the prior. Although 'baseline2' (BMWE grouping) shows a drop in performance in the JP–EN / EN–JP 50k sentence pair setting, Prior Model I results in an increase in performance in the same setting. Except for EN–ES 200k, our Prior Model I was better than 'baseline2'. For EN–JP NTCIR using 200k sentence pairs, we obtained an absolute improvement of 0.77 Bleu points compared to the 'baseline'; for EN–JP using 50k sentence pairs, 0.75 Bleu points; and for ES–EN Europarl corpus using 200k sentence pairs, 0.63 Bleu points. In contrast, Prior Model II did not work well. The possible reason for this is the misspecification, i.e. the modelling by IBM Model 4 was wrong in terms of the given data. One piece of evidence for this is that most of the enforced alignments were found correct in a manual inspection.

For EN–JP NTCIR using the same corpus of 200k, although the number of unique MWEs ex-

[6] This is because it needs to maintain potentially an $\ell \times m$ matrix, where ℓ denotes the number of English tokens in the corpus and m denotes the number of foreign tokens, even if the matrix is sparse. Prior Model I only requires an $\hat{\ell} \times \hat{m}$ matrix where $\hat{\ell}$ is the number of English tokens in a sentence and $\hat{m}$ is the number of foreign tokens in a sentence, which is only needed until this information is incorporated in a posterior probability during the iterative process.

size	EN-JP	Bleu	JP-EN	Bleu
50k	baseline	16.33	baseline	22.01
50k	baseline2	16.10	baseline2	21.71
50k	prior I	17.08	prior I	22.11
50k	prior II	16.02	prior II	20.02
200k	baseline	23.42	baseline	21.68
200k	baseline2	24.10	baseline2	22.32
200k	prior I	24.22	prior I	22.45
200k	prior II	23.22	prior II	21.00

size	FR-EN	Bleu	EN-FR	Bleu
50k	baseline	17.68	baseline	17.80
50k	baseline2	17.76	baseline2	18.00
50k	prior I	17.81	prior I	18.02
50k	prior II	17.01	prior II	17.30
200k	baseline	18.40	baseline	18.20
200k	baseline2	18.80	baseline2	18.50
200k	prior I	18.99	prior I	18.60
200k	prior II	18.20	prior II	17.50

size	ES-EN	Bleu	EN-ES	Bleu
50k	baseline	16.21	baseline	15.17
50k	baseline2	16.61	baseline2	15.60
50k	prior I	16.91	prior I	15.87
50k	prior II	16.15	prior II	14.60
200k	baseline	16.87	baseline	17.62
200k	baseline2	17.40	baseline2	18.21
200k	prior I	17.50	prior I	18.20
200k	prior II	16.50	prior II	17.10

Table 4: Results. Baseline is plain GIZA++ / Moses (without BMWE grouping / prior), baseline2 is with BMWE grouping, prior I / II are with BMWE grouping and prior.

tracted by the statistical method and the heuristic method varies significantly, the total number of MWEs by each method becomes comparable. The resulting Bleu score for the heuristic method (24.24 / 22.48 Blue points for 200k EN–JP / JP–EN) is slightly better than that of the statistical method. The possible reason for this is related to the way the heuristic method groups terms including reference numbers, while the statistical method does not. As a result, the complexity of the alignment model simplifies slightly in the case of the heuristic method.

7 Conclusion

This paper presents a new method of incorporating BMWEs into word alignment. We first detect BMWEs in a bidirectional way and then use this information to do groupings and to enforce already known alignment links. For the latter process, we replace the maximum likelihood estimate in the M-step of the EM algorithm with the MAP estimate; this replacement allows the incorporation of the prior in the M-step of the EM algorithm. We include an experimental investigation into incorporating extracted BMWEs into a word aligner. Although there is some work which incorporates BMWEs in groupings, they do not enforce alignment links.

There are several ways in which this work can be extended. Firstly, although we assume that our a priori partial annotation is reliable, if we extract such MWEs automatically, we cannot avoid erroneous pairs. Secondly, we assume that the reason why our Prior Model II did not work was due to the misspecification (or wrong modelling). We would like to check this by discriminative modelling. Thirdly, although here we extract BMWEs, we can extend this to extract paraphrases and non-literal expressions.

8 Acknowledgments

This research is supported by the Science Foundation Ireland (Grant 07/CE/I1142) as part of the Centre for Next Generation Localisation (`http://www.cngl.ie`) at Dublin City University and Trinity College Dublin. We would also like to thank the Irish Centre for High-End Computing.

References

Bishop, Christopher M. 2006. *Pattern Recognition and Machine Learning*. Springer. Cambridge, UK

Brown, Peter F., Vincent .J.D Pietra, Stephen A.D.Pietra, Robert L. Mercer. 1993. *The Mathematics of Statistical Machine Translation: Parameter Estimation*. Computational Linguistics. 19(2), pp. 263–311.

Callison-Burch, Chris, David Talbot and Miles Osborne. 2004. *Statistical Machine Translation with*

Word- and Sentence-Aligned Parallel Corpora. Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL'04), Main Volume. Barcelona, Spain, pp. 175–182.

Fujii, Atsushi, Masao Utiyama, Mikio Yamamoto, Takehito Utsuro, Terumasa Ehara, Hiroshi Echizen-ya, Sayori Shimohata. 2010. *Overview of the Patent Translation Task at the NTCIR-8 Workshop*. Proceedings of the 8th NTCIR Workshop Meeting on Evaluation of Information Access Technologies: Information Retrieval, Question Answering and Cross-lingual Information Access, pp. 293–302.

Graca, Joao de Almeida Varelas, Kuzman Ganchev, Ben Taskar. 2007. *Expectation Maximization and Posterior Constraints*. In Neural Information Processing Systems Conference (NIPS), Vancouver, BC, Canada, pp. 569–576.

Gale, William, and Ken Church. 1991. *A Program for Aligning Sentences in Bilingual Corpora*. In Proceedings of the 29th Annual Meeting of the Association for Computational Linguistics. Berkeley CA, pp. 177–184.

Koehn, Philipp, Franz Och, Daniel Marcu. 2003. *Statistical Phrase-Based Translation*. In Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics. Edmonton, Canada. pp. 115–124.

Koehn, Philipp. 2005. *Europarl: A Parallel Corpus for Statistical Machine Translation*. In Conference Proceedings: the tenth Machine Translation Summit. Phuket, Thailand, pp.79-86.

Koehn, Philipp, 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, 2007. *Moses: Open source toolkit for Statistical Machine Translation*. Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions, Prague, Czech Republic, pp. 177–180.

Koehn, Philipp. 2010. *Statistical Machine Translation*. Cambridge University Press. Cambridge, UK.

Kupiec, Julian. 1993. *An Algorithm for finding Noun Phrase Correspondences in Bilingual Corpora.* In Proceedings of the 31st Annual Meeting of Association for Computational Linguistics. Columbus. OH. pp. 17–22.

Lambert, Patrik and Rafael Banchs. 2006. *Grouping Multi-word Expressions According to Part-Of-Speech in Statistical Machine Translation*. In Proceedings of the EACL Workshop on Multi-Word-Expressions in a Multilingual Context. Trento, Italy, pp. 9–16.

McLachlan, Geoffrey J. and Thriyambakam Krishnan, 1997. *The EM Algorithm and Extensions*. Wiley Series in probability and statistics. New York, NY.

Moore, Robert C.. 2003. *Learning Translations of Named-Entity Phrases from Parallel Corpora*. In Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics. Budapest, Hungary. pp. 259–266.

Moore, Robert C.. 2004. *On Log-Likelihood-Ratios and the Significance of Rare Events*. In Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing (EMNLP). Barcelona, Spain, pp. 333–340.

Och, Franz and Herman Ney. 2003. *A Systematic Comparison of Various Statistical Alignment Models*. Computational Linguistics. 29(1), pp. 19–51.

Resnik, Philip and I. Dan Melamed, 1997. *Semi-Automatic Acquisition of Domain-Specific Translation Lexicons*. Proceedings of the 5th Applied Natural Language Processing Conference. Washington, DC., pp. 340–347.

Talbot, David. 2005. *Constrained EM for parallel text alignment*, Natural Language Engineering, 11(3): pp. 263–277.

Utiyama, Masao and Hitoshi Isahara. 2003. *Reliable Measures for Aligning Japanese-English News Articles and Sentences*, In Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics. Sapporo, Japan, pp. 72–79.

Vogel, Stephan, Hermann Ney, Christoph Tillmann 1996. *HMM-Based Word Alignment in Statistical Translation*. In Proceedings of the 16th International Conference on Computational Linguistics. Copenhagen, Denmark, pp. 836–841.

Co-occurrence Graph Based Iterative Bilingual Lexicon Extraction From Comparable Corpora

Diptesh Chatterjee and **Sudeshna Sarkar** and **Arpit Mishra**
Department of Computer Science and Engineering
Indian Institute of Technology Kharagpur
{diptesh,sudeshna,arpit}@cse.iitkgp.ernet.in

Abstract

This paper presents an iterative algorithm for bilingual lexicon extraction from comparable corpora. It is based on a bag-of-words model generated at the level of sentences. We present our results of experimentation on corpora of multiple degrees of comparability derived from the FIRE 2010 dataset. Evaluation results on 100 nouns shows that this method outperforms the standard context-vector based approaches.

1 Introduction

Bilingual dictionaries play a pivotal role in a number of Natural Language Processing tasks like Machine Translation and Cross Lingual Information Retrieval(CLIR). Machine Translation systems often use bilingual dictionaries in order to augment word and phrase alignment (Och and Ney, 2003). CLIR systems use bilingual dictionaries in the query translation step (Grefenstette, 1998). However, high coverage electronic bilingual dictionaries are not available for all language pairs. So a major research area in Machine Translation and CLIR is bilingual dictionary extraction. The most common approach for extracting bilingual dictionary is applying some statistical alignment algorithm on a parallel corpus. However, parallel corpora are not readily available for most language pairs. Also, it takes a lot of effort to actually get the accurate translations of sentences. Hence, constructing parallel corpora involves a lot of effort and time. So in recent years, extracting bilingual dictionaries from comparable corpora has become an important area of research. Comparable corpora consist of documents on similar topics in different languages. Unlike parallel corpora, they are not sentence aligned. In fact, the sentences in one language do not have to be the exact translations of the sentence in the other language. However, the two corpora must be on the same domain or topic. Comparable corpora can be obtained more easily than parallel corpora. For example, a collection of news articles from the same time period but in different languages can form a comparable corpora. But after careful study of news articles in English and Hindi published on same days at the same city, we have observed that along with articles on similar topics, the corpora also contain a lot of articles which have no topical similarity. Thus, the corpora are quite noisy, which makes it unsuitable for lexicon extraction. Thus another important factor in comparable corpora construction is the degree of similarity of the corpora.

Approaches for lexicon extraction from comparable corpora have been proposed that use the bag-of-words model to find words that occur in similar lexical contexts (Rapp, 1995). There have been approaches proposed which improve upon this model by using some linguistic information (Yuu and Tsujii, 2009). However, these require some linguistic tool like dependency parsers which are not commonly obtainable for resource-poor languages. For example, in case of Indian languages like Hindi and Bengali, we still do not have good enough dependency parsers. In this paper, we propose a word co-occurrence based approach for lexicon extraction from comparable corpora using English and Hindi as the source and target languages respectively. We do not use any language-

Proceedings of the 4th International Workshop on Cross Lingual Information Access at COLING 2010, pages 35–42,
Beijing, August 2010

specific resource in our approach.

We did experiments with 100 words in English,and show that our approach performs significantly better than the the Context Heterogeneity approach (Fung, 1995). We show the results over corpora with varying degrees of comparability.

The outline of the paper is as follows. In section 2, we analyze the different approaches for lexicon extraction from comparable corpora. In section 3, we present our algorithm and the experimental results. In section 4, we present an analysis of the results followed by the conclusion and future research directions in section 5.

2 Previous Work

One of the first works in the area of comparable corpora mining was based on word co-occurrence based approach (Rapp, 1995). The basic assumption behind this approach was two words are likely to occur together in the same context if their joint probability of occurrence in a corpus exceeds the probability that the words occur randomly. In his paper, Rapp made use of a similarity matrix and using a joint probability estimate determined the word maps. However this approach did not yield significantly good results.

The "Context Heterogeneity" approach was one of the pioneering works in this area. It uses a 2-dimensional context vector for each word based on the right and left context. The context vector depended on how many distinct words occur in the particular context and also the unigram frequency of the word to be translated. Euclidean distance between context vectors was used as a similarity measure.

Another approach used Distributed Clustering of Translational Equivalents for word sense acquisition from bilingual comparable corpora (Kaji, 2003). However, the major drawback of this paper is the assumption that translation equivalents usually represent only one sense of the target word. This may not be the case for languages having similar origin, for example, Hindi and Bengali.

Approaches using context information for extracting lexical translations from comparable corpora have also been proposed (Fung and Yee, 1998; Rapp, 1999). But they resulted in very poor coverage. These approaches were improved upon by extracting phrasal alignments from comparable corpora using joint probability SMT model (Kumano et al., 2007) .

Another proposed method uses dependency parsing and Dependency Heterogeneity for extracting bilingual lexicon (Yuu and Tsujii, 2009) . This approach was similar to that of Fung, except they used a dependency parser to get the tags for each word and depending on the frequency of each tag they defined a vector to represent each word in question. Here too, Euclidean similarity was used to compute the similarity between two words using their context vectors. However, this method is dependent on availability of a dependency parser for the languages and is not feasible for languages for which resources are scarce.

3 Bilingual Dictionary Extraction Using Co-occurrence Information

3.1 Motivation

The Context Heterogeneity and Dependency Heterogeneity approaches suffer from one major drawback. They do not use any kind of information about how individual words combine in a particular context to form a meaningful sentence. They only use some statistics about the number of words that co-occur in a particular context or the number of times a word receives a particular tag in dependency parsing. So, we wished to study if the quality of dictionary extracted would improve if we consider how individual words co-occur in text and store that information in the form of a vector, with one dimension representing one word in the corpus. One important point to note here is that the function words in a language are usually very small in number. If we need to construct a dictionary of function words in two languages, that can be done without much effort manually. Also, the function words do not play an important role in CLIR applications, as they are usually stripped off.

Our algorithm is based on the intuition that words having similar semantic connotations occur together. For example, the words "bread" is more likely to occur with "eat" than with "play". Our algorithm uses this distribution of co-occurrence frequency along with a small initial seed dictio-

nary to extract words that are translations of one another. We define a co-occurrence vector of words in both the languages, and also record the number of times two words co-occur. To find the translation for word W_x, we check for the words co-occurring with W_x such that this word already has a map in the other language, and compute a scoring function using all such words co-occurring with W_x. In short, we use the already existing information to find new translations and add them to the existing lexicon to grow it. Below is a snapshot of a part of the data from one of our experiments using the FIRE 2010[1] corpus. For each word in English and Hindi, the co-occurrence data is expressed as a list of tuples. Each tuple has the form **(word, co-occurrence frequency)**. For the Hindi words, the English meaning has been provided in parenthesis. For the seed lexicon and final lexicon, the format is **(source word, target word, strength)**.
English:

1. **teacher**:{(training,49),(colleges,138), (man,22)}
2. **car**:{(drive,238),(place,21)}
3. **drive**:{(car,238),(steer,125),(city,12), (road,123)}

Hindi:

1. **ghar(home)**:{(khidki(window),133),(makAn (house),172), (rAstA(road),6)}
2. **gAdi(car)**:{(rAsta,92),(chAlak(driver),121), (signal,17)}
3. **shikshaka(teacher)**:{(vidyalaya(school),312), (makAn(house),6)}

Seed lexicon:

1. (colleges,vidyalaya,0.4)
2. (colleges,mahavidyalaya(college),0.6)
3. (car,gAdi,1.0)

The following is a snapshot from the final results given by the algorithm:

1. (car,gAdi,1.0)
2. (teacher,shikshak,0.62)
3. (teacher, vidyalaya,0.19)
4. (road, rAsta, 0.55)

3.2 The Algorithm

For extracting bilingual lexicon, we have not considered the function words of the two languages. In order to filter out the function words, we have made use of the assumption that content words usually have low frequency in the corpus, whereas function words have very high frequency. First, we define some quantities:

Let the languages be **E** and **H**.
W_e = Set of words in **E** $= \{e_1, e_2, ..., e_N\}$
W_h = Set of words in **H** $= \{h_1, h_2, ..., h_M\}$

$|W_e| = N$
$|W_h| = M$

MAP = Initial map given
$= \{(e_i, h_j, w_{ij}) | w_{ij} = wt(e_i, h_j), e_i \in W_e, h_j \in W_h\}$

E_M = Set of words in **E** which are included in entries of MAP

H_M = Set of words in **H** which are included in entries of MAP

$Co\_occ(x)$ = Set of words which co-occur with word x

$$Co\_occ'(x) = \begin{cases} Co\_occ(x) \cap E_M & \text{if } x \in W_e \\ Co\_occ(x) \cap H_M & \text{if } x \in W_h \end{cases}$$

$Wt_e(x) = \{W_{ey} | y \in W_e \text{ and } y \in Co\_occ(x)\}$
$Wt_h(x) = \{W_{hy} | y \in W_h \text{ and } y \in Co\_occ(x)\}$

Given a comparable corpus, we follow the following steps of processing:

1. A sentence segmentation code is run to segment the corpus into sentences.
2. The sentence-segmented corpus is cleaned of all punctuation marks and special symbols by replacing them with spaces.

[1] Forum For Information Retrieval
http://www.isical.ac.in/∼clia/index.html

Algorithm 1 Algorithm to Extract Bilingual Dictionary by using word Co-occurrence Information

```
repeat
  for $e_i \in W_e$ do
    for $h_j \in W_h$ do
      if $(e_i, h_j, 0) \in MAP$ then
        $wt(e_i, h_j) = \frac{\sum_{e \in Co\_occ'(e_i)} \sum_{h \in Co\_occ'(h_j)} (W_{ij} W_{ee_i} W_{hh_j})}{\sum_{e \in Co\_occ'(e_i)} \sum_{h \in Co\_occ'(h_j)} (W_{ee_i} W_{hh_j})}$
      end if
    end for
  end for
  Select the pair with highest value of $wt(e_i, b_j)$ and add it to the existing map and normalize
until termination
```

3. The collection frequency of all the terms are computed and based on a threshold, the function words are filtered out.

4. The co-occurrence information is computed at sentence-level for the remaining terms. In a sentence, if words w_i and w_j both occur, then $w_i \in Co\_occ(w_j)$ and vice versa.

5. Since we can visualize the co-occurrence information in the form of a graph, we next cluster the graph into C clusters.

6. From each cluster C_i, we choose some fixed number number of words and manually find out their translation in the target language. This constitutes the initial map.

7. Next we apply Algorithm 1 to compute the word maps.

The time complexity of the algorithm is $O(IM^2N^2)$, where I is the number of iterations of the algorithm.

3.3 Corpus Construction

The corpora used for evaluating our algorithm were derived from the FIRE 2010 English and Hindi corpora for the ad-hoc retrieval task. These corpora contained news articles spanning over a time period of three years from two Indian newspapers, "The Dainik Jagaran" in Hindi and "The Telegraph" in English. However, due to the extreme level of variation of the topics in these corpora, we applied a filtering algorithm to select a subset of the corpora.

Our approach to make the text similar involved reducing the corora based on matching Named Entities. Named Entities of English and Hindi corpus were listed using LingPipe[2] and a Hindi NER system built at IIT Kharagpur(Saha et al., 1999). The listed Named Entities of the two corpora were compared to find the matching Named Entities. Named Entities in Hindi Unicode were converted to iTRANS[3] format and matched with English Named Entities using edit distance. Unit cost was defined for each insert and delete operation. Similar sounding characters like 's', 'c','a', 'e' etc were assigned a replacement cost of 1 and other characters were assigned a replacement cost of 2. Two Named Entities were adjudged matching if:
$(2 * Cost)/(WL_h + WL_e) < 0.5$
where,
$WL_h =$ Length of Hindi word
$WL_e =$ Length of English word
Using this matching scheme, accuracy of matching of Hindi and English Named Entities was found to be $> 95\%$. It was observed that there are large number of Named Entities with small frequency and few Named Entities with large frequency. So a matching list was prepared which contained only those Named Entities which had frequency larger than a $\sqrt{MaxFreq}$. This ensured that matching list had words with high frequency in both corpus.So English words with frequency larger than 368 and Hindi words with frequency larger than 223 were considered for matching. Based on this matching list, the two

[2] http://alias-i.com/lingpipe/

[3] http://www.aczoom.com/itrans/

Language	Total NE	Unique NE	NE with freq larger than $\sqrt{MaxFreq}$	NE Matched	Total No of docs	% of NE covered	
						According to Zipf's Law	In the actual corpus
Hindi	1195474	37606	686	360	54271	63.0%	74.3%
English	5723292	137252	2258	360	87387	65.2%	71.0%

Table 1: Statistics of the main corpora used for extraction

Corpus	Max Freq Word	Max Freq	$\sqrt{MaxFreq}$
Hindi	bharat	50072	223
English	calcutta	135780	368

Table 2: Criteria used for thresholding in the two corpora

Matching % of NE per document	Total documents in corpora	
	Hindi	English
$> 10\%$	34694	16950
$> 20\%$	14872	4927
$> 30\%$	2938	1650

Table 3: Statistics of extracted corpora

corpora were reduced by including only those files each of which contained more than a certain fixed percentage of total matching Named Entities. The corpus statistics are provided in tables 1, 2 and 3. We assume that distribution of Named Entities follows Zipf's law (Zipf, 1949). And analysis shows that Named Entities with frequency greater than the chosen threshold lead to high coverage both theoretically and in practice (Table 1). Hence, the threshold was chosen as $\sqrt{MaxFreq}$. The differences in the theoretical and actual values can be attributed to the poor performance of the NER systems, especially the Hindi NER system, whose output contained a number of false positives.

3.4 Experimental Setup

The languages we used for our experiments were English and Hindi. English was the source language and Hindi was chosen as the target. For our experiments, we used a collection frequency threshold of 400 to filter out the function words. The words having a collection frequency more than 400 were discarded. This threshold was obtained manually by "Trial and Error" method in order to perform an effective function word filtering. For each corpora, we extracted the co-occurrence information and then clustered the co-occurrence graph into 20 clusters. From each cluster we chose 15 words, thus giving us an overall initial seed dictionary size of 300. We ran the algorithm for 3000 iterations.

For graph clustering, we used the Graclus system (Dhillon et al., 2007) which uses a weighted kernel k-means clustering algorithm at various levels of coarseness of the input graph.

3.5 Evaluation Method and Results

For evaluation, we have used the Accuracy and MMR measure (Voorhees, 1999). The measures are defined as follows:

$$Accuracy = \frac{1}{N}\sum_{i=1}^{N} t_i$$

$$\text{where, } t_i = \begin{cases} 1 & \text{if correct translation in top } n \\ 0 & \text{otherwise} \end{cases}$$

$$MMR = \frac{1}{N}\sum_{i=1}^{N} \frac{1}{rank_i}$$

$$\text{where, } rank_i = \begin{cases} r_i & \text{if } r_i \leq n \\ 0 & \text{otherwise} \end{cases}$$

n means top n evaluation

r_i means rank of correct translation in top n ranking

N means total number of words used for evaluation

For our experiments, we have used:

Corpus	Context Heterogeneity		Co-occurrence	
	Acc	MMR	Acc	MMR
$> 10\%$	0.14	0.112	0.16	0.135
$> 20\%$	0.21	0.205	0.27	0.265
$> 30\%$	0.31	0.285	0.35	0.333

Table 4: Comparison of performance between Context Heterogeneity and Co-occurrence Approach for manual evaluation

$n = 5$
$N = 100$

The 100 words used for evaluation were chosen randomly from the source language.

Two evaluation methods were followed - manual and automated. In the manual evaluation, a person who knows both English and Hindi was asked to find the candidate translation in the target language for the words in the source language. Using this gold standard map, the Accuracy and MMR values were computed.

In the second phase (automated), lexicon extracted is evaluated against English to Hindi wordnet[4]. The evaluation process proceeds as follows:

1. Hashmap is created with English words as keys and Hindi meanings as values.
2. English words in the extracted lexicon are crudely stemmed so that inflected words match the root words in the dictionary. Stemming is done by removing the last 4 characters, one at a time and checking if word found in dictionary.
3. Accuracy and MMR are computed.

