SAND2007-0385C

Generative Latent Semantic Analysis:

How (Computational) Linguistics Can Aid Information Retrieval

Peter A. Chew
Sandia National Laboratories
P. O. Box 5800, MS 1012
Albuquerque, NM 87185-1012, USA
pchew@sandia.gov

Ahmed Abdelali
New Mexico State University
P.O. Box 30002, Mail Stop 3CRL
Las Cruces, NM 88003-8001, USA
ahmed@crl.nmsu.edu

Abstract

Traditionally, the fields of information retrieval (IR) and computational linguistics (CL) have crossed paths only to a limited extent. One of the reasons that this is the case is that speed of processing is often critical to information retrieval, and some of what CL might have to offer IR – for example an enrichment of linguistic structure – would, if implemented at run-time, slow down the retrieval of documents to an unacceptable degree. In this paper, we set out a method for how richer linguistic analysis (including generative analysis of a type which could be output by a computationally-implemented phrase-structure grammar) can be incorporated into IR to yield demonstrable benefits in precision/recall. This is done in such a way that there is no decrease in the speed of retrieval at run-time. The way is paved for many different subfields within CL (morphology and syntax for example) to begin to contribute more to IR.

1 Introduction

In this paper, we consider three different methods for increasing the linguistic sophistication of information retrieval (IR) in ways which would perhaps be thought of more as the domain of computational linguistics (CL). Our objective is to determine which of these methods improves performance, and to what extent. The paper is organized as follows. Section 2 describes the background to our work. Section 3 describes the framework we use for IR (or more specifically cross-language information retrieval). In section 4, we present our results. Finally, we conclude on our findings in section 5.

2 Background

Although there are important differences between the goals of IR and CL, the former being driven largely by the need to provide a way for users to locate relevant documents, and the latter traditionally having a theoretical emphasis, it is perhaps surprising that CL has not contributed more to IR than it has. This state of affairs was the background to the workshop at ACL 2006, 'How Can Computational Linguistics Improve Information Retrieval?' (ACL 2006).

Perhaps the most significant reason that the reluctance of the IR community to take up what CL has to offer is that parsing – an important part of the domain of CL – tends to be slow (see form example Wu et al. 2005). In IR, where the speed with which documents are returned to the user is a critical consideration, any form of parsing beyond light stemming tends to lead to unacceptable degradation of performance.

In this paper, we focus on what parsing can offer to cross-language information retrieval (CLIR) in particular, and we show that there are ways of obtaining the benefits of parsing without suffering the loss of performance.

3 The IR framework

The framework that we use for IR is multilingual Latent Semantic Indexing (LSI) as described by Berry et al. (1994:21, and used by Landauer and Littman (1990) and Young (1994). A number of different approaches to CLIR have been proposed; generally, they rely either on the use of a parallel corpus for training, or translation of the IR query. It should be noted that the same techniques that we will propose here might also be applicable within IR frameworks other than LSI.

In the standard multilingual LSI framework, a document-by-term matrix is formed from a parallel aligned corpus. If there are p languages, m documents (each of which is translated into each of the p languages), and n distinct linguistic terms across all languages, then the document-by-term matrix is of dimensions m by n (not $m \times p$ by n). Each cell in the matrix represents a weighted frequency of a particular term t (in any language) in a particular document k. The weighting scheme we used was a standard log-entropy scheme in which the weighted frequency W is given by:

$$W = \log_2 (F + 1) \times (1 + H_t / \log_2 (N))$$

where F is the raw frequency of t in k, H_t is the standard 'p log p' measure of the entropy of the term across all documents, and N is the number of documents in the corpus. The last term in the expression above, $\log_2$ (N), is the maximum entropy that any term can have in the corpus, and therefore $(1 + H_t / \log_2 (N))$ is 1 for the most distinctive terms in the corpus, 0 for those which are least distinctive. The log-entropy weighting scheme has been shown to outperform other schemes such as tf-idf in LSI-based retrieval (see for example Dumais 1991).

The sparse document-by-term matrix is subjected to singular value decomposition (SVD), and a reduced non-sparse matrix is output. Generally, we used the output corresponding to the top 300 singular values in our experiments.