As a reference measure, we have used Fung's method of Context Heterogeneity with a context window size of 4. The results are tabulated in Tables 4 and 6. We can see that our proposed algorithm shows a significant improvement over the Context Heterogeneity method. The degree of improvement over the Context Heterogeneity is summarized in Tables 5 and 7. For auto evaluation, We see that the proposed approach shows the maximum improvement (83.33% in Accuracy and 66.67% in MMR) in performance when the corpus size is medium. For very large (too general) corpora, both the approaches give identical result while for very small (too specific) corpora, the proposed approach gives slightly better results than the reference.

The trends are similar for manual evaluation. Once again, the maximum improvement is observed for the medium sized corpus ($> 20\%$). However, in this evaluation system, the proposed approach performs much better than the reference even for the large (more general) corpora.

Corpus	Accuracy	MMR
$> 10\%$	↑ 14.28%	↑ 20.53%
$> 20\%$	↑ 28.57%	↑ 29.27%
$> 30\%$	↑ 12.9%	↑ 16.84%

Table 5: Degree of improvement shown by Co-occurrence approach over Context Heterogeneity for manual evaluation

Corpus	Context Heterogeneity		Co-occurrence	
	Acc	MMR	Acc	MMR
$> 10\%$	0.05	0.08	0.05	0.08
$> 20\%$	0.06	0.06	0.11	0.10
$> 30\%$	0.13	0.11	0.15	0.13

Table 6: Comparison of performance between Context Heterogeneity and Co-occurrence Approach for auto-evaluation

Corpus	Accuracy	MMR
$> 10\%$	0.0%	0.0%
$> 20\%$	↑ 83.33%	↑ 66.67%
$> 30\%$	↑ 15.38%	↑ 18.18%

Table 7: Degree of improvement shown by Co-occurrence approach over Context Heterogeneity for auto-evaluation

[4] Downloadable from http://sanskritdocuments.org/hindi/dict/eng-hin-itrans.html

4 Discussion

The co-occurrence based approach used in this paper is quite a simple approach in the sense that it does not make use of any kind of linguistic information. From the aforementioned results we can see that a model based on simple word co-occurrence highly outperforms the “Context Heterogeneity” model in almost all the cases. One possible reason behind this is the amount of information captured by our model is more than that captured by the “Context Heterogeneity” model. “Context Heterogeneity” does not model actual word-word interactions. Each word is represented by a function of the number of different contexts it can occur in. However, we represent the word by a co-occurrence vector. This captures all possible contexts of the word. Also, we can actually determine which are the words which co-occur with any other word. So our model captures more semantics of the word in question than the “Context Heterogeneity” model, thereby leading to better results. Another possible factor is the nature in which we compute the translation scores. Due to the iterative nature of the algorithm and since we normalize after each iteration, some of the word pairs that received unduly high score in an earlier iteration end up having a substantially low score. However, since the “Context Heterogeneity” does only a single pass over the set of words, it fails to tackle this problem.

The seed dictionary plays an important role in our algorithm. A good seed dictionary gives us some initial information to work with. However, since “Context Heterogeneity” does not use a seed dictionary, it loses out on the amount of information initially available to it. Since the seed dictionary size for our approach is quite small, it can be easily constructed manually. However, how the seed dictionary size varies with corpus size is an issue that remains to be seen.

Another important factor in our algorithm is the way in which we have defined the co-occurrence vectors. This is not the same as the context vector that we define in case of Context Heterogeneity. In a windowed context vector, we fail to capture a lot of dependencies that might be captured using a sentence-level co-occurrence. This problem is especially more visible in case of free-word-order languages like the Indo-European group of languages. For these languages, a windowed context vector is also likely to introduce many spurious dependencies. Since Hindi is a language of this family, our algorithm captures many more correct semantic dependencies than Context Heterogeneity algorithm, resulting in better preformance.

Another strong point of our proposed approach is the closeness of the values of Accuracy and MMR. This shows that the translation candidates extracted by our algorithm are not only correct, but also the best translation candidate gets the highest score with high probability. This is a very important factor in Machine Translation systems, where a more accurate dictionary would give us an improved performance.

A noticeable point about the evaluation scores is the difference in scores given by the automated system and the manual system. This can be attributed to synonymy and spelling errors. In the target language Hindi, synonymy plays a very important part. It is not expected that all synonyms of a particular word may be present in an online dictionary. In such cases, the manual evaluator marks a translation pair as True, whereas the automated system marks it as False. Instances of spelling errors have also been found. For example, for the word ”neighbors”, the top translation provided by the system was ”paDosana”(female neighbor). If we consider root form of words, this is correct. But the actual translation should be ”paDosiyAn”(neighbors, may refer to both male and female). Thus the auto evaluation system tags it as False, whereas the manual evaluator tags it as True. There are many more such occurrences throughout.

Apart from that, the manual evaluation process has been quite relaxed. Even if the properties like tense, number of words does not match, as long as the root forms match the manual evaluator has marked it as True. But this is not the case for the automated evaluator. Although stemming has been done, but problems still persist which can be only solved by lemmatization, because Hindi is a highly inflected language.

5 Conclusion and Future Work

In this paper we present a completely new approach for extracting bilingual lexicon from comparable corpora. We show the results of experimentation on corpora of different levels of comparability. The basic feature of this approach is that it is language independent and needs no additional resource. We could not compare its performance with the Dependency Heterogeneity algorithm due to the lack of resources for Hindi. So this can be taken up as a future work. Also, the algorithm is quite inefficient. Another direction of research can be in trying to explore ways to reduce the complexity of this algorithm. We can also try to incorporate more linguistic information into this model instead of just word cooccurrence. It remains to be seen how these factors affect the performance of the algorithm. Another important question is what should be the size of the seed dictionary for optimum performance of the algorithm. This too can be taken up as a future research direction.

References

Dhillon, I., Y. Guan, and B. Kulis. 2007. Weighted graph cuts without eigenvectors: A multilevel approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 29:11:1944–1957, November.

Fung, Pascale and Lo Yuen Yee. 1998. An ir approach for translating new words from nonparallel, comparable texts. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics / the 17th International Conference on Computational Linguistics*, pages 414–420.

Fung, Pascale. 1995. Compiling bilingual lexicon entries from a non-parallel english-chinese corpus. In *Third Annual Workshop on Very Large Corpora*, Boston, Massachusetts, June.

Grefenstette, G. 1998. The problem of cross-language information retrieval. *Cross-language Information Retrieval*.

Kaji, H. 2003. Word sense acquisition from bilingual comparable corpora. In *Proc. of HLT-NAACL 2003 Main papers*, pages 32–39.

Kumano, T., H. Takana, and T. Tokunaga. 2007. Extracting phrasal alignments from comparable corpora by using joint probability smt model. In *Proc. of TMI*.

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

Rapp, Reinhard. 1995. Identifying word translations in non-parallel texts. In *Proc. of TMI*.

Rapp, Reinhard. 1999. Automatic identification of word translations from unrelated english and german corpora. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*, pages 519–526.

Saha, Sujan Kumar, Sudeshna Sarkar, and Pabitra Mitra. 1999. A hybrid feature set based maximum entropy hindi named entity recognition. In *Proceedings of the Third International Joint Conference on Natural Language Processing*, pages 343–349, Hyderabad, India, January.

Voorhees, E.M. 1999. The trec-8 question answering track report. In *Proceedings of the 8^{th} Text Retrieval Conference*.

Yuu, K. and J. Tsujii. 2009. Extracting bilingual dictionary from comparable corpora with dependency heterogeneity. In *Proc. of NAACL-HLT, short papers*, pages 121–124.

Zipf, George Kingsley. 1949. *Human Behaviour and the Principle of Least Effort: an Introduction to Human Ecology*. Addison-Wesley.

A Voting Mechanism for Named Entity Translation in English–Chinese Question Answering

Ling-Xiang Tang[1], Shlomo Geva[1], Andrew Trotman[2], Yue Xu[1]

[1]Faculty of Science and Technology
Queensland University of Technology
{l4.tang,s.geva,yue.xu}@qut.edu.au

[2]Department of Computer Science
University of Otago
andrew@cs.otago.ac.nz

Abstract

In this paper, we describe a voting mechanism for accurate named entity (NE) translation in English–Chinese question answering (QA). This mechanism involves translations from three different sources: machine translation, online encyclopaedia, and web documents. The translation with the highest number of votes is selected. We evaluated this approach using test collection, topics and assessment results from the NTCIR-8 evaluation forum. This mechanism achieved 95% accuracy in NEs translation and 0.3756 MAP in English–Chinese cross-lingual information retrieval of QA.

1 Introduction

Nowadays, it is easy for people to access multi-lingual information on the Internet. Key term searching on an information retrieval (IR) system is common for information lookup. However, when people try to look for answers in a different language, it is more natural and comfortable for them to provide the IR system with questions in their own natural languages (e.g. looking for a Chinese answer with an English question: "*what is Taiji"*?). Cross-lingual question answering (CLQA) tries to satisfy such needs by directly finding the correct answer for the question in a different language.

In order to return a cross-lingual answer, a CLQA system needs to understand the question, choose proper query terms, and then extract correct answers. Cross-lingual information retrieval (CLIR) plays a very important role in this process because the relevancy of retrieved documents (or passages) affects the accuracy of the answers.

A simple approach to achieving CLIR is to translate the query into the language of the target documents and then to use a monolingual IR system to locate the relevant ones. However, it is essential but difficult to translate the question correctly. Currently, machine translation (MT) can achieve very high accuracy when translating general text. However, the complex phrases and possible ambiguities present in a question challenge general purpose MT approaches. Out-of-vocabulary (OOV) terms are particularly problematic. So the key for successful CLQA is being able to correctly translate all terms in the question, especially the OOV phrases.

In this paper, we discuss an approach for accurate question translation that targets the OOV phrases and uses a translation voting mechanism. This mechanism involves translations from three different sources: machine translation, online encyclopaedia, and web documents. The translation with the highest number of votes is selected. To demonstrate this mechanism, we use Google Translate

Proceedings of the 4th International Workshop on Cross Lingual Information Access at COLING 2010, pages 43–51,
Beijing, August 2010

(GT)[1] as the MT source, Wikipedia as the encyclopaedia source, and Google web search engine to retrieve Wikipedia links and relevant Web document snippets.

English questions on the Chinese corpus for CLQA are used to illustrate of this approach. Finally, the approach is examined and evaluated in terms of translation accuracy and resulting CLIR performance using the test collection, topics and assessment results from NTCIR-8[2].

<table>
<tr><th>English Question Templates (QTs)</th></tr>
<tr><td>who [is | was | were | will], what is the definition of, what is the [relationship | interrelationship | inter-relationship] [of | between], what links are there, what link is there, what [is | was | are | were | does | happened], when [is | was | were | will | did | do], where [will | is | are | were], how [is | was | were | did], why [does | is | was | do | did | were | can | had], which [is | was | year], please list, describe [relationship | interrelationship | inter-relationship] [of | between], could you [please | EMPTY] give short description[s] to, who, where, what, which, how, describe, explain</td></tr>
<tr><th>Chinese QT Counterparts</th></tr>
<tr><td>之间有什么关系，的定义是什么，的关系是什么，发生了什么事，是什么关系, 是什么时候，的关系如何，之间有什么，请简短简述，请简单简述，什么时候，什么关系, 的关系是, 有何关系，关系如何，有何相关，有何渊源，为什么会，为什么要，为什么能，是哪一年，什么时候，位于哪里，什么样的，你能不能，相互之间, 代表什么，简短简述，简单简述，简短描述，简单描述，为什么, 是什么，什么是，的关系，在哪里，怎么样, 有哪些，什么事，是哪个，是哪家，有什么，请列出，请列举, 请描述, 哪一年，请简述, 能不能, 的定义, 何时, 谁是，是谁, 如何，哪个，列举，请问，何谓，何以，为何，描述，有何，简述，哪些，什么，之间，有关, 定义，解释</td></tr>
</table>

Table 1. Question templates

2 CLIR Issue and Related Work

In CLIR, retrieving documents with a cross-lingual query with out-of-vocabulary phrases has always been difficult. To resolve this problem, an external resource such as Web or Wikipedia is often used to discover the possible translation for the OOV term. Wikipedia and other Web documents are thought of as treasure troves for OOV problem solving because they potentially cover the most recent OOV terms.

The Web-based translation method was shown to be an effective way to solve the OOV phrase problem (Chen et al., 2000; Lu et al., 2007; Zhang & Vines, 2004; Zhang et al., 2005). The idea behind this method is that a term/phrase and its corresponding translation normally co-exist in the same document because authors often provide the new terms' translation for easy reading.

In Wikipedia the language links provided for each entry cover most popular written languages, therefore, it was used to solve a low coverage issue on named entities in EuroWordNet (Ferrández et al., 2007); a number of research groups (Chan et al., 2007; Shi et al., 2008; Su et al., 2007; Tatsunori Mori, 2007) employed Wikipedia to tackle OOV problems in the NTCIR evaluation forum.

3 CLQA Question Analysis

Questions for CLQA can be very complex. For example, "*What is the relationship between the movie "Riding Alone for Thousands of Miles" and ZHANG Yimou?*". In this example, it is important to recognise two named entities (*"Riding Alone for Thousands of Miles"* and *"ZHANG Yimou"*) and to translate them precisely.

In order to recognise the NEs in the question, first, English question template phrases in Table 1 are removed from question; next, we use the Stanford NLP POS tagger (The Stanford Natural Language Processing Group, 2010) to identify the named entities; then translate them accordingly. Chinese question template phrases are also pruned from the translated question at the end to reduce the noise words in the final query.

There are three scenarios in which a term or phrase is considered a named entity. First, it is consecutively labelled NNP or NNPS (University of Pennsylvania, 2010). Second, term(s) are grouped by quotation marks. For example, to extract a named entity from the example question above, three steps are needed:

1. Remove the question template phrase "*What is the relationship between*" from the question.
2. Process the remaining using the POS tagger, giving *"the_DT movie_NN ``_`` Riding_NNP Alone_NNP for_IN Thou-*

[1] http://translate.google.com.

[2] http://research.nii.ac.jp/ntcir/ntcir-ws8/ws-en.html.

sands_NNS of_IN Miles_NNP ``_``and_CC ZHANG_NNP Yimou_NNP ?_.”

3. *“Riding Alone for Thousands of Miles”* is between two tags (``) and so is an entity, and the phrase “*ZHANG Yimou”*, as indicated by two consecutive NNP tags is also a named entity.

Third, if a named entity recognised in the two scenarios above is followed in the question by a phrase enclosed in bracket pairs, this phrase will be used as a *tip term* providing additional information about this named entity. For instance, in the question “*Who is David Ho (Da-i Ho)?”*, “*Da-i Ho”* is the tip term of the named entity “*David Ho*”.

4 A Voting Mechanism for Named Entity Translation (VMNET)

Observations have been made:

- Wikipedia has over 100,000 Chinese entries describing various up-to-date events, people, organizations, locations, and facts. Most importantly, there are links between English articles and their Chinese counterparts.
- When people post information on the Internet, they often provide a translation (where necessary) in the same document. These pages contain bilingual phrase pairs. For example, if an English term/phrase is used in a Chinese article, it is often followed by its Chinese translation enclosed in parentheses.
- A web search engine such as Google can identify Wikipedia entries, and return popular bi-lingual web document snippets that are closely related to the query.
- Statistical machine translation relying on parallel corpus such as Google Translate can achieve very high translation accuracy.

Given these observations, there could be up to three different sources from which we can obtain translations for a named entity; the task is to find the best one.

4.1 VMNET Algorithm

A Google search on the extracted named entity is performed to return related Wikipedia links and bilingual web document snippets. Then from the results of Web search and MT, three different translations could be acquired.

Wikipedia Translation

The Chinese equivalent Wikipedia pages could be found by following the language links in English pages. The title of the discovered Chinese Wikipedia page is then used as the *Wikipedia translation*.

Bilingual Clue Text Translation

The Chinese text contained in the snippets returned by the search engine is processed for *bilingual clue text translation*. The phrase in a different language enclosed in parentheses which come directly after the named entity is used as a candidate translation. For example, from a web document snippet, “*YouTube - Sean Chen (陳信安) dunks on Yao Ming…*”, “*陳信安*” can be extracted and used as a candidate translation of “*Sean Chen*”, who is a basket ball player from Taiwan.

Machine Translation

In the meantime, translations for the named entity and its tip term (if there is one) are also retrieved using Google Translate.

Regarding the translation using Wikipedia, the number of results could be more than one because of ambiguity. So for a given named entity, we could have at least one, but possibly more than three candidate translations.

With all possible candidate translations, the best one then can be selected. Translations from all three sources are equally weighted. Each translation contributes one vote, and the votes for identical translation are cumulated. The best translation is the one with the highest number of votes. In the case of a tie, the first choice of the best translation is the Wikipedia translation if only one Wiki-entry is found; otherwise, the priority for choosing the best is bilingual clue text translation, then machine translation.

4.2 Query Generation with VMNET

Because terms can have multiple meanings, ambiguity often occurs if only a single term is given in machine translation. A state-of-the-art MT toolkit/service could perform better if more contextual information is provided. So a better translation is possible if the whole sentence is given (e.g. the question). For this reason, the machine translation of the question is the whole query and not with the templates removed.

However, issues arise: 1) how do we know if all the named entities in question are translated correctly? 2) if there is an error in named entity translation, how can it be fixed? Particularly for case 2, the translation for the whole question is considered acceptable, except for the named entity translation part. We intend to keep most of the translation and replace the bad named entity translation with the good one. But finding the incorrect named entity translation is difficult because the translation for a named entity can be different in different contexts. The missing boundaries in Chinese sentences make the problem harder. To solve this, when a translation error is detected, the question is reformatted by replacing all the named entities with some nonsense strings containing special characters as place holders. These place holders remain unchanged during the translation process. The good NE translations then can be put back for the nearly translated question.

Given an English question Q, the detailed steps for the Chinese query generation are as following:

1. Retrieve machine translation T_{mt} for the whole question from Google Translate.
2. Remove question template phrase from question.
3. Process the remaining using the POS tagger.
4. Extract the named entities from the tagged words using the method discussed in Section 3.
5. Replace each named entity in question Q with a special string S_i,$(i =0,1,2,..)$ which makes nonsense in translation and is formed by a few non-alphabet characters. In our experiments, S_i is created by joining a double quote character with a ^ character and the named entity *id* (a number, starting from 0, then increasing by 1 in order of occurrence of the named entity) followed by another double quote character. The final S_i, becomes **"^*id*"**. The resulting question is used as Q_s.
6. Retrieve machine translation T_{qs} for Q_s from Google Translate. Since S_i consists of special characters, it remains unchanged in T_{qs}.
7. Start the VMNET loop for each named entity.
8. With an option set to return both English and Chinese results, Google the named entity and its tip term (if there is one).
9. If there are any English Wikipedia links in the top 10 search results, then retrieve them all. Else, jump to step 12.
10. Retrieve all the corresponding Chinese Wikipedia articles by following the languages links in the English pages. If none, then jump to step 12.
11. Save the title $NET_{wiki}(i)$ of each Chinese Wikipedia article *Wiki(i)*.
12. Process the search results again to locate a bilingual clue text translation candidate - NET_{ct}, as discussed in Section 4.1.
13. Retrieve machine translation NET_{mt}, and NET_{tip} for this named entity and its tip term (if there is one).
14. Gather all candidate translations: $NET_{wiki}(*)$, NET_{ct}, NET_{tip}, and NET_{mt} for voting. The translation with the highest number of votes is considered the best (NET_{best}). If there is a tie, NET_{best} is then assigned the translation with the highest priority. The priority order of candidate translation is $NET_{wiki}(0)$ *(if sizeof(*$NET_{wiki}(*)$*)=1)* $> NET_{ct} > NET_{mt}$. It means when a tie occurs and if there are more than one Wikipedia translation, all the Wikipedia translations are skipped.
15. If T_{mt} does not contain NET_{best}, it is then considered a faulty translation.
16. Replace S_i in T_{qs} with NET_{best}.
17. If NET_{best} is different from any $NET_{wiki}(i)$ but can be found in the content of a Wikipedia article (*Wiki(i)*), then the corresponding $NET_{wiki}(i)$ is used as an additional query term, and appended to the final Chinese query.
18. Continue the VMNET loop and jump back to step 8 until no more named entities remain in the question.
19. If T_{mt} was considered a faulty translation, use T_{qs} as the final translation of Q. Otherwise, just use T_{mt}. The Chinese question template phrases are pruned from the translation for the final query generation.

A short question translation example is given below:

- For the question “*What is the relationship between the movie "Riding Alone for Thousands of Miles" and ZHANG Yimou?*”, retrieving its Chinese translation from a MT service, we get the following: 之间有什么电影“利民为千里单独的关系”和张艺谋.
- The translation for the movie name *"Riding Alone for Thousands of Miles"* of “*ZHANG Yimou”* is however incorrect.
- Since the question is also reformatted into “*What is the relationship between the movie "^0" and “^1”?*”, machine translation returns a second translation: 什么是电影之间的关系“^ 0”和“^ 1”？
- VMNET obtains the correct translations: 千里走单骑 and 张艺谋, for two named entities *"Riding Alone for Thousands of Miles"* and “*ZHANG Yimou”* respectively.
- Replace the place holders with the correct translations in the second translation and give the final Chinese translation：什么是电影之间的关系“千里走单骑”和“张艺谋”？

5 Information Retrieval

5.1 Chinese Document Processing

Approaches to Chinese text indexing vary: Unigrams, bigrams and whole words are all commonly used as tokens. The performance of various IR systems using different segmentation algorithms or techniques varies as well (Chen et al., 1997; Robert & Kwok, 2002). It was seen in prior experiments that using an indexing technique requiring no dictionary can have similar performance to word-based indexing (Chen, et al., 1997). Using bigrams that exhibit high mutual information and unigrams as index terms can achieve good results. Motivated by indexing efficiency and without the need for Chinese text segmentation, we use both bigrams and unigrams as indexing units for our Chinese IR experiments.

5.2 Weighting Model

A slightly modified BM25 ranking function was used for document ordering.

When calculating the inverse document frequency, we use:

$$IDF(q_i) = log\frac{N}{n} \quad (1)$$

where ***N*** is the number of documents in the corpus, and ***n*** is the document frequency of query termq_i. The retrieval status value of a document ***d*** with respect to query $q(q_1, \dots, q_m)$ is given as:

$$rsv(q,d) =$$

$$\sum_{i=0}^{m} \frac{tf(q_i,d) * (k_1 + 1)}{tf(q_i,d) + k_1 * \left(1 - b + b * \frac{len(d)}{avgdl}\right)} * IDF(q_i) \quad (2)$$

where $tf(q_i, d)$ is the term frequency of term q_i in document ***d***; $len(d)$ is the length of document ***d*** in words and *avgdl* is the mean document length. The number of bigrams is included in the document length. The values of the tuneable parameters k_1 and b used in our experiments are 0.7 and 0.3 respectively.

6 CLIR Experiment

6.1 Test Collection and Topics

Table 2 gives the statistics of the test collection and the topics used in our experiments. The collection contains 308,845 documents in simplified Chinese from *Xinhua News*. There are in total 100 topics consisting of both English and Chinese questions. This is a NTCIR-8 collection for ACLIA task.

Corpus	#docs	#topics
Xinhua Chinese (simplified)	308,845	100

Table 2. Statistics of test corpus and topics

6.2 Evaluation Measures

The evaluation of VMNET performance covers two main aspects: translation accuracy and CLIR performance.

As we focus on named entity translation, the translation accuracy is measured using the precision of translated named entities at the topic level. So the translation precision -*P* is defined as:

$$P = \frac{c}{N} \quad (3)$$

where *c* is the number of topics in which all the named entities are correctly translated; *N* is the number of topics evaluated.

The effectiveness of different translation methods can be further measured by the resulting CLIR performance. In NTCIR-8, CLIR performance is measured using the mean average precision. The MAP values are obtained by running the ir4qa_eval2 toolkit with the assessment results [3] on experimental run s(NTCIR Project, 2010). MAP is computed using only 73 topics due to an insufficient number of relevant document found for the other 27 topics (Sakai et al., 2010). This is the case for all NTCIR-8 ACLIA submissions and not our decision.

It also must be noted that there are five topics that have misspelled terms in their English questions. The misspelled terms in those 5 topics are given in Table 3. It is interesting to see how different translations cope with misspelled terms and how this affects the CLIR result.

Topic ID	Misspelling	Correction
ACLIA2-CS-0024	*Qingling*	*Qinling*
ACLIA2-CS-0035	*Initials D*	*Initial D*
ACLIA2-CS-0066	*Kasianov*	*Kasyanov*
ACLIA2-CS-0074	*Northern Territories*	*northern territories*
ACLIA2-CS-0075	*Kashimir*	*Kashmir*

Table 3. The misspelled terms in topics

6.3 CLIR Experiment runs

A few experimental runs were created for VMNET and CLIR system performance evaluation. Their details are listed in Table 7. Those with name *CS-CS* are the Chinese monolingual IR runs; and those with the name *EN-CS* are the English-to-Chinese CLIR runs. Mono-lingual IR runs are used for benchmarking our CLIR system performance.

7 Results and Discussion

7.1 Translation Evaluation

The translations in our experiments using Google Translate reflect only the results retrieved at the time of the experiments because Google Translate is believed to be improved over time.

The result of the final translation evaluation on the 100 topics is given in Table 4. Google Translate had difficulties in 13 topics. If all thirteen named entities in those topics where Google Translate failed are considered OOV terms, the portion of topics with OOV phrases is relatively small. Regardless, there is an 8% improvement achieved by VMNET reaching 95% precision.

Method	c	N	P
Google Translate	87	100	87%
VMNET	95	100	95%

Table 4. Translation Evaluation Results

There are in total 14 topics in which Google Translate or VMNET failed to correctly translate all named entities. These topics are listed in Table 8. Interestingly, for topic (ACLIA2-CS-0066) with the misspelled term “*Kasianov*”, VMNET still managed to find a correct translation (米哈伊尔•米哈伊洛维奇•卡西亚诺夫). This has to be attributed to the search engine’s capability in handling misspellings. On the other hand, Google Translate was correct in its translation of “Northern Territories” of Japan, but VMNET incorrectly chose “Northern Territory” (of Australia). For the rest of the misspelled phrases (*Qingling, Initials D, Kashimir*), neither Google Translate nor VMNET could pick the correct translation.

7.2 IR Evaluation

The MAP values of all experimental runs corresponding to each query processing technique and Chinese indexing strategy are given in Table 5. The results of mono-lingual runs give benchmarking scores for CLIR runs.

As expected, the highest MAP 0.4681 is achieved by the monolingual run VMNET-CS-CS-01-T, in which the questions were manually segmented and all the noise words were removed.

It is encouraging to see that the automatic run VMNET-CS-CS-02-T with only question template phrase removal has a slightly lower MAP 0.4419 than that (0.4488) of the best performance CS-CS run in the NTCIR-8 evaluation forum (Sakai, et al., 2010).

If unigrams were used as the only indexing units, the MAP of VMNET-CS-CS-04-T dropped from 0.4681 to 0.3406. On the other hand, all runs using bigrams as indexing units either exclusively or jointly performed very well. The MAP of run VMNET-CS-CS-05-T using bigrams only is 0.4653, which is slightly

[3] http://research.nii.ac.jp/ntcir/ntcir-ws8/ws-en.html.

lower than that of the top performer run VMNET-CS-CS-01-T, which used two forms of indexing units. However, retrieval performance could be maximised by using both unigrams and bigrams as indexing units.

The highest MAP (0.3756) of a CLIR run is achieved by run VMNET-EN-CS-03-T, which used VMNET for translation. Comparing it to our manual run VMNET-CS-CS-01-T, there is around 9% performance degradation as a result of the influence of noise words in the questions, and the possible information loss or added noise due to English-to-Chinese translation, even though the named entities translation precision is relatively high.

The best EN-CS CLIR run (MAP 0.4209) in all submissions to the NTCIR-8 ACLIA task used the same indexing technique (bigrams and unigrams) and ranking function (BM25) as run VMNET-EN-CS-03-T but with "query expansion based on RSV" (Sakai, et al., 2010). The MAP difference 4.5% between the forum best run and our CLIR best run could suggest that using query expansion is an effective way to improve the CLIR system performance.

Runs VMNET-EN-CS-01-T and VMNET-EN-CS-04-T, that both used Google Translate provide direct comparisons with runs VMNET-EN-CS-02-T and VMNET-EN-CS-03-T, respectively, which employed VMNET for translation. All runs using VMNET performed better than the runs using Google Translate.

Run Name	MAP
NTCIR-8 CS-CS BEST	**0.4488**
VMNET-CS-CS-01-T	0.4681
VMNET-CS-CS-02-T	**0.4419**
VMNET-CS-CS-03-T	0.4189
VMNET-CS-CS-04-T	0.3406
VMNET-CS-CS-05-T	0.4653
NTCIR-8 EN-CS BEST	**0.4209**
VMNET-EN-CS-01-T	0.3161
VMNET-EN-CS-02-T	0.3408
VMNET-EN-CS-03-T	**0.3756**
VMNET-EN-CS-04-T	0.3449

Table 5. Results of all experimental runs

The different performances between CLIR runs using Google Translate and VMENT is the joint result of the translation improvement and other translation differences. As shown in Table 8, VMNET found the correct translations for 8 more topics than Google Translate. It should be noted that there are two topics (ACLIA2-CS-0008 and ACLIA2-CS-0088) not included in the final CLIR evaluation (Sakai, et al., 2010). Also, there is one phrase, "Kenneth Yen (K. T. Yen) (严凯泰)", which VMNET couldn't find the correct translation for, but it detected a highly associated term "Yulon - 裕隆汽车", an automaker company in Taiwan; Kenneth Yen is the CEO of *Yulon*. Although *Yulon* is not a correct translation, it is still a good query term because it is then possible to find the correct answer for the question: "*Who is Kenneth Yen*?". However, this topic was not included in the NTCIR-8 IR4QA evaluation.

Moreover, it is possible to have multiple explanations for a term. In order to discover as many question-related documents as possible, alternative translations found by VMNET are also used as additional query terms. They are shown in Table 6. For example, 丁克 is the Chinese term for DINK in Mainland China, but 顶客族 is used in Taiwan. Furthermore, because VMNET gives the Wikipedia translation the highest priority if only one entry is found, a person's full name is used in person name translation rather than the short commonly used name. For example, *Cheney* (former vice president of U.S.) is translated into 迪克・切尼 rather than just 切尼.

NE	VMNET	Wiki Title
Princess Nori	纪宫公主	黑田清子
DINK	丁克	顶客族
BSE	疯牛病	牛海绵状脑病
Three Gorges Dam	三峡大坝	三峡工程

Table 6. Alternative translations

The biggest difference, 3.07%, between runs that used different translation is from runs VMNET-EN-CS-03-T and VMNET-EN-CS-04-T, which both pruned the question template phrase for simple query processing. Although the performance improvement is not obvious, the correct translations and the additional query terms found by VMNET are still very valuable.

8 Conclusions

General machine translation can already achieve very good translation results, but with our proposed approach we can further improve the translation accuracy. With a proper adjustment of this approach, it could be used in a situation where there is a need for higher precision of complex phrase translation.

The results from our CLIR experiments indicate that VMNET is also capable of providing high quality query terms. A CLIR system can achieve good results for answer finding by using the VMNET for translation, simple indexing technique (bigrams and unigrams), and plain question template phrase pruning.

Run Name	**Indexing Units**	**Query Processing**
VMNET-CS-CS-01-T	U + B	Manually segment the question and remove all the noise words
VMNET-CS-CS-02-T	U + B	Prune the question template phrase
VMNET-CS-CS-03-T	U + B	Use the whole question without doing any extra processing work
VMNET-CS-CS-04-T	U	As VMNET-CS-CS-01-T
VMNET-CS-CS-05-T	B	As VMNET-CS-CS-01-T
VMNET-EN-CS-01-T	U + B	Use Google Translate on the whole question and use the entire translation as query
VMNET-EN-CS-02-T	U + B	Use VMNET translation result without doing any further processing
VMNET-EN-CS-03-T	U + B	As above, but prune the Chinese question template from translation
VMNET-EN-CS-04-T	U + B	Use Google Translate on the whole question and prune the Chinese question template phrase from the translation

Table 7. The experimental runs. For indexing units, U means unigrams; B means bigrams.

Topic ID	**Question with OOV Phrases**	**Correct**	**GT**	**VMNET**
ACLIA2-CS-0002	What is the relationship between the movie "***Riding Alone for Thousands of Miles***" and ZHANG Yimou?	千里走单骑	利民为千里单独	千里走单骑
ACLIA2-CS-0008	Who is ***LI Yuchun***?	李宇春	李玉春	李宇春
ACLIA2-CS-0024	Why does ***Qingling*** build "panda corridor zone"	秦岭	宋庆龄	宋庆龄
ACLIA2-CS-0035	Please list the events related to the movie "***Initials D***".	头文字 D	缩写 D 的事件	缩写 D 的事件
ACLIA2-CS-0036	Please list the movies in which ***Zhao Wei*** participated.	赵薇	照委	赵薇
ACLIA2-CS-0038	What is the relationship between Xia Yu and ***Yuan Quan***.	袁泉	袁区广	袁泉
ACLIA2-CS-0048	Who is ***Sean Chen(Chen Shin-An)***?	陈信安	肖恩陈（陈新的）	陳信安
ACLIA2-CS-0049	Who is ***Lung Yingtai***?	龙应台	龙瀛台	龙应台
ACLIA2-CS-0057	What is the disputes between China and Japan for the undersea natural gas field in the ***East China Sea***?	东海	东中国海域	东海
ACLIA2-CS-0066	What is the relationship between two Russian politicians, ***Kasianov*** and Putin?	卡西亚诺夫	Kasianov	米哈伊尔・米哈伊洛维奇・卡西亚诺夫
ACLIA2-CS-0074	Where are Japan's ***Northern Territories*** located?	北方领土	北方领土	北领地
ACLIA2-CS-0075	Which countries have borders in the ***Kashimir*** region?	克什米尔	Kashimir	Kashimir
ACLIA2-CS-0088	What is the relationship between the Golden Globe Awards and ***Broken-back Mountain***?	断臂山	残破的背山	断臂山
ACLIA2-CS-0089	What is the relationship between *Kenneth Yen(**K. T. Yen**)* and China?	严凯泰	肯尼思日元（观塘日元）	裕隆汽车

Table 8. The differences between Google Translate and VMNET translation of OOV phrases in which GT or VMNET was wrong.

References

Chan, Y.-C., Chen, K.-H., & Lu, W.-H. (2007). *Extracting and Ranking Question-Focused Terms Using the Titles of Wikipedia Articles.* Paper presented at the NTCIR-6.

Chen, A., He, J., Xu, L., Gey, F. C., & Meggs, J. (1997, 1997). *Chinese text retrieval without using a dictionary.* Paper presented at the SIGIR '97: Proceedings of the 20th annual international ACM SIGIR conference on Research and development in information retrieval.

Chen, A., Jiang, H., & Gey, F. (2000). Combining multiple sources for short query translation in Chinese-English cross-language information retrieval. 17-23.

Ferrández, S., Toral, A., Ferrández, Ó., Ferrández, A., & Muñoz, R. (2007). Applying Wikipedia's Multilingual Knowledge to Cross–Lingual Question Answering *Natural Language Processing and Information Systems* (pp. 352-363).

Lu, C., Xu, Y., & Geva, S. (2007). Translation disambiguation in web-based translation extraction for English–Chinese CLIR. 819-823.

NTCIR Project. (2010). Tools. from http://research.nii.ac.jp/ntcir/tools/tools-en.html

Robert, W. P. L., & Kwok, K. L. (2002). A comparison of Chinese document indexing strategies and retrieval models. *ACM Transactions on Asian Language Information Processing (TALIP), 1*(3), 225-268.

Sakai, T., Shima, H., Kando, N., Song, R., Lin, C.-J., Mitamura, T., et al. (2010). *Overview of NTCIR-8 ACLIA IR4QA.* Paper presented at the Proceedings of NTCIR-8, to appear.

Shi, L., Nie, J.-Y., & Cao, G. (2008). *RALI Experiments in IR4QA at NTCIR-7.* Paper presented at the NTCIR-7.

Su, C.-Y., Lin, T.-C., & Wu, S.-H. (2007). *Using Wikipedia to Translate OOV Terms on MLIR.* Paper presented at the NTCIR-6.

Tatsunori Mori, K. T. (2007). *A method of Cross-Lingual Question-Answering Based on Machine Translation and Noun Phrase Translation using Web documents.* Paper presented at the NTCIR-6.

The Stanford Natural Language Processing Group. (2010). Stanford Log-linear Part-Of-Speech Tagger. from http://nlp.stanford.edu/software/tagger.shtml

University of Pennsylvania. (2010). POS tags. from http://bioie.ldc.upenn.edu/wiki/index.php/POS_tags

Zhang, Y., & Vines, P. (2004). *Using the web for automated translation extraction in cross-language information retrieval*. Paper presented at the Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval.

Zhang, Y., Vines, P., & Zobel, J. (2005). Chinese OOV translation and post-translation query expansion in Chinese–English cross-lingual information retrieval. *ACM Transactions on Asian Language Information Processing (TALIP), 4*(2), 57-77.

Ontology driven content extraction using interlingual annotation of texts in the OMNIA project

Achille Falaise, David Rouquet, Didier Schwab, Hervé Blanchon, Christian Boitet
LIG-GETALP, University of Grenoble
{Firstname}.{Lastname}@imag.fr

Abstract

OMNIA is an on-going project that aims to retrieve images accompanied with multilingual texts. In this paper, we propose a generic method (language and domain independent) to extract conceptual information from such texts and spontaneous user requests. First, texts are labelled with interlingual annotation, then a generic extractor taking a domain ontology as a parameter extract relevant conceptual information. Implementation is also presented with a first experiment and preliminary results.

1 Introduction

The OMNIA project (Luca Marchesotti et al., 2010) aims to retrieve images that are described with multilingual free companion texts (captions, comments, etc.) in large Web datasets. Images are first classified with formal descriptors in a lightweight ontology using automatic textual and visual analysis. Then, users may express spontaneous queries in their mother tongue to retrieve images. In order to build both formal descriptors and queries for the ontology, a content extraction in multilingual texts is required.

Multilingual content extraction does not imply translation. It has been shown in (Daoud, 2006) that annotating words or chunks with interlingual lexemes is a valid approach to initiate a content extraction. We thus skip syntactical analysis, an expensive and low quality process, and get language-independent data early in our flow, allowing further treatments to be language-independent. We use the lightweight ontology for image classifications as the formal knowledge representation tha determines relevant information to extract. This ontology is considered as a domain parameter for the content extractor.

We are testing this method on a database provided for the image retrieval challenge CLEF09 by the Belgium press agency Belga. The database contains 500K images with free companion texts of about 50 words (about 25M words in total). The texts in the database are in English only, and we ”simulate” multilinguism with partially post-edited machine translation.

The rest of the paper is organized as follow. We first depict our general architecture deployed for CLIA and then detail the various processes involved : interlingual annotation, conceptual vector based disambiguation and ontology driven content extraction. We conclude with the first results of experimentations on the CLEF09 data.

2 General architecture

2.1 General process

In our scenario, there are two types of textual data to deal with : companion texts in the database (captions), but also user requests. The two are processed in a very similar way.

The general architecture is depicted in figure 1. The main components, that will be described in detail, may be summarized as follows:

- Texts (both companions and requests) are first lemmatised with a language-dependent piece of software. Ambiguities are preserved in a Q-graph structure presented in section 3.1.2.

Proceedings of the 4th International Workshop on Cross Lingual Information Access at COLING 2010, pages 52–60,
Beijing, August 2010

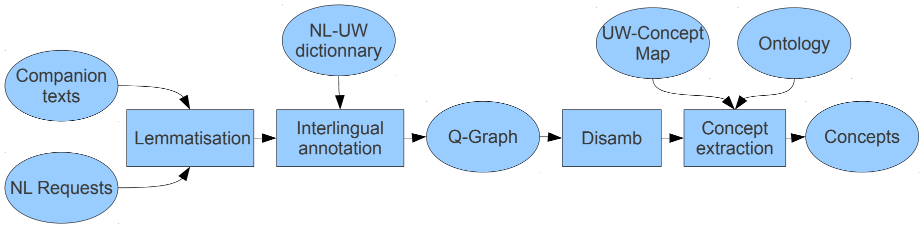


Figure 1: General architecture of CLIA in the OMNIA project

- Then, the lemmatised texts are annotated with interlingual (ideally unambiguous) lexemes, namely Universal Words (UW) presented in section 3.1.1. This adds a lot of ambiguities to the structure, as an actual lemma may refer to several semantically different lexemes.

- The possible meanings for lemmas are then weighted in the Q-graph through a disambiguation process.

- Finally, relevant conceptual information is extracted using an alignment between a domain ontology and the interlingual lexemes.

The conceptual information in the output may adopt different shapes, such as a weighted conceptual vector, statements in the A-Box of the ontology or annotations in the original text, etc.

In the case of OMNIA, conceptual information extracted from companion texts is stored in a database, while conceptual information extracted from users requests are transformed into formal requests for the database (such as SQL, SPARQL, etc.).

2.2 Implementation

The general process is implemented following a Service Oriented Architecture (SOA). Each part of the process corresponds to a service.

This allowed us to reuse part of existing resources developed on heterogeneous platforms using web interfaces (in the best case REST interfaces (Fielding, 2000), but frequently only HTML form-based interfaces). A service supervisor has been built to deal with such an heterogeneity and address normalization issues (e.g. line-breaks, encoding, identification, cookies, page forwarding, etc.).

This architecture is able to process multiple tasks concurrently, allowing to deal with users requests in real time while processing companion texts in the background.

3 Interlingual annotation

We present in this section the preliminary treatments of multilingual texts (image companion texts or user requests) that are required for our content extraction process (Rouquet and Nguyen, 2009a).

In order to allow a content extraction in multilingual texts, we propose to represent texts with the internal formalism of the Q-Systems and to annotate chunks with UNL interlingual lexemes (UW) . Roughly, we are making an interlingual lemmatisation, containing more information than simple tagging, that is not currently proposed by any lemmatisation software.

3.1 Resources and data structures

3.1.1 The Universal Network Language

UNL (Boitet et al., 2009; Uchida Hiroshi et al., 2009) is a pivot language that represents the meaning of a sentence with a semantic abstract structure (an hyper-graph) of an equivalent English sentence.

The vocabulary of UNL consists in a set of Universal Words (UW). An UW consists of:

1. a *headword*, if possible derived from English, that can be a word, initials, an expression or even an entire sentence. It is a label for the concepts it represents in its original language ;

2. a *list of restrictions* that aims to precisely specify the concept the UW refers to. Restrictions are semantic relations with other

UW. The most used is the "icl" relation that points to a more general UW.

Examples :

- book(icl>do, agt>human, obj>thing) and book(icl>thing).
  Here, the sense of the headword is focused by the attributes.
- ikebana(icl>flower_arrangement).
  Here, the headword comes from Japanese.
- go_down.
  Here, the headword does not need any refinement.

Ideally, an UW refers unambiguously to a concept, shared among several languages. However, UW are designed to represent acceptions in a language ; we therefore find distinct UW that refer to the same concept as for "affection" and "disease".

We are mainly using the 207k UW built by the U++ Consortium (Jesus Cardeñosa et al., 2009) from the synsets of the Princeton WordNet, that are linked to natural languages via bilingual dictionaries. The storage of these dictionaries can be supported by a suitable platform like PIVAX (Nguyen et al., 2007) or a dedicated database. The gain of a pivot language is illustrated in figure 2. If we want to add a new language in the multilingual system, we just need to create the links with the pivot but not with all the other languages.

3.1.2 The Q-Systems

We can think of inserting the UW annotations with tags (e.g. XML) directly along the source text as in table 1. However, this naive approach is not adequate to represent the segmentation ambiguities that can occur in the text interpretation (in the example of table 1, we list the different possible meanings for "in", but cannot represent "waiting", "room" and "waiting room" as three possible lexical units).

In order to allow the representation of segmentation and other ambiguities, that can occur in a text interpretation, we propose to use the Q-Systems. They represent texts in an adequate graph structure decorated with bracketed expressions (trees) and, moreover, allow processing on this structure via graph rewriting rules (a set of such rewriting rules is a so called Q-System).

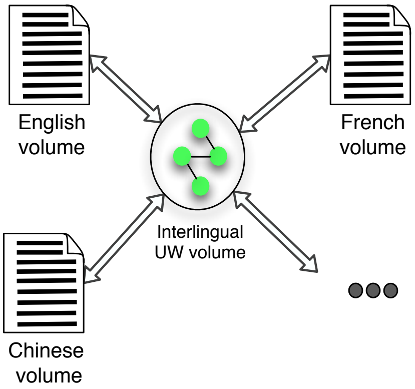


Figure 2: Multilingual architecture with a pivot

in a waiting room
`<tag uw='in(icl-sup-how), in(icl-sup-adj), in(icl-sup-linear_unit, equ-sup-inch)'>`**in**`</tag> <tag uw='unk'>`**a**`</tag> <tag uw='waiting_room(icl-sup-room, equ-sup-lounge)'>`**waiting room**`</tag>`

Table 1: Naive annotation of a text fragment

An example of the Q-System formalism is given in figure 3 of section 3.2.3. It presents successively : the textual input representing a Q-graph, a rewriting rule and a graphical view of the Q-graph obtained after the application of the rule (and others).

The Q-Systems were proposed by Alain Colmeraurer at Montreal University (Colmerauer, 1970). For our goal, they have three main advantages :

- they provide the formalized internal structure for linguistic portability that we mentioned in the introduction (Hajlaoui and Boitet, 2007) ;
- they unify text processing with powerful graph rewriting systems ;

- they allow the creation or the edition of a process by non-programmers (e.g. linguists) using SLLP (Specialized Language for Linguistic Programming).

We are actually using a reimplementation of the Q-Systems made in 2007 by Hong-Thai Nguyen during his PhD in the LIG-GETALP team (Nguyen, 2009).

3.2 Framework of the annotation process

3.2.1 Overview

The annotation process is composed by the following steps :

1. splitting the text in fragments if too long ;
2. lemmatisation with a specialized software ;
3. transcription to the Q-Systems format ;
4. creation of local bilingual dictionaries (source language - UW) for each fragment with PIVAX ;
5. execution of those dictionaries on the fragments ;

3.2.2 Lemmatisation

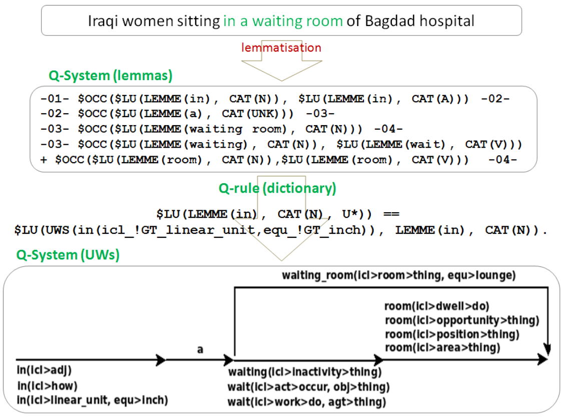
As we want to use dictionaries where entries are lemmas, the first step is to lemmatise the input text (i.e. to annotate occurrences with possible lemmas). This step is very important because it although gives the possible segmentations of the text in lexical units. It brings two kinds of ambiguities into play : on one hand, an occurrence can be interpreted as different lemmas, on the other, there can be several possible segmentations (eventually overlapping) to determine the lexical units.

For content extraction or information retrieval purpose, it is better to preserve an ambiguity than to badly resolve it. Therefore we expect from a lemmatiser to keep all ambiguities and to represent them in a confusion network (a simple tagger is not suitable). Several lemmatiser can be used to cover different languages. For each of them, we propose to use a dedicated ANTLR grammar (Terence Parr et al., 2009) in order to soundly transform the output in a Q-graph.

To process the Belga corpus, we developed a lemmatiser that produce natively Q-graphs. It is based on the morphologic dictionary DELA[1] available under LGPL licence.

3.2.3 Local dictionaries as Q-Systems

Having the input text annotated with lemmas, with the Q-System formalism, we want to use the graph rewriting possibilities to annotate it with UW. To do so, we use PIVAX export features to produce rules that rewrite a lemma in an UW (see figure 3). Each rule correspond to an entry in the bilingual dictionary. To obtain a tractable Q-Systems (sets of rules), we built local dictionaries that contain the entries for fragments of the text (about 250 words in the first experiment).


Iraqi women sitting in a waiting room of Bagdad hospital
lemmatisation
Q-System (lemmas)

```
-01- $OCC($LU(LEMME(in), CAT(N)), $LU(LEMME(in), CAT(A))) -02-
-02- $OCC($LU(LEMME(a), CAT(UNK))) -03-
-03- $OCC($LU(LEMME(waiting room), CAT(N))) -04-
-03- $OCC($LU(LEMME(waiting), CAT(N)), $LU(LEMME(wait), CAT(V)))
+ $OCC($LU(LEMME(room), CAT(N)),$LU(LEMME(room), CAT(V)))  -04-
```

```
$LU(LEMME(in), CAT(N), U*)) ==
$LU(UWS(in(icl_!GT_linear_unit,equ_!GT_inch)), LEMME(in), CAT(N)).
```

Q-System (UWs)
waiting_room(icl>room>thing, equ>lounge)
room(icl>dwell>do)
room(icl>opportunity>thing)
room(icl>position>thing)
room(icl>area>thing)
a
in(icl>adj)
in(icl>how)
in(icl>linear_unit, equ>inch)
waiting(icl>inactivity>thing)
wait(icl>act>occur, obj>thing)
wait(icl>work>do, agt>thing)

Figure 3: Creation and execution of a Q-System

Considering the significant quantity of ambiguities generated by this approach (up to a dozen UW for a single word), we need to include a disambiguation process. This process, based on conceptual vectors, is presented in the next section.

4 *Conceptual vector based disambiguation*

Vectors have been used in NLP for over 40 years. For information retrieval, the standard vector model (SVM) was invented by Salton (Salton, 1991) during the late 60's, while for meaning representation, latent semantic analysis (LSA)

[1] http://infolingu.univ-mlv.fr/DonneesLinguistiques/Dictionnaires/telechargement.html

was developed during the late 80's (Deerwester et al., 1990). These approaches are inspired by distributional semantics (Harris et al., 1989) which hypothesises that a word meaning can be defined by its co-text. For example, the meaning of ‘*milk*’ could be described by {‘*cow*’, ‘*cat*’, ‘*white*’, ‘*cheese*’, ‘*mammal*’, . . . }. Hence, distributional vector elements correspond directly (for SVM) or indirectly (for LSA) to lexical items from utterances.

The conceptual vector model is different as it is inspired by componential linguistics (Hjelmlev, 1968) which holds that the meaning of words can be described with semantic components. These can be considered as atoms of meaning (known as primitives (Wierzbicka, 1996)), or also only as constituents of the meaning (known as semes, features (Greimas, 1984), concepts, ideas). For example, the meaning of ‘*milk*’ could be described by {*LIQUID*, *DAIRY PRODUCT*, *WHITE*, *FOOD*, . . . }. Conceptual vectors model a formalism for the projection of this notion in a vectorial space. Hence, conceptual vector elements correspond to concepts indirectly, as we will see later.

For textual purposes[2], conceptual vectors can be associated to all levels of a text (word, phrase, sentence, paragraph, whole texts, etc.). As they represent ideas, they correspond to the notion of *semantic field*[3] at the lexical level, and to the overall thematic aspects at the level of the entire text.

Conceptual vectors can also be applied to lexical meanings. They have been studied in word sense disambiguation (WSD) using isotopic properties in a text, i.e. redundancy of ideas (Greimas, 1984). The basic idea is to maximise the overlap of shared ideas between senses of lexical items. This can be done by computing the angular distance between two conceptual vectors (Schwab and Lafourcade, 2007).

In our case, conceptual vectors are used for automatic disambiguation of texts. Using this method, we calculate confidence score for each UW hypothesis appearing in the Q-Graph.

5 *Ontology driven content extraction*

The content extraction has to be leaded by a "knowledge base" containing the informations we want to retrieve.

5.1 Previous works in content extraction

This approach has its roots in machine translation projects such as C-Star II (1993-1999) (Blanchon and Boitet, 2000) and Nespole! (2000-2002) (Metze et al., 2002), for on the fly translation of oral speech acts in the domain of tourism. In these projects, semantic transfer was achieved through an IF (Inter-exchange Format), that is a semantic pivot dedicated to the domain. This IF allows to store information extracted from texts but is although used to lead the content extraction process by giving a formal representation of the relevant informations to extract, according to the domain.

The Nespole! IF consists of 123 concepts from the tourism domain, associated with several arguments and associable with speech acts markers. The extraction process is based on patterns. As an example, the statement ”*I wish a single room from September 10th to 15th*” may be represented as follows:

```
{ c:give-information+disposition+room
  ( disposition=(desire, who=i),
    room-spec=
    ( identifiability=no,single_room ),
    time=
    ( start-time=(md=10),
      end-time(md=15, month=9)
    )
  )
}
```

5.2 Ontologies as parameter for the domain

In the project OMNIA, the knowledge base has the form of a lightweight ontology for image classification [4]. This ontology contains 732 concepts in the following domains : animals, politics, religion, army, sports, monuments, transports, games, entertainment, emotions, etc. To us, using an ontology has the following advantages :

- Ontologies give an axiomatic description of a domain, based on formal logics (usu-

[2] Conceptual vectors can be associated with any content, not only text: images, videos, multimedia, Web pages, etc.

[3] The semantic field is the set of ideas conveyed by a term.

[4] http://kaiko.getalp.org/kaiko/ontology/OMNIA/OMNIA_current.owl

ally description logics (Baader et al., 2003)) with an explicit semantic. Thus, the knowledge stored in them can be used soundly by software agents;

- Ontological structures are close to the organisation of ideas as semantic networks in human mind (Aitchenson, 2003) and are labeled with strings derived from natural languages. Thus humans can use them (browsing or contributing) in a pretty natural way;
- Finally, with the advent of the Semantic Web and normative initiatives such as the W3C[5], ontologies come with a lot of shared tools for editing, querying, merging, etc.

As the content extractor might only process UW annotations, it is necessary that the knowledge base is whether expressed using UW or linked to UW. The ontology is here considered as a domain parameter of content extraction and can be changed to improve preformances on specific data collections. Therefore, given any OWL ontology[6], we must be able to link it with a volume of UW considering the following constraints :
Creating manually such correspondences is costly due to the size of resources so an automatic process is requiered.
Ontologies and lexicons evolve over the time so an alignment must be adaptable to incremental evolutions of resources.
The correspondences must be easily manipulated by users so they can manually improve the quality of automatically created alignments with post-edition.

Constructing and maintaining an alignment between an ontology and an UW lexicon is a challenging task (Rouquet and Nguyen, 2009b). Basically, any lexical resource can be represented in an ontology language as a graph. We propose to use an OWL version of the UW volume available on Kaiko website [7]. It allows us to benefit of classical ontology matching techniques and tools (Euzenat and Shvaiko, 2007) to represent, compute and manipulate the alignment. We implemented two string based matching techniques on top of the alignment API (Euzenat, 2004). Specific disambiguation methods are in development to improve the alignment precision. Some of them are based on conceptual vectors presented in section 4, others will adapt structural ontology matching techniques. This approach to match an ontology with a lexical resource is detailled in (Rouquet et al., 2010).

5.3 The generic extractor

In the case of the OMNIA project, the system output format is constraint by the goal of an integration with visual analysis results, in a larger multimodal system. The visual analysis systems are also based on concept extraction, but does not need an ontology to organise concepts. Therefore, our results has to remain autonaumous, which means without references to the ontology used to extract concepts. So, we use a simple concept vector as output, with intensity weights; practically, a simple data-value pairs sequence formatted in XML.

Concept extraction is achieved through a 3 steps process, has shown in figure 4.

1. *Concept matching*: each UW in the Q-Graph, that matches a concept according to the UW-concept map, is labelled with this concept.
2. *Confidence calculation*: each concept label is given a confidence score, in accordance with the score of the UW carrying the concept, obtained after disambiguation, and pondered according to the number of UWs in the Q-Graph. It is planed to take into account a few linguistics hints here, such as negations, and intensity adverbs.
3. *Score propagation*: because we need autonomous results, we have to perform all ontology-based calculation before releasing them. The confidence scores are propagated in the ontology concept hierarchy: for each

[5] http://www.w3.org/

[6] http://www.w3.org/2004/OWL/

[7] http://kaiko.getalp.org

labelled concept, its score is added to the super-concept, and so on.

The ontology and the derivated UW-concept map are considered as parameters for the treatments, and may be replaced in accordance with the domain, and the relevance of the concepts and their hierarchy, according to the task.

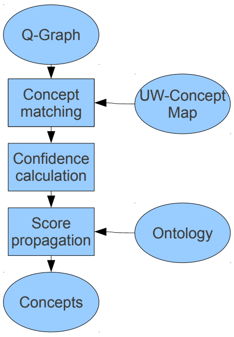


Figure 4: Detail of concept extraction.

6 Experiments

For a first experiment, we used a small dataset, containing:

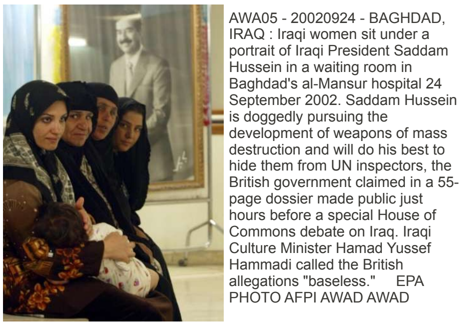
- a sub-corpus of 1046 English companion texts from CLEF09 corpus (press pictures and captions of about 50 words),
- a 159 concepts ontology, designed for picture and emotions depiction,
- a UW-concept map comprising 3099 UW.

It appeared that, with this parameters, concepts where extracted for only 25% of the texts. This preliminary result stressed the importance of recall for such short texts. However, there were many ways to improve recall in the system:

- improve the ontology, in order to better cover the press domain;
- significantly increase the quantity of UW linked to concepts (only 3099 obtained for this experiment), by considering synonyms during the linking process;
- using UW restrictions during concept matching for UW that are not directly linked to a concept, as these restrictions are a rich source of refined semantic information.

A second experiment with an improved ontology, including 732 concepts, and the use of UW restrictions, showed very promising results. Concepts were retrieved from 77% of texts. The remaining texts were very short (less than 10 words, sometime just date or name).

For example, we extracted the following concepts from the picture and companion text reproduced in figure 5.



AWA05 - 20020924 - BAGHDAD, IRAQ : Iraqi women sit under a portrait of Iraqi President Saddam Hussein in a waiting room in Baghdad's al-Mansur hospital 24 September 2002. Saddam Hussein is doggedly pursuing the development of weapons of mass destruction and will do his best to hide them from UN inspectors, the British government claimed in a 55-page dossier made public just hours before a special House of Commons debate on Iraq. Iraqi Culture Minister Hamad Yussef Hammadi called the British allegations "baseless." EPA PHOTO AFPI AWAD AWAD

Figure 5: Picture document and companion text example.

CONCEPT	WEIGHT
BUILDING	0.098
HOSPITAL	0.005
HOUSE	0.043
MINISTER	0.016
OTHER_BUILDING	0.005
PEOPLE	0.142
PERSON	0.038
POLITICS	0.032
PRESIDENT	0.016
RESIDENTIAL_BUILDING	0.043
WOMAN	0.005

As this results were more consistent, we could have a preliminary survey about precision, on a 30 texts sample. While disambiguation implementation is still at an early stage, weights were not yet taken into account. A concept match can be considered correct following two criterons :

1. **Visual relevance** considers a concept as correct if carried by an element of the picture; for instance, the match of concept

"SPORT" is regarded as correct for a picture containing a minister of sports, even if not actually performing any sport.

2. **Textual relevance** considers a concept as correct if carried by a word of the text, as parts of texts may involve concepts that are not actually present in the picture, such as contextual information, previous events, etc.

124 concepts were found in 23 texts (7 texts had no concept match):

1. 99 concepts were correct according to the visual relevance,
2. 110 were correct according to the textual relevance,
3. 14 were totally incorrect.

We thus have an overall precision score of 0.798 according to the visual relevance and 0.895 according to the textual relevance. Most of the errors where caused by ambiguity problems, and may be addressed with disambiguation process that are not fully implemented yet.

7 Conclusion and perspectives

We exposed a generic system designed to extract content (in the form of concepts) from multilingual texts. Our content extraction process is generic regarding to two aspects :

- it is language independent, as it process an interlingual representation of the texts
- the content to be extracted can be specified using a domain ontology as a parameter

This is an ongoing work, and disambiguation through conceptual vectors is expected to improve accuracy, giving significant weights to the hypothetical meanings of words.

In the long run, we will focus on integration with visual content extractors, speed optimization to achieve a real-time demonstrator and detailled evaluation of the method.

References

Aitchenson, J. 2003. *Words in the Mind. An Introduction to the Mental Lexicon.* Blackwell Publishers.

Baader, De Franz, Diego Calvanese, Deborah McGuinness, Peter Patel-Schneider, and Daniele Nardi. 2003. *The Description Logic Handbook.* Cambridge University Press.

Blanchon, H. and C. Boitet. 2000. Speech translation for french within the C-STAR II consortium and future perspectives. In *Proc. ICSLP 2000*, pages 412–417, Beijing, China.

Boitet, Christian, Igor Boguslavskij, and Jesus Cardeñosa. 2009. An evaluation of UNL usability for high quality multilingualization and projections for a future UNL++ language. In *Computational Linguistics and Intelligent Text Processing*, pages 361–373.

Colmerauer, A. 1970. Les systèmes-q ou un formalisme pour analyser et synthétiser des phrases sur ordinateur. *département d'informatique de l'Université de Montréal, publication interne*, 43, September.

Daoud, Daoud. 2006. *Il faut et on peut construire des systèmes de commerce électronique à interface en langue naturelle restreints (et multilingues) en utilisant des méthodes orientées vers les sous-langages et le contenu.* Ph.D. thesis, UJF, September.

Deerwester, Scott C., Susan T. Dumais, Thomas K. Landauer, George W. Furnas, and Richard A. Harshman. 1990. Indexing by latent semantic analysis. *Journal of the American Society of Information Science*, 41(6).

Euzenat, Jérôme and Pavel Shvaiko. 2007. *Ontology matching.* Springer, Heidelberg (DE).

Euzenat, Jérôme. 2004. An API for ontology alignment. In *Proceedings of the 3rd International Semantic Web Conference*, pages 698–7112, Hiroshima, Japan.

Fielding, Roy T. 2000. *Architectural styles and the design of network-based software architectures.* Ph.D. thesis, University of California.

Greimas, Algirdas Julien. 1984. *Structural Semantics: An Attempt at a Method.* University of Nebraska Press.

Hajlaoui, Najeh and Christian Boitet. 2007. Portage linguistique d'applications de gestion de contenu. In *TOTh07*, Annecy.

Harris, Zellig S., Michael Gottfried, Thomas Ryckman, Paul Mattick Jr., Anne Daladier, T.N. Harris, and S. Harris. 1989. *The form of Information in Science, Analysis of Immunology Sublanguage*, volume 104 of *Boston Studies in the Philosophy of Science*. Kluwer Academic Publisher, Dordrecht.

Hjelmlev, Louis. 1968. *Prolégolème à une théorie du langage*. éditions de minuit.

Jesus Cardeñosa et al. 2009. The U++ consortium (accessed on september 2009). http://www.unl.fi.upm.es/consorcio/index.php, September.

Luca Marchesotti et al. 2010. The Omnia project (accessed on may 2010). http://www.omnia-project.org, May.

Max Silberztein. 2009. NooJ linguistic software (accessed on september 2009). http://www.nooj4nlp.net/pages/nooj.html, September.

Metze, F., J. McDonough, H. Soltau, A. Waibel, A. Lavie, S. Burger, C. Langley, L. Levin, T. Schultz, F. Pianesi, R. Cattoni, G. Lazzari, N. Mana, and E. Pianta. 2002. The Nespole! speech-to-speech translation system. In *Proceedings of HLT-2002 Human Language Technology Conference*, San Diego, USA, march.

Nguyen, H.T., C. Boitet, and G. Sérasset. 2007. PIVAX, an online contributive lexical data base for heterogeneous MT systems using a lexical pivot. In *SNLP*, Bangkok, Thailand.

Nguyen, Hong-Thai. 2009. EMEU_w, a simple interface to test the Q-Systems (accessed on september 2009). http://sway.imag.fr/unldeco/SystemsQ.po?localhost= /home/nguyenht/SYS-Q/MONITEUR/, September.

Rouquet, David and Hong-Thai Nguyen. 2009a. Interlingual annotation of texts in the OMNIA project. Poznan, Poland.

Rouquet, David and Hong-Thai Nguyen. 2009b. Multilinguïsation d'une ontologie par des corespondances avec un lexique pivot. In *TOTh09*, Annecy, France, May.

Rouquet, David, Cassia Trojahn, Didier Scwab, and Gilles Sérasset. 2010. Building correspondences between ontologies and lexical resources. In *to be published*.

Salton, Gerard. 1991. The Smart document retrieval project. In *Proc. of the 14th Annual Int'l ACM/SIGIR Conf. on Research and Development in Information Retrieval*, Chicago.

Schwab, Didier and Mathieu Lafourcade. 2007. Lexical functions for ants based semantic analysis. In *ICAI'07- The 2007 International Conference on Artificial Intelligence*, Las Vegas, Nevada, USA, juin.

Terence Parr et al. 2009. ANTLR parser generator (accessed on september 2009). http://www.antlr.org/, September.

Uchida Hiroshi et al. 2009. The UNDL foundation (accessed on september 2009). http://www.undl.org/, September.

Wierzbicka, Anna. 1996. *Semantics: Primes and Universals*. Oxford University Press.

Towards multi-lingual summarization: A comparative analysis of sentence extraction methods on English and Hebrew corpora

Marina Litvak and **Hagay Lipman** and **Assaf Ben Gur** and **Mark Last**
Ben Gurion University of the Negev
{litvakm, lipmanh, bengura, mlast}@bgu.ac.il

Slava Kisilevich and **Daniel Keim**
University of Konstanz
slaks@dbvis.inf.uni-konstanz.de
Daniel.Keim@uni-konstanz.de

Abstract

The trend toward the growing multilinguality of the Internet requires text summarization techniques that work equally well in multiple languages. Only some of the automated summarization methods proposed in the literature, however, can be defined as "language-independent", as they are not based on any morphological analysis of the summarized text. In this paper, we perform an in-depth comparative analysis of language-independent sentence scoring methods for extractive single-document summarization. We evaluate 15 published summarization methods proposed in the literature and 16 methods introduced in (Litvak et al., 2010). The evaluation is performed on English and Hebrew corpora. The results suggest that the performance ranking of the compared methods is quite similar in both languages. The top ten bilingual scoring methods include six methods introduced in (Litvak et al., 2010).

1 Introduction

Automatically generated summaries can significantly reduce the information overload on professionals in a variety of fields, could prove beneficial for the automated classification and filtering of documents, the search for information over the Internet and applications that utilize large textual databases.

Document summarization methodologies include *statistic*-based, using either the classic vector space model or a graph representation, and *semantic*-based, using ontologies and language-specific knowledge (Mani & Maybury, 1999). Although the use of language-specific knowledge can potentially improve the quality of automated summaries generated in a particular language, its language specificity ultimately restricts the use of such a summarizer to a single language. Only systems that perform equally well on different languages in the absence of any language-specific knowledge can be considered language-independent summarizers.

As the number of languages used on the Internet increases continiously (there are at least 75 different languages according to a estimate performed by A. Gulli and A. Signorini[1] in the end of January 2005), there is a growing need for language-independent statistical summarization techniques that can be readily applied to text in any language without using language-specific morphological tools.

In this work, we perform an in-depth comparative analysis of 16 methods for language-independent extractive summarization introduced in (Litvak et al., 2010) that utilize either vector or graph-based representations of text documents computed from word segmentation and 15 state-of-the art language-independent scoring methods. The main goal of the evaluation experiments, which focused on English and Hebrew corpora, is to find the most efficient language-independent sentence scoring methods

[1] http://www.cs.uiowa.edu/ asignori/web-size/

Proceedings of the 4th International Workshop on Cross Lingual Information Access at COLING 2010, pages 61–69,
Beijing, August 2010

in terms of summarization accuracy and computational complexity across two different languages.

This paper is organized as follows. The next section describes related work in extractive summarization. Section 3 reviews the evaluated language-independent sentence scoring approaches. Section 4 contains our experimental results on English and Hebrew corpora. The last section comprises conclusions and future work.

2 Related Work

Extractive summarization is aimed at the selection of a subset of the most relevant fragments, which can be paragraphs, sentences, keyphrases, or keywords from a given source text. The extractive summarization process usually involves ranking, such that each fragment of a summarized text gets a relevance score, and extraction, during which the top-ranked fragments are extracted and arranged in a summary in the same order they appeared in the original text. Statistical methods for calculating the relevance score of each fragment can rely on such information as: fragment *position* inside the document, its *length*, whether it contains *keywords* or *title* words.

Research by Luhn (1958), in which the significance factor of a sentence is based on the frequency and the relative position of significant words within that sentence, is considered the first on automated text summarization. Luhn's work was followed shortly thereafter by that of Edmundson (1969) and some time later by studies from Radev et al. (2001) and Saggion et al. (2003), all of who applied linear combinations of multiple statistical methods to rank sentences using the vector space model as a text representation. In (Litvak et al., 2010) we improve the summarization quality by identifying the best linear combination of the metrics evaluated in this paper.

Several information retrieval and machine learning techniques have been proposed for determining sentence importance (Kupiec et al., 1995; Wong et al., 2008). Gong and Liu (2001) and Steinberger and Jezek (2004) showed that singular value decomposition (SVD) can be applied to generate extracts.

Among text representation models, graph-based text representations have gained popularity in automated summarization, as they enable the model to be enriched with syntactic and semantic relations. Salton et al. (1997) were among the first to attempt graph-based ranking methods for single document extractive summarization by generating similarity links between document paragraphs. The important paragraphs of a text were extracted using degree scores. Erkan and Radev (2004) and Mihalcea (2005) introduced approaches for unsupervised extractive summarization that rely on the application of iterative graph based ranking algorithms. In their approaches, each document is represented as a graph of sentences interconnected by similarity relations.

3 Language-Independent Scoring Methods for Sentence Extraction

Various language dependent and independent sentence scoring methods have been introduced in the literature. We selected the 15 most prominent language independent methods for evaluation. Most of them can be categorized as *frequency*, *position*, *length*, or *title*-based, and they utilize vector representation. *TextRank (ML_TR)* is the only method that is based on graph representation, but there are also *position* and *length*-based methods that calculate scores using the overall structure of a document. We have also considered 16 methods proposed in (Litvak et al., 2010), including 13 based on the *graph-theoretic* representation (Section 3.1).

Figure 1 (Litvak et al., 2010) shows the taxonomy of the 31 methods considered in our work. All methods introduced in (Litvak et al., 2010) are denoted by an asterisk (*). Methods requiring a threshold value $t \in [0, 1]$ that specifies the portion of the top rated terms considered significant are marked by a cross in Figure 1 and listed in Table 1 along with the optimal average threshold values obtained after evaluating the methods

Table 1: Selected thresholds for threshold-based scoring methods

Method	Threshold
LUHN	0.9
LUHN_DEG	0.9
LUHN_PR	0.0
KEY	[0.8, 1.0]
KEY_DEG	[0.8, 1.0]
KEY_PR	[0.1, 1.0]
COV	0.9
COV_DEG	[0.7, 0.9]
COV_PR	0.1

on English and Hebrew documents (Litvak et al., 2010).

The methods are divided into three main categories: *structure*-, *vector*-, and *graph*-based methods, and each category also contains an internal taxonomy. Sections 3.2, 3.3, and 3.4 present structure-, *vector*-, and *graph*-based methods, respectively. With each description, a reference to the original work where the method was proposed for extractive summarization is included. We denote sentence by S and text document by D.

3.1 Text Representation Models

The vector-based scoring methods listed below use *tf* or *tf-idf* term weights to evaluate sentence importance while that used by the graph-based methods (except for TextRank) is based on the word-based graph representation model presented in Schenker et al. (2004). We represent each document by a directed, labeled, unweighted graph in which nodes represent unique terms (distinct normalized words) and edges represent order-relationships between two terms. Each edge is labeled with the IDs of sentences that contain both words in the specified order.

3.2 Structure-based Scoring Methods

In this section, we describe the existing structure-based methods for multilingual sentence scoring. These methods do not require any text representation and are based on its structure.

– *Position* (Baxendale, 1958):

POS_L Closeness to the end of the document: $score(S_i) = i$, where i is a sequential number of a sentence in a document;

POS_F Closeness to the beginning of the document: $score(S_i) = \frac{1}{i}$;

POS_B Closeness to the borders of the document: $score(S_i) = max(\frac{1}{i}, \frac{1}{n-i+1})$, where n is the total number of sentences in D.

– *Length* (Satoshi et al., 2001):

LEN_W Number of *words* in a sentence;

LEN_CH Number of *characters* in a sentence.

3.3 Vector-based Scoring Methods

In this section, we describe the vector-based methods for multilingual sentence scoring, that are based on the vector space model for text representation.

– *Frequency*-based:

LUHN (Luhn, 1958) $score(S) = max_{c_i \in \{clusters(S)\}}\{cs_i\}$, where clusters are portions of a sentence bracketed by keywords[2] and $cs_i = \frac{|keywords(c_i)|^2}{|c_i|}$.

KEY (Edmundson, 1969) Sum of the keyword frequencies: $score(S) = \sum_{i \in \{keywords(S)\}} tf_i$, where tf_i is term in-document frequency of keyword i.

COV (Kallel et al., 2004) Ratio of keyword numbers (Coverage): $score(S) = \frac{|keywords(S)|}{|keywords(D)|}$

TF (Vanderwende et al., 2007) Average term frequency for all sentence words: $score(S) = \frac{\sum_{i \in \{words(S)\}} tf_i}{|S|}$.

TFISF (Neto et al., 2000) Average term frequency inverted sentence frequency for all sentence words: $score(S) = \sum_{i \in \{words(S)\}} tf_i \times isf_i$, where $isf_i = 1 - \frac{log(n_i)}{log(n)}$, where n is the number of sentences in a document and n_i is the number of sentences containing word i.

SVD (Steinberger & Jezek, 2004) $score(S)$ is equal to the length of a sentence vector in $\Sigma^2 V^T$ after computing the Singular Value Decomposition of a term by sentence matrix $A = U \Sigma V^T$

– *Title* (Edmundson, 1969) similarity[3] to the title, $score(S) = sim(S, T)$:

TITLE_O using overlap similarity: $\frac{|S \cap T|}{min\{|S|, |T|\}}$

TITLE_J using Jaccard similarity: $\frac{|S \cap T|}{|S \cup T|}$

[2] Luhn's experiments suggest an optimal limit of 4 or 5 non-significant words between keywords.

[3] Due to multilingual focus of our work, *exact* word matching was used in all similarity-based methods.

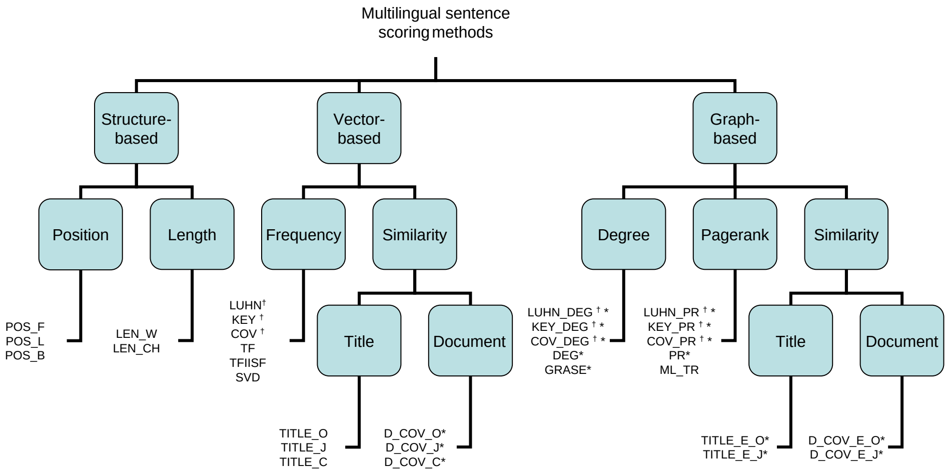


Figure 1: Taxonomy of statistical language-independent sentence scoring methods (Litvak et al., 2010)

TITLE_C using cosine similarity:
$sim(\vec{S},\vec{T}) = cos(\vec{S},\vec{T}) = \frac{\vec{S}\times\vec{T}}{|\vec{S}|\times|\vec{T}|}$

– *Document Coverage* (Litvak et al., 2010). These methods score a sentence according to its similarity to the rest of the sentences in the document ($D - S$) based on the following intuition: the more document content is covered by a sentence, the more important the sentence is to a summary. Redundant sentences containing repetitive information are removed using a similarity filter. $score(S) = sim(S, D - S)$:

D_COV_O using Overlap similarity: $\frac{|S\cap T|}{min\{|S|,|D-S|\}}$

D_COV_J using Jaccard similarity: $\frac{|S\cap T|}{|S\cup D-S|}$

D_COV_C using Cosine similarity:
$cos(\vec{S}, \vec{D - S}) = \frac{\vec{S}\times\vec{D-S}}{|\vec{S}|\times|\vec{D-S}|}$

3.4 Graph-based Scoring Methods

In this section, we describe the methods for multilingual sentence scoring using the graph text representation based on sentence (ML_TR) or word (all except ML_TR) segmentation.

ML_TR Multilingual version of TextRank (Mihalcea, 2005) without morphological analysis. Each document is represented as a directed graph of nodes that stand for sentences interconnected by similarity (*overlap*) relationship. To each edge connecting two vertices the weight is assigned and equal to the similarity value between the corresponding sentences. We used backward links, as it was the most successful according to the reported results in (Mihalcea, 2005). $score(S)$ is equal to PageRank (Brin & Page, 1998) of its node, according to the formula adapted to the weights assigned to edges.

– *Degree*-based (Litvak et al., 2010):[4]

LUHN_DEG A graph-based extension of the LUHN measure, in which a node degree is used instead of a word frequency: words are considered significant if they are represented by nodes of a higher degree than a predefined threshold (see Table 1).

KEY_DEG Graph-based extension of KEY measure.

COV_DEG Graph-based extension of COV measure.

DEG Average degree for all sentence nodes: $score(S) = \frac{\sum_{i\in\{words(S)\}} Deg_i}{|S|}$.

GRASE(GRaph-based Automated Sentence Extractor) Modification of Salton's algorithm (Salton et al., 1997) using the graph

[4] All proposed here degree-based methods, except for GRASE, use undirected graphs and degree of nodes as a predictive feature. The methods based on the directed word graphs and distinguishing between in- and out-links were outperformed in our preliminary experiments by the undirected approach.

representation defined in Section 3.1 above. In our graph representation, all sentences are represented by paths, completely or partially. To identify the relevant sentences, we search for the *bushy* paths and extract from them the sentences that appear the most frequently. Each sentence in the *bushy* path gets a domination score that is the number of edges with its label in the path normalized by the sentence length. The relevance score for a sentence is calculated as a sum of its domination scores over all paths.

– *PageRank*-based:[5]

LUHN_PR A graph-based extension of the LUHN measure in which the node PageRank value is used instead of the word frequency: keywords are those words represented by nodes with a PageRank score higher than a predefined threshold (see Table 1).

KEY_PR Graph-based extension of KEY measure.

COV_PR Graph-based extension of COV measure.

PR Average PageRank for all sentence nodes: $score(S) = \frac{\sum_{i \in \{words(S)\}} PR_i}{|S|}$.

– *Similarity*-based. Edge matching techniques similar to those of Nastase and Szpakowicz (2006) are used. Edge matching is an alternative approach to measure the similarity between graphs based on the number of common edges:

TITLE_E_O Graph-based extension of TITLE_O – Overlap-based edge matching between title and sentence graphs.

TITLE_E_J Graph-based extension of TITLE_J – Jaccard-based edge matching between title and sentence graphs.

D_COV_E_O Graph-based extension of D_COV_O – Overlap-based edge matching between sentence and document complement (the rest of a document sentences) graphs.

D_COV_E_J Graph-based extension of D_COV_J – Jaccard-based edge matching between sentence and document complement graphs.

4 Experiments

4.1 Overview

The quality of the above-mentioned sentence ranking methods was evaluated through a comparative experiment on corpora of English and Hebrew texts. These two languages, which belong to different language families (Indo-European and Semitic languages, respectively), were intentionally chosen for this experiment to increase the generality of our evaluation. The main difference between these languages, is that Hebrew morphology allows morphemes to be combined systematically into complex word-forms. In different contexts, the same morpheme can appear as a separate word-form, while in others it appears agglutinated as a suffix or prefix to another word-form (Adler, 2009).

The goals of the experiment were as follows:
- To evaluate the performance of different approaches for extractive single-document summarization using graph and vector representations.
- To compare the quality of the multilingual summarization methods proposed in our previous work (Litvak et al., 2010) to the state-of-the-art approaches.
- To identify sentence ranking methods that work equally well on both languages.

4.2 Text Preprocessing

Extractive summarization relies critically on proper sentence segmentation to insure the quality of the summarization results. We used a sentence splitter provided with the MEAD summarizer (Radev et al., 2001) for English and a simple splitter for Hebrew splitting the text at every period, exclamation point, or question mark.[6]

4.3 Experimental Data

For English texts, we used the corpus of summarized documents provided for the single doc-

[5] Using undirected word graphs with PageRank does not make sense, since for an undirected graph a node pagerank score is known to be proportional to its degree. Reversing links will result in hub scores instead authority. The methods distinguishing between authority and hub scores were outperformed in our preliminary experiments by the degree-based approach.

[6] Although the same set of splitting rules may be used for both languages, separate splitters were used since the MEAD splitter is restricted to European languages.

ument summarization task at the Document Understanding Conference 2002 (DUC, 2002). This benchmark dataset contains 533 news articles, each of which is at least ten sentences long and has two to three human-generated abstracts of approximately 100 words apiece.

However, to the best of our knowledge, no summarization benchmarks exist for the Hebrew language texts. To collect summarized texts in Hebrew, we set up an experiment[7] in which 50 news articles of 250 to 830 words each from the *Haaretz*[8] newspaper internet site were summarized by human assessors by extracting the most salient sentences. In total, 70 undergraduate students from the Department of Information Systems Engineering, Ben Gurion University of the Negev participated in the experiment. Ten documents were randomly assigned to each of the 70 study participants who were instructed (1) To dedicate at least five minutes to each document, (2) To ignore dialogs and citations, (3) To read the whole document before starting sentence extraction, (4) To ignore redundant, repetitive, or overly detailed information, (5) To obey the minimal and maximal summary constraints of 95 and 100 words, respectively. Summaries were assessed for quality by procedure described in (Litvak et al., 2010).

4.4 Experimental Results

We evaluated English and Hebrew summaries using the ROUGE-1, 2, 3, 4, L, SU and W metrics[9], described in Lin (2004). Our results were not statistically distinguishable and matched the conclusion of Lin (2004). However, because ROUGE-1 showed the largest variation across the methods, all results in the following comparisons are presented in terms of ROUGE-1 metric. Similar to the approach described in Dang (2006), we performed multiple comparisons between the sentence scoring methods. The Friedman test was used to reject the null hypothesis (all methods perform the same) at the 0.0001 significance level, after which we ran the Bonferroni-Dunn test (Demsar, 2006) for pairwise comparisons. Tables 2 and 3 show the results of multiple comparisons and are arranged in descending order with the best approaches on top. Methods not sharing any common letter were significantly different at the 95% confidence level.

Table 2: English: Multiple comparisons of sentence ranking approaches using the Bonferroni-Dunn test of ROUGE-1 Recall

Approach	ROUGE-1													
COV_DEG*	0.436	A												
KEY_DEG*	0.433	A	B											
KEY	0.429	A	B	C										
COV_PR*	0.428	A	B	C	D									
COV	0.428	A	B	C	D									
D_COV_C*	0.428	A	B	C	D									
D_COV_J*	0.425		B	C	D	E								
KEY_PR*	0.424		B	C	D	E								
LUHN_DEG*	0.422			C	D	E	F							
POS_F	0.419					E	F	G						
LEN_CH	0.418			C	D	E	F	G						
LUHN	0.418				D	E	F	G						
LUHN_PR*	0.418					E	F	G	H					
LEN_W	0.416				D	E	F	G	H					
ML_TR	0.414					E	F	G	H					
TITLE_E_J*	0.413						F	G	H	I				
TITLE_E_O*	0.413						F	G	H	I				
D_COV_E_J*	0.410						F	G	H	I				
D_COV_O*	0.405							G	H	I	J			
TFISF	0.405							G	H	I	J			
DEG*	0.403							G	H	I	J			
D_COV_E_O*	0.401								H	I	J	K		
PR*	0.400							G	H	I	J	K		
TITLE_J	0.399									I	J	K		
TF	0.397									I	J	K		
TITLE_O	0.396										J	K		
SVD	0.395									I	J	K		
TITLE_C	0.395										J	K		
POS_B	0.392											K	L	
GRASE*	0.372												L	
POS_L	0.339													M

The Pearson correlation between methods ranking in English and Hebrew was 0.775, which was larger than zero at a significance level of 0.0001. In other words, most of the methods were ranked in nearly the same relative positions in both corpora, and the top ranked methods performed equally well in both languages. The differences in ranking were caused by morphological differences between two languages.

To determine which approaches performed best in both languages, we analyzed the clustering results of the methods in both corpora and found the intersection of the top clusters from the two clustering results. For each language, a document-method matrix of ROUGE scores was created with methods represented by vectors of their ROUGE scores for each document in a corpora. Since most scores are not normally

[7] The software enabling easy selection and storage of sentences to be included in the document extract, can be provided upon request.

[8] http://www.haaretz.co.il

[9] ROUGE toolkit was adapted to Hebrew by specifying "token" using Hebrew alphabet

Table 3: Hebrew: Multiple comparisons of sentence ranking approaches using the Bonferroni-Dunn test of ROUGE-1 Recall

Approach	ROUGE-1						
D_COV_J*	0.574	A					
KEY	0.570	A	B				
COV_DEG*	0.568	A	B				
POS_F	0.567	A	B				
COV	0.567	A	B				
TITLE_J	0.567	A	B				
POS_B	0.565	A	B				
LUHN_PR*	0.560	A	B	C			
LUHN_DEG*	0.560	A	B	C			
D_COV_E_J*	0.559	A	B	C			
LUHN	0.559	A	B	C			
TITLE_E_J*	0.556	A	B	C			
TITLE_E_O*	0.556	A	B	C			
KEY_DEG*	0.555	A	B	C			
LEN_W	0.555	A	B	C			
LEN_CH	0.553	A	B	C			
KEY_PR*	0.546	A	B	C			
COV_PR*	0.546	A	B	C			
TITLE_O	0.545	A	B	C			
D_COV_C*	0.543	A	B	C			
TITLE_C	0.541	A	B	C			
ML_TR	0.519	A	B	C	D		
TFISF	0.514	A	B	C	D		
D_COV_E_O*	0.498	A	B	C	D		
SVD	0.498	A	B	C	D		
D_COV_O*	0.466		B	C	D		
TF	0.427			C	D	E	
DEG*	0.399				D	E	F
PR*	0.331					E	F
GRASE*	0.243						F
POS_L	0.237						F

Table 4: English: Correlation between sentence ranking approaches using Pearson

Approach	Correlated With
POS_F	(LUHN_PR, 0.973), (TITLE_E_J, 0.902), (TITLE_E_O, 0.902)
TITLE_O	(TITLE_J, 0.950)
LEN_W	(LEN_CH, 0.909)
KEY_PR	(COV_PR, 0.944)
TITLE_E_O	(TITLE_E_J, 0.997)

Table 5: Hebrew: Correlation between sentence ranking approaches using Pearson

Approach	Correlated With
KEY	(KEY_DEG, 0.930)
COV	(D_COV_J, 0.911)
POS_F	(POS_B, 0.945), (LUHN_DEG, 0.959), (LUHN_PR, 0.958)
POS_B	(LUHN_DEG, 0.927), (LUHN_PR, 0.925)
TITLE_O	(TITLE_E_J, 0.920), (TITLE_E_O, 0.920)
TITLE_J	(TITLE_E_J, 0.942), (TITLE_E_O, 0.942)
LEN_W	(LEN_CH, 0.954), (KEY_PR, 0.912)
LEN_CH	(KEY_PR, 0.936), (KEY_DEG, 0.915), (COV_DEG, 0.901)
LUHN_DEG	(LUHN_PR, 0.998)
KEY_DEG	(COV_DEG, 0.904)

Table 6: Ranking of the best bilingual scores

Scoring method	Rank in English corpus	Rank in Hebrew corpus	Text Representation
KEY	3	2	vector
COV	4	4	vector
KEY_DEG	2	10	graph
COV_DEG	1	3	graph
KEY_PR	6	12	graph
COV_PR	4	12	graph
D_COV_C	4	14	vector
D_COV_J	5	1	vector
LEN_W	10	10	structure
LEN_CH	9	11	structure

distributed, we chose the K-means algorithm, which does not assume normal distribution of data, for clustering. We ran the algorithm with different numbers of clusters ($2 \leq K \leq 10$), and for each K, we measured two parameters: the minimal distance between neighboring clusters in the clustered data for each language and the level of similarity between the clustering results for the two languages. For both parameters, we used the regular Euclidean distance. For $K \geq 6$, the clusters were highly similar for each language, and the distance between English and Hebrew clustering data was maximal. Based on the obtained results, we left results only for $2 \leq K \leq 5$ for each corpus. Then, we ordered the clusters by the average ROUGE score of each cluster's instances (methods) and identified the methods appearing in the top clusters for all K values in both corpora. Table 6 shows the resulting top ten scoring methods with their rank in each corpus. Six methods introduced in this paper, such as *Document Coverage* (*D_COV_C/J*) and graph adaptations of *Coverage* (*COV_DEG/PR*) and *Key* (*KEY_DEG/PR*), are among these top ten bilingual methods.

Neither *vector*- nor *graph*-based text representation models, however, can claim ultimate superiority, as methods based on both models prominently in the top-evaluated cluster. Moreover, highly-correlated methods (see Tables 4 and 5 for highly-correlated pairs of methods in English and Hebrew corpora, respectively) appear in the same cluster in most cases. As a result, some pairs from among the top ten methods are highly-correlated in at least one language, and only one from each pair can be considered. For example, *LEN_W* and *LEN_CH* have high correlation coefficients (0.909 and 0.954 in English and Hebrew, respectively). Since *LEN_CH* is more appropriate for multilingual processing due to variations in the rules of tokenization between languages (e.g., English vs. German), it may be considered a preferable multilingual metric.

In terms of summarization quality and computational complexity, all scoring functions presented in Table 6 can be considered to perform equally well for bilingual extractive summarization. Assuming their efficient implementation, all methods have a linear computational complexity, $O(n)$, relative to the total number of words in a document. *KEY_PR* and *COV_PR* require additional $O(c(|E|+|V|))$ time for running PageRank, where c is the number of iterations it needs to converge, $|E|$ is the number of edges, and $|V|$ is the number of nodes (distinct words) in a document graph. Since neither $|E|$ nor $|V|$ in our graph representation can be as large as n, the total computation time for *KEY_PR* and *COV_PR* metrics is also linear relative to the document size.

In terms of implementation complexity, *LEN_W* and *LEN_CH* are simpliest, since they even do not require any preprocessing and representation building; *KEY* and *COV* require keywords identification; *D_COV_C,* and *D_COV_J* require vector space model building; *KEY_DEG* and *COV_DEG* need graphs building (order of words); whereas *KEY_PR* and *COV_PR*, in addition, require PageRank implementation.

5 Conclusion and Future Research

In this paper, we conducted in-depth, comparative evaluations of 31 existing (16 of which are mostly graph-based modifications of existing state-of-the-art methods, introduced in (Litvak et al., 2010)) scoring methods[10] using English and Hebrew language texts.

The experimental results suggest that the relative ranking of methods performance is quite similar in both languages. We identified methods that performed significantly better in only one of the languages and those that performed equally well in both languages. Moreover, although vector and graph-based approaches were among the top ranked methods for bilingual application, no text representation model presented itself as markedly superior to the other.

Our future research will extend the evaluations of language-independent sentence ranking metrics to a range of other languages such as German, Arabic, Greek, and Russian. We will adapt similarity-based metrics to multilingual application by implementing them via n-gram matching instead of exact word matching. We will further improve the summarization quality by applying machine learning on described features. We will use additional techniques for summary evaluation and study the impact of morphological analysis on the top ranked bilingual scores using part-of-speech (POS) tagging[11], anaphora resolution, named entity recognition, and taking word sense into account.

Acknowledgments

We are grateful to Michael Elhadad and Galina Volk for providing the ROUGE toolkit adapted to Hebrew alphabet.

References

Adler, M. (2009). Hebrew morphological disambiguation: An unsupervised stochastic word-based approach. Dissertation. http://www.cs.bgu.ac.il/ adlerm/dat/thesis.pdf.

Baxendale, P. (1958). Machine-made index for technical literature-an experiment. *IBM Journal of Research and Development*, *2*, 354–361.

Brin, S., & Page, L. (1998). The anatomy of a large-scale hypertextual web search engine. *Computer networks and ISDN systems*, *30*, 107–117.

Dang, H. T. (2006). Overview of DUC 2006. *Proceedings of the Document Understanding Conference*.

Demsar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, *7*, 1–30.

DUC (2002). Document understanding conference. http://duc.nist.gov.

Edmundson, H. P. (1969). New methods in automatic extracting. *J. ACM*, *16*.

Erkan, G., & Radev, D. R. (2004). LexRank: Graph-based lexical centrality as salience in text summarization. *Journal of Artificial Intelligence Research*, *22*, 457–479.

[10] We will provide the code for our summarizer upon request.

[11] Our experiments have shown that syntactic filters, which select only lexical units of a certain part of speech, do not significantly improve the performance of the evaluated bilingual scoring methods.

Gong, Y., & Liu, X. (2001). Generic text summarization using relevance measure and latent semantic analysis. *Proceedings of the 24th ACM SIGIR conference on Research and development in information retrieval* (pp. 19–25).

Kallel, F. J., Jaoua, M., Hadrich, L. B., & Hamadou, A. B. (2004). Summarization at LARIS laboratory. *Proceedings of the Document Understanding Conference*.

Kupiec, J., Pedersen, J., & Chen, F. (1995). A trainable document summarizer. *Proceedings of the 18th annual international ACM SIGIR conference* (pp. 68–73).

Lin, C.-Y. (2004). ROUGE: A package for automatic evaluation of summaries. *Proceedings of the ACL'04 Workshop: Text Summarization Branches Out* (pp. 74–81).

Litvak, M., Last, M., & Friedman, M. (2010). A new approach to improving multilingual summarization using a genetic algorithm. *Proceedings of the Association for Computational Linguistics (ACL) 2010*. Uppsala, Sweden.

Luhn, H. P. (1958). The automatic creation of literature abstracts. *IBM Journal of Research and Development*, *2*, 159–165.

Mani, I., & Maybury, M. (1999). *Advances in automatic text summarization*.

Mihalcea, R. (2005). Language independent extractive summarization. *AAAI'05: Proceedings of the 20th national conference on Artificial intelligence* (pp. 1688–1689).

Nastase, V., & Szpakowicz, S. (2006). A study of two graph algorithms in topic-driven summarization. *Proceedings of the Workshop on Graph-based Algorithms for Natural Language*.

Neto, J., Santos, A., Kaestner, C., & Freitas, A. (2000). Generating text summaries through the relative importance of topics. *Lecture Notes in Computer Science*, 300–309.

Radev, D., Blair-Goldensohn, S., & Zhang, Z. (2001). Experiments in single and multidocument summarization using MEAD. *First Document Understanding Conference*.

Saggion, H., Bontcheva, K., & Cunningham, H. (2003). Robust generic and query-based summarisation. *EACL '03: Proceedings of the tenth conference on European chapter of the Association for Computational Linguistics*.

Salton, G., Singhal, A., Mitra, M., & Buckley, C. (1997). Automatic text structuring and summarization. *Information Processing and Management*, *33*, 193–207.

Satoshi, C. N., Satoshi, S., Murata, M., Uchimoto, K., Utiyama, M., & Isahara, H. (2001). Sentence extraction system assembling multiple evidence. *Proceedings of 2nd NTCIR Workshop* (pp. 319–324).

Schenker, A., Bunke, H., Last, M., & Kandel, A. (2004). Classification of web documents using graph matching. *International Journal of Pattern Recognition and Artificial Intelligence*, *18*, 475–496.

Steinberger, J., & Jezek, K. (2004). Text summarization and singular value decomposition. *Lecture Notes in Computer Science*, 245–254.

Vanderwende, L., Suzuki, H., Brockett, C., & Nenkova, A. (2007). Beyond SumBasic: Task-focused summarization with sentence simplification and lexical expansion. *Information processing and management*, *43*, 1606–1618.

Wong, K., Wu, M., & Li, W. (2008). Extractive summarization using supervised and semi-supervised learning. *Proceedings of the 22nd International Conference on Computational Linguistics* (pp. 985–992).

More Languages, More MAP?: A Study of Multiple Assisting Languages in Multilingual PRF

Vishal Vachhani Manoj K. Chinnakotla Mitesh M. Khapra Pushpak Bhattacharyya
Department of Computer Science and Engineering,
Indian Institute of Technology Bombay

{vishalv,manoj,miteshk,pb}@cse.iitb.ac.in

Abstract

Multilingual Pseudo-Relevance Feedback (MultiPRF) is a framework to improve the PRF of a *source language* by taking the help of another language called *assisting language*. In this paper, we extend the MultiPRF framework to include multiple assisting languages. We consider three different configurations to incorporate multiple assisting languages - a) Parallel - all assisting languages combined simultaneously b) Serial - assisting languages combined in sequence one after another and c) Selective - dynamically selecting the best feedback model for each query. We study their effect on MultiPRF performance. Results using multiple assisting languages are mixed and it helps in boosting MultiPRF accuracy only in some cases. We also observe that MultiPRF becomes more robust with increase in number of assisting languages.

1 Introduction

Pseudo-Relevance Feedback (PRF) (Buckley et al., 1994; Xu and Croft, 2000; Mitra et al., 1998) is known to be an effective technique to improve the effectiveness of Information Retrieval (IR) systems. In PRF, the top *'k'* documents from the ranked list retrieved using the initial keyword query are *assumed to be relevant*. Later, these documents are used to refine the user query and the final ranked list is obtained using the above refined query. Although PRF has been shown to improve retrieval, it suffers from the following drawbacks: (a) *Lexical and Semantic Non-Inclusion*: the type of term associations obtained for query expansion is restricted to only co-occurrence based relationships in the feedback documents and (b) *Lack of Robustness*: due to the inherent assumption in PRF, *i.e.*, relevance of top *k* documents, performance is sensitive to that of the initial retrieval algorithm and as a result is not robust. Typically, larger coverage ensures higher proportion of relevant documents in the top *k* retrieval (Hawking et al., 1999). However, some resource-constrained languages do not have adequate information coverage in their own language. For example, languages like Hungarian and Finnish have meager online content in their own languages.

Multilingual Pseudo-Relevance Feedback (MultiPRF) (Chinnakotla et al., 2010a) is a novel framework for PRF to overcome the above limitations of PRF. It does so by taking the help of a different language called the *assisting language*. Thus, the performance of a resource-constrained language could be improved by harnessing the good coverage of another language. MulitiPRF showed significant improvements on standard CLEF collections (Braschler and Peters, 2004) over state-of-art PRF system. On the web, each language has its own exclusive topical coverage besides sharing a large number of common topics with other languages. For example, information about Saudi Arabia government policies and regulations is more likely to be found in Arabic language web and also information about a local event in Spain is more likely to be covered in Spanish web than in English. Hence, using multiple languages in conjunction is more likely to ensure satisfaction of the user information need and hence will be more robust.

In this paper, we extend the MultiPRF framework to multiple assisting languages. We study

Proceedings of the 4th International Workshop on Cross Lingual Information Access at COLING 2010, pages 70–78,
Beijing, August 2010

the various possible ways of combining the models learned from multiple assisting languages. We propose three different configurations for including multiple assisting languages in MultiPRF - a) Parallel b) Serial and c) Selective. In Parallel combination, all the assisting languages are combined simultaneously using interpolation. In Serial configuration, the assisting languages are applied in sequence one after another and finally, in Selective configuration, the best feedback model is dynamically chosen for each query. We experiment with each of the above configurations and present both quantitative and qualitative analysis of the results. Results using multiple assisting languages are mixed and it helps in boosting MultiPRF accuracy only in some cases. We also observe that MultiPRF becomes more robust with increase in number of assisting languages. Besides, we also study the relation between number of assisting languages, coverage and the MultiPRF accuracy.

The paper is organized as follows: Section 2, explains the Language Modeling (LM) based PRF approach. Section 3, describes the MultiPRF approach. Section 4 explains the various configurations to extend MultiPRF for multiple assisting languages. Section 6 presents the results and discussions. Finally, Section 7 concludes the paper.

2 PRF in the LM Framework

The Language Modeling (LM) Framework allows PRF to be modeled in a principled manner. In the LM approach, documents and queries are modeled using multinomial distribution over words called *document language model* $P(w|D)$ and *query language model* $P(w|\Theta_Q)$ respectively. For a given query, the document language models are ranked based on their proximity to the query language model, measured using KL-Divergence.

$$KL(\Theta_Q||D) \ = \sum_w P(w|\Theta_Q) \cdot log \frac{P(w|\Theta_Q)}{P(w|D)}$$

Since the query length is short, it is difficult to estimate Θ_Q accurately using the query alone. In PRF, the top k documents obtained through the initial ranking algorithm are assumed to be relevant and used as feedback for improving the estimation of Θ_Q. The feedback documents contain both relevant and noisy terms from which the feedback language model is inferred based on a Generative Mixture Model (Zhai and Lafferty, 2001).

Symbol	Description
Θ_Q	Query Language Model
$\Theta_{L_1}^F$	Feedback Language Model obtained from PRF in L_1
$\Theta_{L_2}^F$	Feedback Language Model obtained from PRF in L_2
$\Theta_{L_1}^{Trans}$	Feedback Model Translated from L_2 to L_1
$t(f\|e)$	Probabilistic Bi-Lingual Dictionary from L_2 to L_1
β, γ	Interpolation coefficients coefficients used in MultiPRF

Table 1: Glossary of Symbols used in explaining MultiPRF

Let $D_F = \{d_1, d_2, \ldots, d_k\}$ be the top k documents retrieved using the initial ranking algorithm. Zhai and Lafferty (Zhai and Lafferty, 2001) model the feedback document set D_F as a mixture of two distributions: (a) the *feedback language model* and (b) the *collection model* $P(w|C)$. The feedback language model is inferred using the EM Algorithm (Dempster et al., 1977), which iteratively accumulates probability mass on the most *distinguishing* terms, *i.e.* terms which are more frequent in the feedback document set than in the entire collection. To maintain query focus the final converged feedback model, Θ_F is interpolated with the initial query model Θ_Q to obtain the final query model Θ_{Final}.

$$\Theta_{Final} = (1 - \alpha) \cdot \Theta_Q + \alpha \cdot \Theta_F$$

Θ_{Final} is used to re-rank the corpus using the KL-Divergence ranking function to obtain the final ranked list of documents. Henceforth, we refer to the above technique as *Model Based Feedback (MBF)*.

3 Multilingual Pseudo-Relevance Feedback (MultiPRF)

Chinnakotla et al. (Chinnakotla et al., 2010a; Chinnakotla et al., 2010b) propose the MultiPRF approach which overcomes the fundamental limitations of PRF with the help of an assisting collection in a different language. Given a query Q in the source language L_1, it is automatically translated into the assisting language L_2. The documents in the L_2 collection are ranked using the query likelihood ranking function (John Lafferty and Chengxiang Zhai, 2003). Using the top k documents, they estimate the feedback model using MBF as described in the previous section. Similarly, they also estimate a feedback model using the original query and the top k documents retrieved from the initial ranking in L_1. Let the resultant feedback models be $\Theta_{L_2}^F$ and $\Theta_{L_1}^F$ respectively. The feedback model estimated in the assisting language $\Theta_{L_2}^F$ is translated back into language L_1 using a probabilistic bi-lingual dictionary $t(f|e)$ from $L_2 \rightarrow L_1$ as follows:

$$P(f|\Theta_{L_1}^{Trans}) = \sum_{\forall \text{ e in } L_2} t(f|e) \cdot P(e|\Theta_{L_2}^F) \quad (1)$$

The probabilistic bi-lingual dictionary $t(f|e)$ is learned from a parallel sentence-aligned corpora in $L_1 - L_2$ based on word level alignments. The probabilistic bi-lingual dictionary acts as a rich source of morphologically and semantically related feedback terms. Thus, the translation model adds related terms in L_1 which have their source as the term from feedback model $\Theta_{L_2}^F$. The final MultiPRF model is obtained by interpolating the above translated feedback model with the original query model and the feedback model of language L_1 as given below:

$$\Theta_{L_1}^{Multi} = (1-\beta-\gamma)\cdot\Theta_Q + \beta\cdot\Theta_{L_1}^F + \gamma\cdot\Theta_{L_1}^{Trans} \quad (2)$$

In order to retain the query focus during back translation, the feedback model in L_2 is interpolated with the translated query before translation of the L_2 feedback model. The parameters β and γ control the relative importance of the original query model, feedback model of L_1 and the translated feedback model obtained from L_1 and are tuned based on the choice of L_1 and L_2.

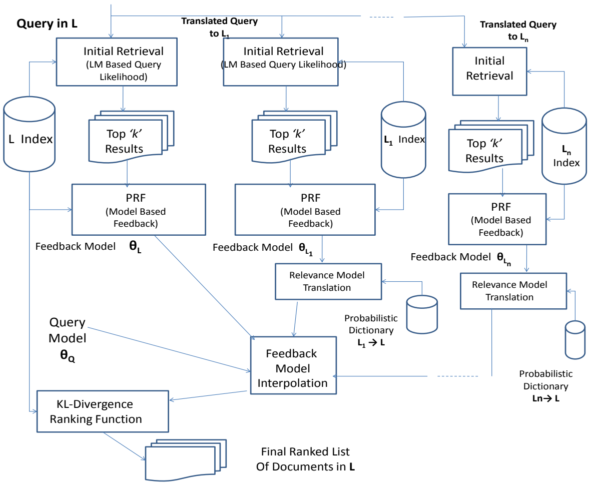

Query in L
Translated Query to L_1
Translated Query to L_n
Initial Retrieval (LM Based Query Likelihood)
Initial Retrieval (LM Based Query Likelihood)
Initial Retrieval
L Index
Top 'k' Results
Top 'k' Results
L_1 Index
Top 'k' Results
L_n Index
PRF (Model Based Feedback)
PRF (Model Based Feedback)
PRF (Model Based Feedback)
Feedback Model θ_L
Feedback Model θ_{L_1}
Feedback Model θ_{L_n}
Relevance Model Translation
Relevance Model Translation
Probabilistic Dictionary $L_1 \rightarrow L$
Query Model θ_Q
Feedback Model Interpolation
Probabilistic Dictionary $Ln \rightarrow L$
KL-Divergence Ranking Function
Final Ranked List Of Documents in L

Figure 1: Schematic of the Multilingual PRF Approach Using Parallel Assistance

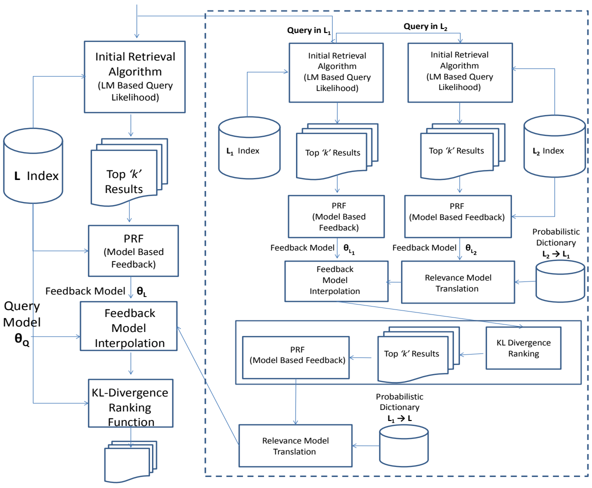

Query in L_1
Query in L_2
Initial Retrieval Algorithm (LM Based Query Likelihood)
Initial Retrieval Algorithm (LM Based Query Likelihood)
Initial Retrieval Algorithm (LM Based Query Likelihood)
L Index
Top 'k' Results
L_1 Index
Top 'k' Results
Top 'k' Results
L_2 Index
PRF (Model Based Feedback)
PRF (Model Based Feedback)
PRF (Model Based Feedback)
Probabilistic Dictionary $L_2 \rightarrow L_1$
Feedback Model θ_{L_1}
Feedback Model θ_{L_2}
Feedback Model Interpolation
Relevance Model Translation
Feedback Model θ_L
Query Model θ_Q
Feedback Model Interpolation
PRF (Model Based Feedback)
Top 'k' Results
KL Divergence Ranking
KL-Divergence Ranking Function
Probabilistic Dictionary $L_1 \rightarrow L$
Relevance Model Translation

Figure 2: Schematic of the Multilingual PRF Approach Using Serial Assistance

4 Extending MultiPRF to Multiple Assisting Languages

In this section, we extend the MultiPRF model described earlier to multiple assisting languages. Since each language produces a different feedback model, there could be different ways of combining these models as suggested below.

Parallel: One way is to include the new assisting language model using one more interpolation coefficient which gives the effect of using multiple assisting languages in *parallel*.

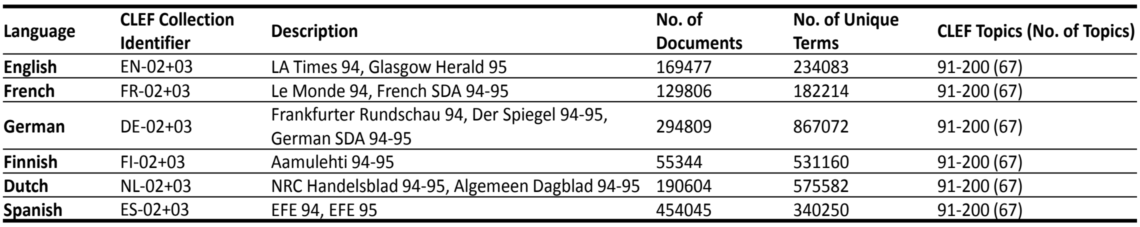
Serial: Alternately, we can have a *serial* combination wherein language L_2 is first assisted by language L_3 and then this MultiPRF system is used to assist the source language L_1.

Selective: Finally, we can have selective assistance wherein we dynamically select which assisting language to use based on the input query.

Below we describe each of these systems in detail.

4.1 Parallel Combination

The MultiPRF model as explained in section 3 interpolates the query model of L_1 with the MBF of L_1 and the translated feedback model of the assisting language L_2. The most natural extension to this approach is to translate the query into multiple languages instead of a single language and collect the feedback terms from the initial re-



Language	CLEF Collection Identifier	Description	No. of Documents	No. of Unique Terms	CLEF Topics (No. of Topics)
English	EN-02+03	LA Times 94, Glasgow Herald 95	169477	234083	91-200 (67)
French	FR-02+03	Le Monde 94, French SDA 94-95	129806	182214	91-200 (67)
German	DE-02+03	Frankfurter Rundschau 94, Der Spiegel 94-95, German SDA 94-95	294809	867072	91-200 (67)
Finnish	FI-02+03	Aamulehti 94-95	55344	531160	91-200 (67)
Dutch	NL-02+03	NRC Handelsblad 94-95, Algemeen Dagblad 94-95	190604	575582	91-200 (67)
Spanish	ES-02+03	EFE 94, EFE 95	454045	340250	91-200 (67)

Table 2: Details of the CLEF Datasets used for Evaluating the MultiPRF approach. The number shown in brackets of the final column CLEF Topics indicate the actual number of topics used during evaluation.

trieval of each of these languages. The translated feedback models resulting from each of these retrievals can then be interpolated to get the final parallel MultiPRF model. Specifically, if L_1 is the source language and $L_2, L_3, \ldots L_n$ are assisting languages then final parallel MultiPRF model can be obtained by generalizing Equation 2 as shown below:

$$\Theta_{L_1}^{MultiAssist} = (1 - \beta - \sum_i \alpha_i) \cdot \Theta_Q + \beta \cdot \Theta_F + \sum_i \alpha_i \cdot \Theta_{L_i}^{Trans} \tag{3}$$

The schematic representation of parallel combination is shown in Figure 1.

4.2 Serial Combination

Let L_1 be the source language and let L_2 and L_3 be two assisting languages. A serial combination can then be achieved by cascading two MultiPRF systems as described below:

1. Construct a MultiPRF system with L_2 as the source language and L_3 as the assisting language. We call this system as L_2L_3-MultiPRF system.

2. Next, construct a MultiPRF system with L_1 as the source language and L_2L_3-MultiPRF as the assisting system.

As compared to a single assistance system where only L_2 is used as the assisting language for L_1, here the performance of language L_2 is first boosted using L_3 as the assisting language. This boosted system is then used for assisting L_1. Also note that unlike parallel assistance here we do not introduce an extra interpolation co-efficient in the original MultiPRF model given in Equation 2. The schematic representation of serial combination is shown in Figure 2.

4.3 Selective Assistance

We motivate selective assistance by posing the following question: *"Given a source language L_1 and two assisting languages L_2 and L_3, is it possible that L_2 is ideal for assisting some queries whereas L_3 is ideal for assisting some other queries?"* For example, suppose L_2 has a rich collection of TOURISM documents whereas L_3 has a rich collection of HEALTH documents. Now, given a query pertaining to TOURISM domain one might expect L_2 to serve as a better assisting language whereas given a query pertaining to the HEALTH domain one might expect L_3 to serve as a better assisting language. This intuition can be captured by suitably changing the interpolation model as shown below:

$$\Theta_L^{Best} = SelectBestModel(\Theta_L^F, \Theta_{L_1}^{Trans}, \Theta_{L_2}^{Trans}, \Theta_{L_{12}}^{Trans})$$

$$\Theta_{L_1}^{Multi} = (1 - \alpha) \cdot \Theta_Q + \alpha \cdot \Theta_L^{Best} \tag{4}$$

where, $SelectBestModel()$ gives the best model for a particular query using the algorithm mentioned below which is based on minimizing the query drift as described in (**?**):

1. Obtain the four feedback models, *viz.*, $\Theta_L^F, \Theta_{L_1}^{Trans}, \Theta_{L_2}^{Trans}, \Theta_{L_{12}}^{Trans}$
2. Build a language model (say, LM) using query Q and top-100 documents of initial retrieval in language L.
3. Find the KL-Divergence between LM and the four models obtained during step 1.
4. Select the model which has minimum KL-Divergence score from LM. Call this model Θ_L^{Best}.
5. Get the final model by interpolating the query model, Θ_Q, with Θ_L^{Best}.

5 Experimental Setup

We evaluate the performance of our system using the standard CLEF evaluation data in six languages, widely varying in their familial relationships - Dutch, German, English, French, Spanish and Finnish. The details of the collections and their corresponding topics used for MultiPRF are given in Table 2. Note that, in each experiment, we choose assisting collections such that the topics in the source language are covered in the assisting collection so as to get meaningful feedback terms. In all the topics, we only use the *title* field. We ignore the topics which have no relevant documents as the true performance on those topics cannot be evaluated.

We use the Terrier IR platform (Ounis et al., 2005) for indexing the documents. We perform standard tokenization, stop word removal and stemming. We use the Porter Stemmer for English and the stemmers available through the Snowball package for other languages. Other than these, we do not perform any language-specific processing on the languages. In case of French, since some function words like *l'*, *d' etc.,* occur as prefixes to a word, we strip them off during indexing and query processing, since it significantly improves the baseline performance. We use standard evaluation measures like *MAP*, *P@5* and *P@10* for evaluation. Additionally, for assessing robustness, we use the Geometric Mean Average Precision (GMAP) metric (Robertson, 2006) which is also used in the TREC Robust Track (Voorhees, 2006). The probabilistic bi-lingual dictionary used in MultiPRF was learnt automatically by running GIZA++: a word alignment tool (Och and Ney, 2003) on a parallel sentence aligned corpora. For all the above language pairs we used the *Europarl Corpus* (Philipp, 2005). We use Google Translate as the query translation system as it has been shown to perform well for the task (Wu et al., 2008). We use two-stage Dirichlet smoothing with the optimal parameters tuned based on the collection (Zhai and Lafferty, 2004). We tune the parameters of MBF, specifically λ and α, and choose the values which give the optimal performance on a given collection. We observe that the optimal parameters γ and β are uniform across collections and vary in the range 0.4-0.48. We uniformly choose the top ten documents for feedback.

Source Langs	**Assist. Langs**		**MBF**	**MultiPRF** (L_1)	**MultiPRF** (L_2)	**MultiPRF** (L_1,L_2)
EN	DE-NL	MAP	0.4495	0.4464	0.4471	**0.4885(4.8)**†
		P@5	0.4955	0.4925	0.5045	0.5164(2.4)
		P@10	0.4328	0.4343	0.4373	0.4463(2.1)
	DE-FI	MAP	0.4495	0.4464	0.4545	**0.4713(3.7)**†
		P@5	0.4955	0.4925	0.5194	0.5224(1.2)
		P@10	0.4328	0.4343	0.4373	0.4507(3.1)
	NL-ES	MAP	0.4495	0.4471	0.4566	**0.4757(4.2)**†
		P@5	0.4955	0.5045	0.5164	0.5224(0.6)
		P@10	0.4328	0.4373	0.4537	0.4448(2.4)
	ES-FR	MAP	0.4495	0.4566	0.4563	**0.48(5.1)**†
		P@5	0.4955	0.5164	0.5075	0.5224(1.2)
		P@10	0.4328	0.4537	0.4343	0.4388(-3.3)
	ES-FI	MAP	0.4495	0.4566	0.4545	**0.48(5.1)**†
		P@5	0.4955	0.5164	0.5194	0.5254(1.7)
		P@10	0.4328	0.4537	0.4373	0.4403(-3.0)
	FR-FI	MAP	0.4495	0.4563	0.4545	0.4774(4.6)
		P@5	0.4955	0.5075	0.5194	**0.5284(4.1)**†
		P@10	0.4328	0.4343	0.4373	0.4373(0.7)
FI	EN-FR	MAP	0.3578	0.3411	0.3553	0.3688(3.8)
		P@5	0.3821	0.394	0.397	**0.4149(4.5)**†
		P@10	0.3105	0.3463	0.3433	0.3433(0.1)
	NL-DE	MAP	0.3578	0.3722	0.3796	0.3929(3.5)
		P@5	0.3821	0.406	0.403	0.4149(3.0)
		P@10	0.3105	0.3478	0.3582	0.3597(0.4)
	ES-DE	MAP	0.3578	0.369	0.3796	**0.4058(6.9)**†
		P@5	0.3821	0.4119	0.403	0.4239(5.2)
		P@10	0.3105	0.3448	0.3582	0.3612(0.8)
	FR-DE	MAP	0.3578	0.3553	0.3796	**0.3988(5.1)**†
		P@5	0.3821	0.397	0.403	0.406(0.7)
		P@10	0.3105	0.3433	0.3582	0.3507(-2.1)
	NL-ES	MAP	0.3578	0.3722	0.369	**0.3875(4.1)**†
		P@5	0.3821	0.406	0.4119	0.4060.0)
		P@10	0.3105	0.3478	0.3448	0.3537(1.7)
	NL-FR	MAP	0.3578	0.3722	0.3553	**0.3875(4.1)**†
		P@5	0.3821	0.406	0.397	0.409(0.7)
		P@10	0.3105	0.3478	0.3433	0.3463(-0.4)
	ES-FR	MAP	0.3578	0.369	0.3553	0.3823(3.6)
		P@5	0.3821	0.4119	0.397	0.4119(0.0)
		P@10	0.3105	0.3448	0.3433	0.3418(-0.9)
FR	EN-ES	MAP	0.4356	0.4658	0.4634	0.4803(3.1)
		P@5	0.4776	0.4925	0.4925	0.4985(1.2)
		P@10	0.4194	0.4358	0.4388	**0.4493(3.1)**†

Table 3: Comparison of MultiPRF Multiple Assisting Languages using parallel assistance framework with MultiPRF with single assisting language. Only language pairs where positive improvements were obtained are reported here. Results marked as ‡ indicate that the improvement was statistically significant over baseline (Maximum of MultiPRF with single assisting language) at 90% confidence level ($\alpha = 0.01$) when tested using a paired two-tailed t-test.

6 Results and Discussion

Tables **??** and **??** present the results for MultiPRF with two assisting languages using parallel assistance and selective assistance framework. Out of the total 60 possible combinations, in Table **??**, we only report the combinations where we have obtained positive improvements greater than 3%. We observe most improvements in English, Finnish and French. We did not observe any improvements using the serial assistance framework over MultiPRF with single assisting lan-

Source Langs	Assist. Langs		Parallel Model	Selective Model
EN	DE-NL	MAP	0.4651	**0.4848**
		P@5	0.5254	0.5224
		P@10	0.4493	**0.4522**
	NL-FI	MAP	0.4387	**0.4502**
		P@5	0.5015	**0.5164**
		P@10	0.4284	**0.4358**
DE	EN-FR	MAP	0.4097	**0.4302**
		P@5	0.594	0.5851
		P@10	0.5149	**0.5179**
	FR-ES	MAP	0.4215	**0.4333**
		P@5	0.591	0.591
		P@10	0.5239	0.5209
	FR-NL	MAP	0.4139	**0.4236**
		P@5	0.5701	0.5701
		P@10	0.5075	**0.5134**
	FR-FI	MAP	0.3925	**0.4055**
		P@5	0.5101	**0.5642**
		P@10	0.4851	**0.5**
	NL-FI	MAP	0.3974	**0.4192**
		P@5	0.5731	0.5612
		P@10	0.497	**0.503**
ES	EN-FI	MAP	0.4436	**0.4501**
		P@5	0.6179	**0.6269**
		P@10	0.5567	**0.5657**
	DE-FI	MAP	0.4542	**0.465**
		P@5	0.6269	0.6179
		P@10	0.5627	0.5582
	NL-FI	MAP	0.4531	**0.4611**
		P@5	0.6269	**0.6299**
		P@10	0.5627	0.5627

Table 4: Results showing the positive improvements of MultiPRF with selective assistance framework over MultiPRF with parallel assistance framework.

guage. Hence, we do not report their results as the results were almost equivalent to single assisting language. As shown in Table **??**, selective assistance does give decent improvements in some language pairs. An interesting point to note in selective assistance is that it helps languages like Spanish whose monolingual performance and document coverage are both high.

6.1 Qualitative Comparison of Feedback Terms using Multiple Languages

In this section, we qualitatively compare the results of MultiPRF with two assisting languages with that of MultiPRF with single assisting language, based on the top feedback terms obtained by each model. Specifically, in Table 5 we compare the terms obtained by MultiPRF using (i) Only L_1 as assisting language, (ii) Only L_2 as assisting language and (iii) Both L_1 and L_2 as assisting languages in a parallel combination. For example, the first row in the above table shows the terms obtained by each model for the English query *"Golden Globes 1994"*. Here, L_1 is French and L_2 is Spanish. Terms like *"Gold"* and *"Prize"* appearing in the translated feedback model of L_1 cause a drift in the topic towards *"Gold Prize"* resulting in a lower MAP score (0.33). Similarly, the terms like *"forrest"* and *"spielberg"* appearing in the translated feedback model of L_2 cause a drift in topic towards *Forrest Gump and Spielberg Oscars* resulting in a MAP score (0.5). However, when the models from two languages are combined, terms which cause a topic drift get ranked lower and as a result the focus of the query is wrenched back. A similar observation was made for the English query *"Damages in Ozone Layer"* using French (L_1) and Spanish (L_2) as assisting languages. Here, terms from the translated feedback model of L_1 cause a drift in topic towards *"militri bacteria"* whereas the terms from the translated feedback model of L_2 cause a drift in topic towards *"iraq war"*. However, in the combined model these terms get lower rank there by bringing back the focus of the query. For the Finnish query *"Lasten oikeudet"* (Children's Rights), in German (L_1), the topic drift is introduced by terms like *"las, gram, yhteis"*. In case of Dutch (L_2), the query drift is caused by *"mandy, richard, slovakia"* (L_2) and in the case of combined model, these terms get less weightage and the relevant terms like *"laps, oikeuks, vanhemp"* which are common in both models, receive higher weightage causing an improvement in query performance.

Next, we look at a few negative examples where the parallel combination actually performs poorer than the individual models. This happens when some *drift-terms* (*i.e.,* terms which can cause topic drift) get mutually reinforced by both the models. For example, for the German query *"Konkurs der Baring-Bank"* (Bankruptcy of Baring Bank) the term *"share market"* which was actually ranked lower in the individual models gets boosted in the combined model resulting in a drift in topic. Similarly, for the German query *"Ehren-Oscar für italienische Regisseure"* (Honorary Oscar for Italian directors) the term *"head office"* which was actually ranked lower in the individual models gets ranked higher in the combined model due to mutual reinforcement resulting in a topic drift.

TOPIC NO.	QUERIES (Meaning in Eng.)	TRANSLATED ENGLISH QUERIES (Assisting Lang.)	L1 MAP	L2 MAP	L1-L2 MAP	Representative Terms with L1 as Single Assisting Language (With Meaning)	Representative Terms with L2 as Single Assisting Language (With Meaning)	Representative Terms with L1& L2 as Assisting Langs. (With Meaning)
English '03 TOPIC 165	Globes 1994	Golden Globes 1994 (FR) Globos de Oro 1994 (ES)	0.33	0.5	1	Gold, prize, oscar, nomin, best award, hollywood, actor, director ,actress, world, won ,list, winner, televi, foreign ,year, press	world, nomin, film, award, delici, planet, earth, actress, list, drama, director, actor, spielberg, music, movie, forrest, hank	oscar, nomin, best, award, hollywood actor, director, cinema, televi, music, actress, drama, role, hank, foreign, gold
Finnish '03 TOPIC 152	Lasten oikeudet (Children's Rights)	Rechte des Kindes (DE) Kinderrechten (NL)	0.2	0.25	0.37	laps (child), oikeuks (rights), oikeud (rights), kind, oikeus (right), isä (father), oikeut (justify), vanhemp (parent), vanhem (parents), las, gram, yhteis, unicef, sunt, äiti(mother), yleissopimnks(conventions)	oikeuks (rights), laps (child), oikeud (right), mandy, richard, slovakia, tähänast (to date), tuomar (judge), tyto, kid, , nuor (young people), nuort (young), sano(said) , perustam(establishing)	laps (child), oikeuks (rights), oikeud (rights), oikeus (right), isä (father, parent), vanhemp (parent), vanhem (parents), oikeut (justify), las, mandy, nuort (young), richard, nuor (young people), slovakia, tähänast (to date),
English '03 TOPIC 148	Damages in Ozone Layer	Dommages à la couche d'ozone (FR) Destrucción de la capa de ozono (ES)	0.08	0.07	0.2	damag, militri, uv, layer, condition, chemic, bacteria, ban, radiat, ultraviolet	damag, weather, atmospher, earth, problem, report, research, harm, iraq, war, scandal, illigel, latin, hair	damag, uv, layer,weather, atmospher, earth, problem, report, research , utraviolet, chemic
German '03 TOPIC 180	Konkurs der Baring-Bank (Bankruptcy of Baring Bank)	Bankruptcy of Barings (EN) Baringsin Konkurssi (FI)	0.55	0.51	0.33	zentralbank(central bank),bankrott(bank cruptcy), investitionsbank, sigapur, london , britisch, index, tokio, england, werbung(advertising), japan	fall, konkurs, bankrott(Bankruptcy), warnsignal(warning), ignoriert, zusammenbruch(collepse), london, singapur, britisch(british), dollar, tokio, druck(pressur), handel(trade)	aktienmarkt(share market), investitionsbank, bankrott, zentralbank(central bank), federal, singapur, london, britisch, index, tokio, dollar, druck, england, dokument(document)
German '03 TOPIC 198	Ehren-Oscar für italienische Regisseure (Honorary Oscar for Italian directors)	Honorary Oscar for Italian Directors (EN) Kunnia-Oscar italialaisille elokuvaohjaajille (FI)	0.5	0.35	0.2	Direktor(director), film, regierungschef(prime) , best antonionis, antonionins, lieb, geschicht(history) , paris, preis, berlin, monitor, kamera	Generaldirektion(General director), film, ehrenmitglied, regisseur, direktor, verleih , itali, oscar, award, antonionins	generaldirektion(head office), ehrenmitglied(honorable member), regierungschef(prime), regisseur(director),oscar, genossenschaftsbank (corporate bank)

Table 5: Qualitative Comparison of MultiPRF Results using two assisting languages with single assisting language.

6.2 Effect of Coverage on MultiPRF Accuracy

A study of the results obtained for MultiPRF using single assisting language and multiple assisting languages with different source languages showed that certain languages are more suited to be benefited by assisting languages. In particular, languages having smaller collections are more likely to be benefited if assisted by a language having a larger collection size. For example, Finnish which has the smallest collection (55344 documents) showed maximum improvement when supported by assisting language(s). Based on this observation, we plotted a graph of the collection size of a source language v/s the average improvement obtained by using two assisting languages to see if their exists a correlation between these two factors. As shown in Figure 3, there indeed exists a high correlation between these two entities. At one extreme, we have a language like Spanish which has the largest collection (454045 documents) and is not benefited much by assisting languages. On the other extreme, we have Finnish which has the smallest collection size and is benefited most by assisting languages.

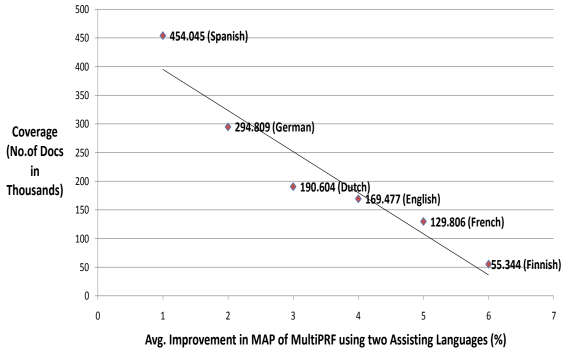


Figure 3: Effect of Coverage on Average MultiPRF MAP using Two Assisting Languages.

6.3 Effect of Number of Assisting Languages on MultiPRF Accuracy

Another interesting question which needs to be addressed is *"Whether it helps to use more than two assisting languages?"* and if so *"Is there an optimum number of assisting languages beyond which there will be no improvement?"*. To answer these questions, we performed experiments using 1-4 assisting languages with each source language. As seen in Figure 4, in general as the number of assisting languages increases the performance saturates (typically after 3 languages). Thus, for 5 out of the 6 source languages, the performance saturates after 3 languages which is in line with what we would intuitively expect. However, in the case of German, on an average, the

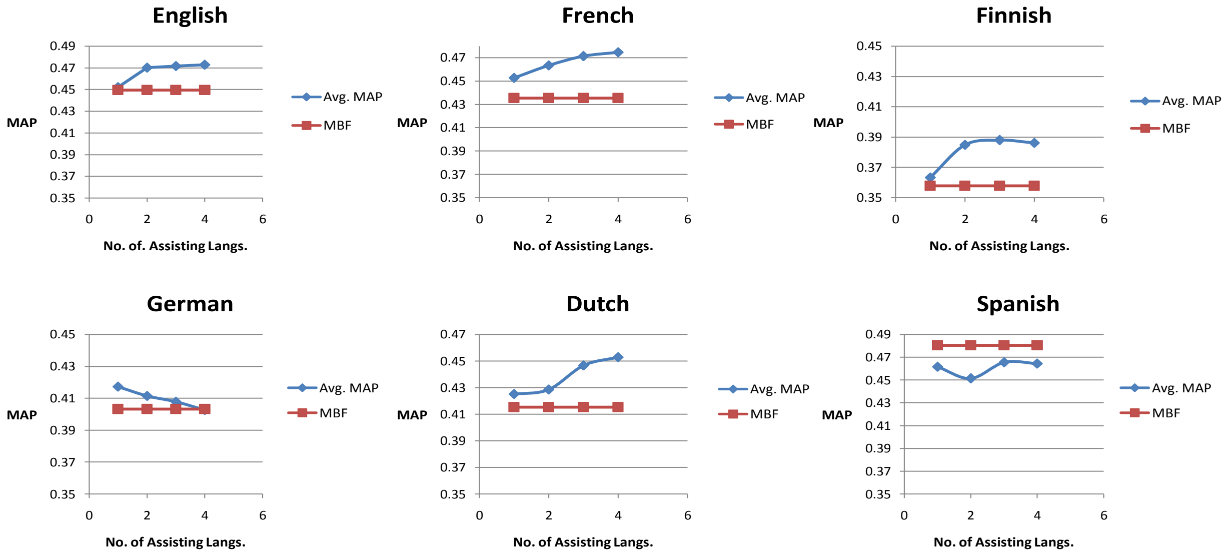


Figure 4: Effect of Number of Assisting Languages on Avg. MultiPRF Performance with Multiple Assistance.

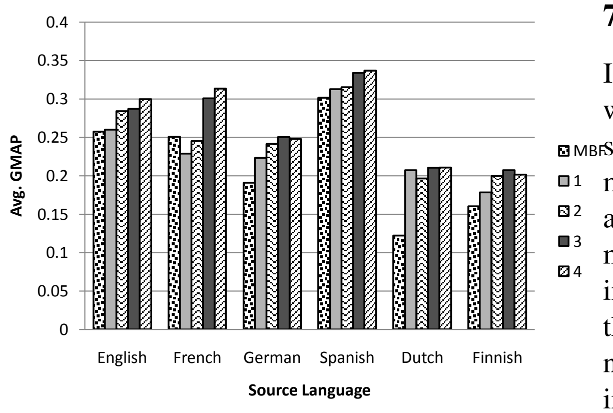


Figure 5: Effect of Number of Assisting Languages on Robustness measured through GMAP.

performance drops as the number of assisting languages is increased. This drop is counter intuitive and needs further investigation.

6.4 Effect of Number of Assisting Languages on Robustness

One of the primary motivations for including multiple assisting languages in MultiPRF was to increase the robustness of retrieval through better coverage. We varied the number of assisting languages for each source and studied the average GMAP. The results are shown in Figure 5. We observe that in almost all the source languages, the GMAP value increases with number of assisting languages and then reaches a saturation after reaching three languages.

7 Conclusion

In this paper, we extended the MultiPRF framework to multiple assisting languages. We presented three different configurations for including multiple assisting languages - a) Parallel b) Serial and c) Selective. We observe that the results are mixed with parallel and selective assistance showing improvements in some cases. We also observe that the robustness of MultiPRF increases with number of assisting languages. We analyzed the influence of coverage of MultiPRF accuracy and observed that it is inversely correlated. Finally, increasing the number of assisting languages increases the MultiPRF accuracy to some extent and then it saturates beyond that limit. Many of the above results (negative results of serial, selective configurations etc.) require deeper investigation which we plan to take up in future.

References

Braschler, Martin and Carol Peters. 2004. Cross-language evaluation forum: Objectives, results, achievements. *Inf. Retr.*, 7(1-2):7–31.

Buckley, Chris, Gerald Salton, James Allan, and Amit Singhal. 1994. Automatic query expansion using smart : Trec 3. In *Proceedings of The Third Text REtrieval Conference (TREC-3*, pages 69–80.

Chinnakotla, Manoj K., Karthik Raman, and Pushpak Bhattacharyya. 2010a. Multilingual pseudo-relevance feedback: English lends a helping hand. In *ACM SIGIR 2010*, Geneva, Switzerland, July. ACM.

Chinnakotla, Manoj K., Karthik Raman, and Pushpak Bhattacharyya. 2010b. Multilingual pseudo-relevance feedback: Performance study of assisting languages. In *ACL 2010*, Uppsala, Sweeden, July. ACL.

Dempster, A., N. Laird, and D. Rubin. 1977. Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society*, 39:1–38.

Hawking, David, Paul Thistlewaite, and Donna Harman. 1999. Scaling up the trec collection. *Inf. Retr.*, 1(1-2):115–137.

John Lafferty and Chengxiang Zhai. 2003. Probabilistic Relevance Models Based on Document and Query Generation. In *Language Modeling for Information Retrieval*, volume 13, pages 1–10. Kluwer International Series on IR.

Mitra, Mandar, Amit Singhal, and Chris Buckley. 1998. Improving automatic query expansion. In *SIGIR '98: Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 206–214, New York, NY, USA. ACM.

Och, Franz Josef and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51.

Ounis, I., G. Amati, Plachouras V., B. He, C. Macdonald, and Johnson. 2005. Terrier Information Retrieval Platform. In *Proceedings of the 27th European Conference on IR Research (ECIR 2005)*, volume 3408 of *Lecture Notes in Computer Science*, pages 517–519. Springer.

Philipp, Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. In *MT Summit*.

Robertson, Stephen. 2006. On gmap: and other transformations. In *CIKM '06: Proceedings of the 15th ACM international conference on Information and knowledge management*, pages 78–83, New York, NY, USA. ACM.

Voorhees, Ellen. 2006. Overview of the trec 2005 robust retrieval track. In *E. M. Voorhees and L. P. Buckland, editors, The Fourteenth Text REtrieval Conference, TREC 2005*, Gaithersburg, MD. NIST.

Wu, Dan, Daqing He, Heng Ji, and Ralph Grishman. 2008. A study of using an out-of-box commercial mt system for query translation in clir. In *iNEWS '08: Proceeding of the 2nd ACM workshop on Improving non english web searching*, pages 71–76, New York, NY, USA. ACM.

Xu, Jinxi and W. Bruce Croft. 2000. Improving the effectiveness of information retrieval with local context analysis. *ACM Trans. Inf. Syst.*, 18(1):79–112.

Zhai, Chengxiang and John Lafferty. 2001. Model-based Feedback in the Language Modeling approach to Information Retrieval. In *CIKM '01: Proceedings of the tenth international conference on Information and knowledge management*, pages 403–410, New York, NY, USA. ACM Press.

Zhai, Chengxiang and John Lafferty. 2004. A Study of Smoothing Methods for Language Models applied to Information Retrieval. *ACM Transactions on Information Systems*, 22(2):179–214.

Multilinguization and Personalization of NL-based Systems

Najeh Hajlaoui
GETALP, LIG, UJF
385 rue de la Bibliothèque, BP n° 53
38041 Grenoble, cedex 9, France
Najeh.Hajlaoui@imag.fr

Christian Boitet
GETALP, LIG, UJF
385 rue de la Bibliothèque, BP n° 53
38041 Grenoble, cedex 9, France
Christian.Boitet@imag.fr

Abstract

Linguistic porting of content management services processing spontaneous utterances in natural language has become important. In most situations, such utterances are noisy, but are constrained by the situation, thus constituting a restricted *sublangage*. In previous papers, we have presented three methods to port such systems to other languages. In this paper, we study how to also *personalize* them by making them capable of *automatic perception adaptation,* using fuzzy evaluation functions. We have reengineered IMRS, a music retrieval NL-based system, to implement that idea, and ported it to French, English and Arabic using an enhanced version of our *external porting* method, building a *unique* content extractor for these three languages. More than 30 persons participated in a preliminary on-line qualitative evaluation of the system.

1 Introduction

Multilingualizing systems handling content expressed in spontaneous natural language is an important but difficult problem, and very few multilingual services are available today. The choice of a particular multilingualization process depends on the translational situation: types and levels of possible accesses, available resources, and linguistic competences of participants involved in the multilingualization task. Three main strategies are possible in principle for multilingualization, by translation, and by internal or external adaptation. We consider here the subproblem of *linguistic porting,* where the content is adapted to another language, but not necessarily to a different cultural environment. We also try to add some level of *personalization*, by *automatic perception adaptation,* based on the use of fuzzy evaluation functions. We use the example of IMRS, an *Impression-based Music-Retrieval System* (Kumamoto, 2004), with a native interface in Japanese, which we have reengineered and ported to French, English and Arabic. The context and objectives of our work are presented in the second section. The third section presents the IMRS original prototype and the possible strategies to achieve porting and personalization. In the fourth section, we give detailed specifications of our reengineered music retrieval system, IMRS-g. In the fifth section, we present the implementation of five music retrieval modes. Finally, we report on the multilingual porting of this system.

2 Methods for porting NL-based content processing systems

The choice of a method for multilingualizing e-commerce services based on content extraction from spontaneous texts depends on two aspects of the translational situation:

- The level of access to resources of the initial application. Four cases are possible: complete access to the source code, access limited to the internal representation, access limited to the dictionary, and no access. In the case of IMRS, the access was limited to the internal representation, visible as a non-linguistic interface in the original prototype (a set of 10 impressions manipulate by a set of 7 checkbox).
- The linguistic qualification level of the persons involved in the process (level of know-

ledge of the source language, competence in NLP) and the resources (corpora, dictionaries) available for the new language(s), in particular for the *sublanguages* at hand.

We concentrate on NLP-based systems that perform specific tasks in restricted domains. Figure 1 shows the general structure of these systems. Examples of such applications and services are: categorization of various documents such as AFP (Agence France Presse) flash reports or customer messages on an ASS (After Sale Service) server, and information extraction to feed or consult a database (e.g. classified ads, FAQ, automated hotlines).

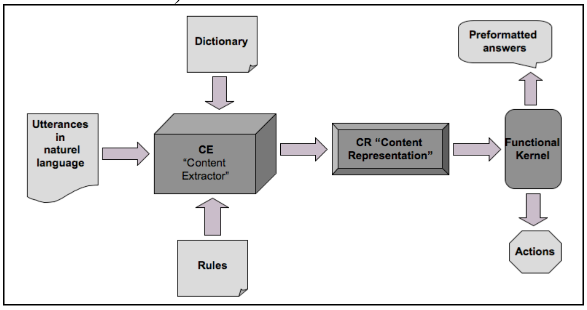
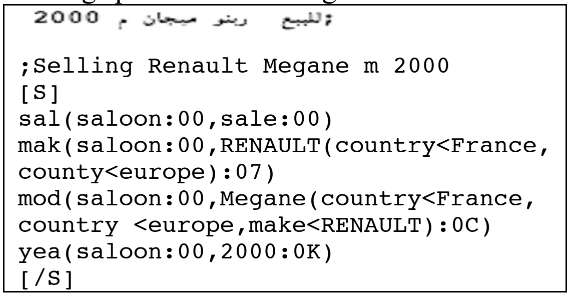


Figure 1: general structure of an NLP-based CMS

We first studied *linguistic porting* of e-commerce systems handling spontaneous utterances in natural languages, that are often noisy, but constrained by the situation, and constitute a more or less restricted *sub-language* (Kittredge, 1982), (Harris, 1968) (Grishman and Kittredge, 1986).

This kind of system uses a specific content representation on which the functional kernel works. In most cases, this content representation is generated from the *native* language L1 by a content extractor. In our PhD, we have identified three possible methods of linguistic porting, and have illustrated them by porting to French CATS (Daoud, 2006), a Classified Ads Transaction System in SMS (Arabic) deployed in Amman on Fastlink, as well as IMRS, mentioned above. The three possible strategies for linguistic porting are internal porting, external porting and porting by machine translation. Figure 2 shows an example of the car domain with the output of the content extractor (CRL-CATS).

In CRL-CATS (Content Representation Language for CATS), a posted SMS is represented as a set of binary relations between objects. It is a kind of semantic graph with a UNL-like syntax (Uchida and Zhu 2005-2006). There are no variables, but the dictionary is used as a type lattice allowing specialization and generalization.



```
;للبيع رينو ميجان م 2000

;Selling Renault Megane m 2000
[S]
sal(saloon:00,sale:00)
mak(saloon:00,RENAULT(country<France,
county<europe):07)
mod(saloon:00,Megane(country<France,
country <europe,make<RENAULT):0C)
yea(saloon:00,2000:0K)
[/S]
```

Figure 2: Example of SMS

2.1 Internal porting

The first possibility consists in adapting the original content extractor of the application from L1 to the target language L2 (see Figure 3); but that is viable only if :

- the developers agree to open their code and tools,
- the code and tools are relatively easy to understand,
- the resources are not too heavy to create (in particular the dictionary).

That method requires of course training the localization team with the tools and methods used.

Under these conditions, adaptation can be done at a very reasonable cost, and further maintenance.

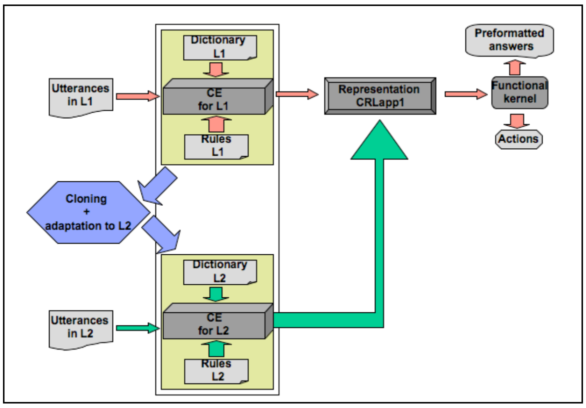


Figure 3: internal porting

We have previously experimented this method (Hajlaoui, 2008) by porting *CATS* from Arabic to French: for that, we adapted its native Arabic content extractor, written in EnCo[1] (Uchida and Zhu 1999), by translating its dictionary, and modifying a few analysis rules.

[1] EnCo is a tool based on rules and dictionaries used for content extraction in original version of CATS system.

2.2 External porting

If there is access only to the internal content representation, the solution consists in adapting an available content extractor for L2 to the sublanguage at hand, and to compile its results into the original content representation (see Figure 4).

For a company wanting to offer multilingualization services, it would indeed be an ideal situation to have a generic content extractor, and to adapt it to each situation (language, sublanguage, domain, content representation, task, other constraints). However, there is still no known generic content extractor of that power, and not even a generic content extractor for particular languages, so that this approach cannot be considered at present. Our approach is then to adapt an existing content extractor, developed for L2 and a different domain/task, or for another language and the same domain/task.

We also applied this method to port CATS from Arabic to French, and experimentation are described in (Hajlaoui, 2008).

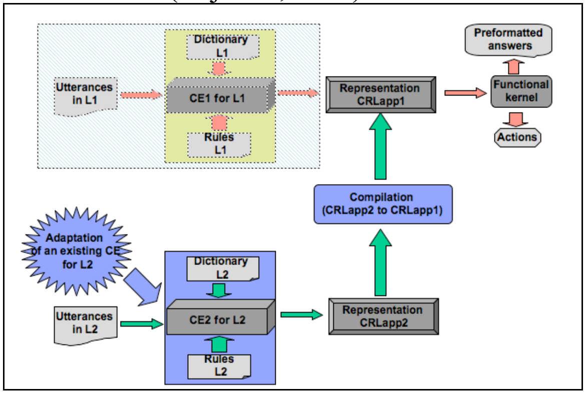


Figure 4: external porting

2.3 Porting by machine translation

If there is no access to the code, dictionary, and internal content representation of the original application, the only possible approach to port it from L1 to L2 is to develop an MT system to automatically translate its (spontaneous) inputs from L2 into L1 (see Figure 5).

Porting CATS from Arabic to French by statistical translation gave a very good performance, and that with a very small training corpus (less than 10 000 words). This proves that, in the case of very small sub-languages, statistical translation may be of sufficient quality, starting from a corpus 100 to 500 smaller than for the general language.

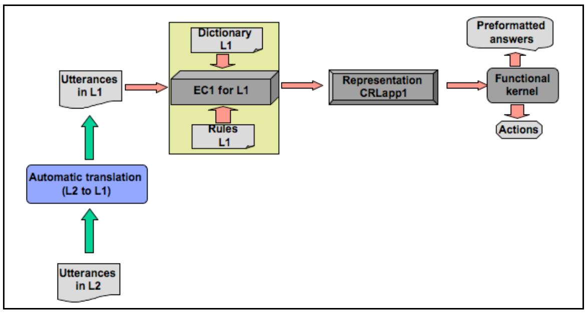


Figure 5: porting by machine translation

2.4 Results and evaluation

We translated manually the evaluation corpus used for the evaluation of CATS Arabic version. It contains 200 real SMS (100 SMS to buy + 100 SMS to sale) posted by real users in Jordan.

We spent 289 mn to translate the 200 Arabic SMS (2082 words is equivalent to 10 words/SMS, approximately 8 standard pages[2]) into a French translation, or about 35 mn per page, and 10 mn per standard page to pass from raw translation to functional translation.

We obtained 200 French SMS considered to be functional (1361 words, or about 6,8 words/SMS, approximately 5 standard pages). We then computed the recall R, the precision P and the F-measure F for each most important property (action "sale or buy", "make", "model", "year", "price").

P = |correct entities identified by the system| / |entities identified by the system|;

R = |correct entities identified by the system| / |entities identified by the human|;

F-measure = 2PR/(P+R)

Table 1 summarizes the percentage (F-measure ratio) of the Arabic-French porting of CATS and shows details in (Hajlaoui, 2008). Properties having numbers as values, like price and year, lower the percentage of porting by external porting, but the advantage is that method requires only accessing the internal representation of the application.

	Minimum	Average	Maximum
Internal porting	95%	98%	100%
External porting	46%	77%	99%
Porting by statistical translation	85%	93%	98%

Table 1: evaluation of three methods used for porting CATS_Cars from Arabic to French.

[2] Standard page = 250 words

In the third part of this article, we describe the multilinguization of IMRS, IMRS-g, which includes a module of queries management, where the queries are expressed either in a natural language or in a graphical interface showing 10 vectors corresponding to the internal content representation. In response to a query, the user receives a set of music pieces that correspond to her/his desired selection criteria.

In addition to the original design, where the NL expressions of the 10 measures are mapped in a fixed way to the integers in the real interval [1, 7], we have tried to apply a small part of the theory of fuzzy sets to improve the representation and evaluation of human perceptions.

3 Multilinguization of IMRS

To port IMRS to several languages, we used the external porting method and we built a new content extractor, which treats simple utterances related to the music domain.

3.1 IMRS

IMRS (Kumamoto and Ohta, 2003) is a non-deployed Web service prototype, developed as an experimental base for a PhD. It allows to retrieve music pieces either by using Japanese queries, or by manipulating a graphical interface with 10 criteria settable by knobs (speed, noise, rhythm...), and showing remarkable values (integers between 1 and 7) expressed by English labels. In IMRS, an utterance processed by the system is a spontaneous sentence or fragment of a sentence. The content extractor transforms it into a vector of 10 real numbers in the interval [1, 7]. The symbol *nil* means *don't care*.

The 10 components are called *Noisy-Quiet, Calm-Agitated, Bright-Dark, Refreshing-Depressing, Solemn-Flippant, Leisurely-Restricted, Pretty-unattractive, Happy-Sad, Relaxed-Aroused, The mind is restored-The mind is vulnerable.* Each has associated grades (interpreted as "concepts" below). For example, the component *Happy-Sad* is characterized by the seven grades: *very happy, happy, a little happy, medium, a little sad, sad* and *very sad*. In the original IMRS, these values always correspond to the integers in the [1, 7] interval, respectively 7.0, 6.0, 5.0, 4.0, 3.0, 2.0 and 1.0.

A request to find a piece of music that gives a happy impression (happy) corresponds to the 10-dimensional vector as follows: (*nil nil nil nil nil nil nil 6.0 nil nil*) (Kumamoto, 2007), but the music pieces can be described by vectors having non-integer components.

Although we had a quite precise description of the internal representation used by IMRS. We could not find information on the rest of the system. Hence, we recreated it to emulate the functions described in the original publications. That includes the system architecture, the design and implementation of the database, the management of requests, and the programming of actually much more than the originally proposed service.

By definition, *linguistic porting* consists in making an application existing in some language L1 available in another language L2, within the same context. Evaluation of the linguistic porting of a content management application can be done at two levels.

- *Evaluation at the internal representation level*. It is an evaluation at the level of components.
- *Evaluation at the task level.* It is an *end-to-end* evaluation of the new version (in L2) of the application.

To make an end-to-end evaluation of IMRS, an IMRS Web-based simulator was developed. It makes it possible to evaluate *in context* the result of linguistic porting (Japanese → French, Arabic, English). A real database with real music pieces, characterized by 10-dimensional vectors as in IMRS, was also created.

The aim of the multilinguization was however not to develop an application strictly equivalent to IMRS, with the addition of being able to handle queries expressed in French, English and Arabic, but to develop an upward compatible, extended application. In particular, we wanted to add other dimensions corresponding to the type of music, the composer, the period of composition, the instruments used, etc. We also wanted to experiment the possibility to associate to each impression such as *happy* a fuzzy set over [1,7] expressed by a membership function (into [0,1]). More details are given below.

3.2 Our IMRS-g system

With the help of a Master student in computer science, Xiang Yin, we have programmed in

PHP/MySQL a Web service called IMRS-g, reimplementing as accurately as possible the system IMRS, and generalizing it.

Not having sufficient expertise in Japanese, we replaced Japanese by French. We also adapted the NLP part to English and Arabic, using the same strategy to handle the three languages.

We then generalized the internal representation by adding other search criteria (such as the type of music, the composer, the period of composition, and the instruments used), and using fuzzy sets.

A large set of music pieces was loaded into the database, and labelled by vectors in a collaborative way. An evaluation of the French version was then conducted as part of a realistic use, with students listening to music.

The first part of the linguistic porting has been very rapid, since it consisted only in translating into French and Arabic the NL labels expressing impressions (*Noisy/Quiet, Calm/Agitated, Sad/Happy,* etc.), by associating them the same values as in IMRS.

The content extractor processes simple utterances and extracts from them a 10-dimensional IMRS vector, and the additional information in the form (lists of) of items.

As in IMRS, a request for a music piece can be made either by typing a query in natural language, or through a graphical interface allowing to manipulate a 10-dimensional vector, and to fill fields for the other types of information.

In response, the user receives a list of links to music pieces corresponding to its selection criteria. Clicking on a link starts the playing of the corresponding music piece.

3.3 Generalization by *fuzzying* the interpretation of the NL labels

The original representation of IMRS seems too rigid to express utterances like *quite calm* or to change the current request using an utterance like *a little slower*. Even if we agree that each term corresponds to an interval of length 1 centred on its reference value, e.g. [5.5, 6.5[for *happy*, [6.5, 7.5] for *very happy*, etc., there are problems at the extremities. Therefore we studied the possibility of better modelling and better processing the requests by using fuzzy logic (Zadeh, 1965).

In order to reason from imperfect knowledge, in contrast to classical logic, fuzzy logic proposes to replace the Boolean variables used in classical logic by fuzzy variables, and the classical *crisp* sets by fuzzy sets.

Let U be a universe of elements. A *fuzzy set* A over U is defined by its membership function (f_A). An element x of U is in A with a degree of membership $f_A(x) \in [0, 1]$. If $f_A(x) \in \{0, 1\}$, A reduces to a classic set, where $x \in A$ if $f_A(x)=1$ and $x \notin A$ if $f_A(x)=0$ (f_A is then simply the characteristic function of A).

In a fuzzy set, an element x more or less belongs to the concept associated to A, or to the concept attached to A (such as *happy*). A fuzzy set is defined by all values of its membership function on its definition domain (which may be discrete or continuous).

For example, the concept *young* might be defined over the universe U of possible (integer) ages U = [0, 120] by the discrete fuzzy set A = ((10 1), (20 0.8), (30 0.6), (40 0.2), (50, 0.1), (60 0), (70 0), (80 0)). The first pair means that a 10-year old is *100% young,* and the fifth that a 50-year old is *10% young*.

Using fuzzy logic, we could say that a piece of music is *100% rapid* if its tempo is 100 (100 crotchets (quarter notes) per minute), with a bell-shaped curve for the membership function, rising from 0 to 1 and then falling, in the range [84, 112]. Then, *rapid* might be understood as *impression of rapidity*. As the impression of rapidity may differ from person to person, that curve may differ accordingly.

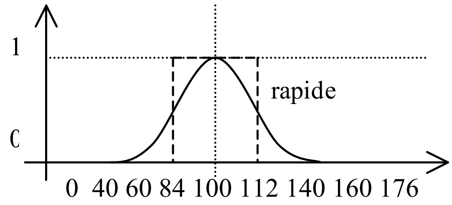


Figure 6. Representation of the rapidity impression

We propose to incorporate the possibility to move from the *perceptions* to digital measurements and to personalize the system by learning parameters of each curve of this type for each user. Such a curve can be characterized by a small number of significant points, such as the maximum, 2 points at the 10% below the maximum, 2 minima on each side, and 2 points at 10% above the global minimum.

To find the criteria for each piece of music, we have developed a website to ask a group of people to listen to music pieces and to give their opinions in terms of impressions, knowing that they will not have the same taste and the same perception. For example, for the same piece, a first listener will say that it is rapid, and a second will find it very rapid. The question here is how to merge these different views into a single impression. We propose two solutions: (1) construct a fuzzy set which is the *average* of those of annotators, possibly by giving greater weight to the annotations of *confirmed* annotators, (2) build several *perception types*, i.e. several fuzzy sets corresponding to subgroups of users with similar perceptions. We know that the Japanese persons find only slow pieces of music that Westerners find very slow.

In this work, we have taken into account previous queries of users or the history of users. For example, if a user requests a piece *a little less noisy*, or *a little more calm*, we should be able to use the information saved in his history, and calculate the new request taking into account the perceptions associated to the last piece of music listened to.

4 Specification and implementation

We specified and implemented a multimedia database for storing music pieces, as well as information representing the impressions of each piece. As said above, we added to the 10 features of IMRS other information types: singer, poet, composer, genre, album and duration, for each music piece. Moreover, to evaluate music, we stored the values of the impressions recorded by contributing users for each piece. These values were used to produce the final values stored in the database. To analyze the impressions of users, we requested further information from each user, as gender and age.

We loaded our database with a set of 354 pieces (89 Western, 265 Eastern) and all information related to each piece (artist, album, genre...). The duration of individual pieces varies between 48 seconds and 22 minutes.

The website has a login page that allows a secured access for each user. For a first connection, the user must register and fill some information from which we compute and store a profile.

If the connection is successful, a list of pieces is displayed. For each piece, a link allows listening to the music and also opens a new page providing an adapted evaluation interface appropriate to the evaluation task.

In the evaluation phase, the user can listen to the selected piece and evaluate it according to the 10 IMRS basic criteria (soft, calm, happy...). For each criterion, we offer a range of values and the user can move a cursor and put it on the value that represents its perception. Next, we propose several ways to search for music pieces.

The cost of multilinguization of the IMRS system was 3 man-months. To this cost, we add 1 man-month for the development and integration task of the content extractor for the three languages (French, Arabic, English).

5 Music retrieval modes

After registering and connecting, users listen to and evaluate music. The evaluation information is recorded directly in the database.

For each dimension, we compute the average of the obtained values. This phase is temporary pending further evaluations to draw the curves associated to each dimension and to each piece.

We defined and implemented five possibilities to search music: by user profile, by impressions, by selecting criteria, by input utterances, and by history.

5.1 Search by user profile

We propose to users music adapted to their preferences recorded in their profiles. The method follows the following steps:

- Find the user profile (ten values that represent his basic impressions) in the database.
- Compute the Euclidean distance between the two vectors formed by the 10 values of profile and the 10 values of each music piece (see example below).
- Sort pieces by distances in ascending order.
- View the nearest 10 pieces.

Here is an example:

User profile: impressions vector

Profile = (Nil 6 3 Nil 2 1 3 5 Nil Nil)

Piece impressions (existing impressions vector):

Piece1 =

(3.5 Nil 2.3 5.0 3.2 Nil 2.6 Nil 6.0 1.4)

Euclidian distance (d):

d= $\sqrt{(4-3.5)^2+(6-4)^2+(3-2.3)^2+(4-5)^2+(2-3.2)^2+(1-4)^2+(3-2.6)^2+(5-4)^2+(4-6)^2+(4-1.4)^2}$

=> d = 5.3,

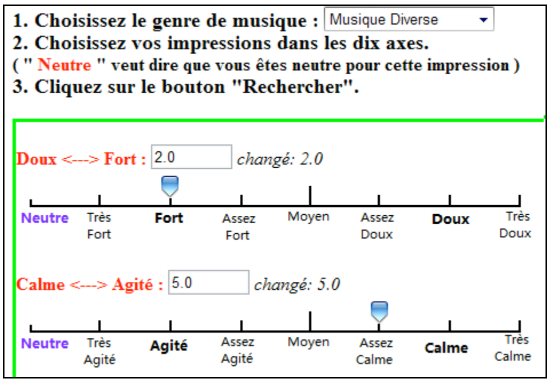
Note: if value = "Nil", we put value := 4.

5.2 Search by impressions

We ask the user to place the cursors on the values that represent his perception. We can limit ourselves to a particular type of music (Western music, Eastern music or light music). The search method has the following steps:

- Choose the kind of music (Western music, Eastern music or light music).
- Place one or more cursors on the values that represent user's perception.
- Compute the Euclidean distance between the two vectors formed by the 10 values of search and the 10 values of each piece.
- Sort pieces by distances in ascending order.
- View the nearest 10 pieces.

Here is an example: we search a noisy (*fort*) and somewhat calm (*assez calme*) piece (see Figure 7).



1. Choisissez le genre de musique : Musique Diverse

2. Choisissez vos impressions dans les dix axes.

(" Neutre " veut dire que vous êtes neutre pour cette impression)

3. Cliquez sur le bouton "Rechercher".

Doux <---> Fort : 2.0 *changé: 2.0*

Calme <---> Agité : 5.0 *changé: 5.0*

Figure 7. Example of search by impressions

The result of the previous request is a set of 10 music pieces.

5.3 Search by selection criteria

We offer four search criteria: artist, album, genre and creation date. The search methods for each of these criteria are similar.

For example, the search by artist follows the following steps:

- Search all artists (singers) existing in the database.
- Choose an artist from this list.
- Search all pieces performed by this artist and show them (by links).

5.4 Search by input utterances

The content extractor works for French, Arabic and English, and handles simple utterances related to the music domain. This program takes as input a corpus of music pieces and gives as output a file containing the corresponding vector representations.

The search method has the following steps:

- Enter an utterance in natural language representing impressions of music search.
- Call a content extractor. The result, which contains a vector representing the desired perceptions, is stored in a text file.
- Extract the vector from the text file.
- For each value of the vector (Vv), if one of the symbols (+, + +, -, --,¬) appears, then we extract the value of the last search of the concerned user (Vo: old value) from the database to compute the new value of search (Vn: new value).

Here are some examples of utterances that correspond to the precedent symbols. +: more noisy, ++: still more noisy, -: less noisy, --: still less noisy, ¬: not noisy.

We treat these symbols with the following rules:

If (V_v == '+') {$V_n = V_o + \alpha$;}
If (V_v == '++') {$V_n = V_o + 2\alpha$;}
If (V_v == '-') {$V_n = V_o - \alpha$;}
If (V_v == '--') {$V_n = V_o - 2\alpha$;}
If (V_v == '¬x') {$V_n = 7 - x$;}
If ($V_n > 7$) {$V_n = 7$;}

- Compute the Euclidean distance between the two vectors formed by the 10 desired values and the 10 values of each piece.
- Sort music by distances in ascending order.
- View the nearest 10 pieces.

5.5 Search by history

We extract from the history of each user five types of information: (a) the kind of desired pieces, (b) their creation date, (c) the artists (performers), (d) the liked albums, (e) the favourite impressions.

The search method has the following steps:

- Search the user's history in the database and check if the user has already searched with the five previous conditions.
- If the user has searched for condition (a) or (b) or (c) or (d), we extract the last value of found for each of them. 4 values are obtained.

If the user searches by impressions (condition (e)), we compute for each dimension the average that represents the history of the searches.

- Search for music using the values obtained at step 2.
  If (e) is verified, we compute the Euclidean distance between the average of impressions representing the history and impressions existing in the database.
  If (e) is not verified, we look for pieces, using only the 4 values obtained by the conditions ((a), (b), (c), (d)).

Here an example of a history of one user. For condition (a), the latest search value is “Pop”. For condition (b), there is no value, i.e. the user did not search by creation date. For condition (c), the latest search value is “1” (number of the artist). For condition (d), the latest search value is “2” (number of the album). For condition (e), there are 3 vectors in the search history:

V1=(2 5 Nil 3 Nil 2 7 1 Nil Nil)
V2=(3 Nil 4.5 2.5 Nil 3.1 6.4 Nil 5 2)
V3=(3.5 4.3 Nil 2.1 Nil Nil Nil 3 Nil Nil)

We compute the average of the history, V_m:
V_m=(2.83 4.65 4.5 2.53 Nil 2.55 6.7 2 5 2)

We search for pieces that verify the complex condition: (Kind of music = 'Pop') AND (Number of the artist ='1') AND (Album ID ='2') AND (Impressions vector is closest to Vm according to the Euclidean distance).

If the search is successful, then the result is optimal. Otherwise, we search pieces that correspond to the second condition: ((Kind of music = 'Pop') OR (Number of the artist ='1') OR (Album ID ='2')) AND (Impressions vector is closest to Vm according to the Euclidean distance).

We refined this search through other combinations formed by the conditions (a), (b), (c), (d) and (e) and differentiated by the OR and AND operators.

6 Multilingual porting

To build our content extractor, we started from a content extractor for French we had previously develop for the same domain, integrated it into IMRS-g, and extended it as explained above (more information type, and fuzzy sets). We then ported it to English and to Arabic, using the second technique of external porting (when one has access to the internal representation).

Seeing the large percentage of common code to the 3 content extractors obtained, we factorised it and obtained a unique content extractor handling input utterances in the music domain in our 3 target languages: French, English and Arabic. This technique is perhaps not generalizable, but it works for this sub-language, which is very small, and for the simple task of extracting information representable in very small textual fragments.

Here are some examples of results for Arabic, French and English:

```
Exemple_Ar 1 : أريد قطعة موسيقية جد هادئة //je veux un
morceau de musique très calme
Musique_Ar 1: musique-spec=(nil 7,0 nil nil
nil nil nil nil nil nil)
Exemple_Ar 2 : أريد قطعة موسيقية قليلة الضجيج //je veux
un morceau de musique un peu bruité
Musique_Ar 2: musique-spec=(3,0 nil nil nil
nil nil nil nil nil nil)
Exemple_Fr 1:je veux un morceau de musique
calme et très solennel
Musique_Fr 1: musique-spec=(nil 6,0 nil nil
7,0 nil nil nil nil nil)
Exemple_Fr 2:je veux un morceau de musique
assez fort et clair
Musique_Fr 2: musique-spec=(3,0 nil nil 6,0
nil nil nil nil nil nil)
Exemple_En 1:I want a calm and very solemn
music
Musique_En 1: music-spec=(nil 6,0 nil nil 7,0
nil nil nil nil nil)
Exemple_En 2:I want a little noisy and bright
music
Musique_En 2: music-spec=(3,0 nil nil 6,0 nil
nil nil nil nil nil)
```

Tableau 1: Examples of results of IMRS-g for Arabic, French and English

Conclusion

We have presented several possible methods for "porting" applications based on handling the content of spontaneous NL messages in a "native" language L1 into another language, L2. In a previous paper, we described experiments and evaluations of these methods.

We tried to do an “end-to-end” evaluation of porting IMRS by building a website that proposes to engage people in evaluation of a set of music pieces, thereby offering them to search for music in different possible modes. To that effect, we have produced a functional Web site (http://www-clips.imag.fr/geta/User/najeh.hajlaoui/Musique/). To date, the evaluation has been done only for French. More than 30 users have participated, perhaps because they were rewarded in a sense: as a kind of compensation, each user could listen to appropriate music adapted to his way of perception and taste. The use of fuzzy logic proved useful and was perhaps even necessary to give some freedom of choice of impressions to users.

Acknowledgments

We thank Yin Xiang, who dedicated his TER internship during his master in CS to the study and implemention of the IMRS-g system.

We would also like to thank our reviewers, who made many constructive remarks and suggestions, which we gladly incorporated in this new version of the paper.

References

Daoud, D. M. (2006). *It is necessary and possible to build (multilingual) NL-based restricted e-commerce systems with mixed sublanguage and contend-oriented methods.* Thèse. Université Joseph Fourier. Grenoble, France. September 23, 2006. 296 p.

Grishman, R. and R. Kittredge. (1986). *Analyzing language in restricted domains.* Hillsdale NJ. Lawrence Erlbaum Associates. 248 p.

Hajlaoui, N. (2008) *Multilinguïsation de systèmes de e-commerce traitant des énoncés spontanés en langue naturelle.* Thèse. Université Joseph Fourier. Grenoble. 25 septembre 2008. 318 p.

Harris, Z. (1968). *Mathematical structures of language.* in The Mathematical Gazette. Vol. 54(388): pp. 173-174. May, 1970.

Kittredge, R. (1978). *Textual cohesion within sublanguages: implications for automatic analysis and synthesis.* Proc. Coling-78. Bergen, Norvège. August 14-18, 1978. Vol. 1/1.

Kittredge, R. (1982). *Variation and Homogeneity of Sublanguages.* in Sublanguage - Studies of Language in Restricted Semantic Domains. Walter de Gruyter. Berlin / New York. 20 p.

Kittredge, R. (1993). *Sublanguage Analysis for Natural Language Processing.* Proc. First Symposium on Natural Language Processing. Thailand, Bangkok pp. 69-83.

Kittredge, R. and J. Lehrberger (1982a). *Sublanguage - Studies of language in restricted semantic domain.* Walter de Gruyter. Berlin / New York.

Kumamoto, T. (2004). *Design and Implementation of Natural Language Interface for Impression-based Music-Retrieval Systems.* Knowledge-Based Intelligent Information and Engineering Systems. Springer Berlin / Heidelberg. October 14, 2004. Vol. 3214/2004: pp. 139-147.

Kumamoto, T. (2007). *A Natural Language Dialogue System for Impression-based Music-Retrieval.* Proc. CICLING-07 (Computational Linguistics and Intelligent Text Processing). Mexico. February 12-24, 2007. 12 p.

Kumamoto, T. and K. Ohta (2003). *Design and Development of Natural Language Interface for an Impression-based Music Retrieval System.* in Joho Shori Gakkai Kenkyu Hokoku. Vol. 4(NL-153): pp. 97-104.

Kurohashi, S. and M. Nagao (1999) Manual for Japanese Morphological Analysis *System JUMAN.* Rap. Language Media Lab. School of Informatics, Kyoto University. Kyoto, Japan. November 1999.

Uchida, H., M. Zhu, et al. (2005-2006). *Universal Networking Language* 10 2-8399-0128-5. 218 p.

Uchida, H. and M. Zhu (1999). Enconverter Specifications, UNU/IAS UNL Center, 33 p.

Zadeh, L. A. (1965). *Fuzzy sets.* Information and Control 8: pp. 338-353.

Author Index