To evaluate the similarity of unseen queries or documents (those not in the training set) to one another, these documents are tokenized, the weighted frequencies are calculated in the same way as they were for the training set, and the results are multiplied by the matrices output by the SVD to project the unseen queries/documents into a 'semantic space', assigning (in our case) 300-dimensional vectors to each document. The similarity of one document to another is generally measured by calculating the cosine between the respective vectors, and that is our approach here.

For training, we used a parallel corpus built up from translations of the Bible which are freely available on the World Wide Web, as proposed by Chew et al. (2006). The multilingual aligned text we used for the work described in this paper was downloaded from the 'Unbound Bible' website (Biola University, 2005-2006), which publishes Unicode files in a tab-delimited format easily loaded into a database. The format also facilitates ensuring the alignment of text across languages by chapter and verse.

The primary languages we used for this work were as follows:

Language	Abbrevation used below
Arabic	AR
English	EN
French	FR
Koine Greek	KG
Russian	RU
Spanish	ES

Table 1. Languages

Importantly for our purposes, the Koine Greek version (the Byzantine/Majority Text 2000) is available in both unparsed and parsed formats; the unparsed format simply includes the Koine Greek text, just as the text is included for other languages. The parsed version, on the other hand, lists each Koine Greek wordform along with fairly detailed morphological information (with roots and inflectional endings separately specified); an example verse is shown in Table 2. To be clear, this morphological tokenization is not something that we compute using a technique from CL; in fact, it is the output of the manual labor of Biblical scholars (for example, the first element of each tokenization, which begins with a 'G', represents the numerical reference for the Greek root from Strong's concordance)[1]. Since we are clearly not bringing

[1] One of the advantages of using the Bible as a parallel corpus not mentioned in previous literature on the subject (Resnik et al 1999, Chew et al 2006) is that the Bible has been subject to extensive linguistic research for theological purposes. The benefit of this research to computational linguists, even where the objectives are non-theological, has in our opinion been somewhat overlooked.

Word-form	Gloss	Morphological Tokenization
ουτως	thus	G3779 ADV
γαρ	for	G1063 CONJ
ηγαπησεν	loved	G25 G5656 V-AAI-3S
ο	the	G3588 T-NSM
θεος	God	G2316 N-NSM
τον	the	G3588 T-ASM
κοσμον	world	G2889 N-ASM
ωστε	that	G5620 CONJ
τον	the	G3588 T-ASM
υιον	son	G5207 N-ASM
αυτου	of him	G846 P-GSM
τον	the	G3588 T-ASM
μονογενη	only-born	G3439 A-ASM
εδωκεν	(he) gave	G1325 G5656 V-AAI-3S
ινα	so that	G2443 CONJ
πας	all	G3956 A-NSM
ο	the	G3588 T-NSM
πιστευων	believing	G4100 G5723 V-PAP-NSM
εις	in	G1519 PREP
αυτον	him	G846 P-ASM
μη	not	G3361 PRT-N
αποληται	perish	G622 G5643 V-2AMS-3S
αλλ	but	G235 CONJ
εχη	have	G2192 G5725 V-PAS-3S
ζωην	life	G2222 N-ASF
αιωνιον	eternal	G166 A-ASF

Table 2. Example of Tokenization in Parsed Greek

our own CL techniques to bear on the information retrieval problem, we still need to clarify how we will show that CL can be useful in IR. We will return to this below.

We aligned all five parallel translations plus the parsed version by verse, and, since there are some minor differences in versification between translations, the data was cleaned to ensure proper alignment to the extent possible. After cleaning and alignment, our parallel corpus consisted of 31,226 text chunks (in most cases, the same as the original verses), aligned across all translations except for New Testament Greek, which covers only the 7,953 verses in the New Testament.

The 'unseen' documents we used as a test set were translations of the 114 suras of the Qu'ran into Arabic, English, French, Russian and Spanish (all the languages in Table 1 except Koine Greek). Since we were interested in the performance of a framework with input from CL relative to that of a framework without such input, the subject matter of the textual data used in the test set was unimportant. What mattered was that the *same* test set was used both with and without the CL input, and that the test set was sufficiently large to give statistically significant results. Although 114 documents is a relatively small number, it should be borne in mind that with all possible pairings of 5 languages (Arabic-English, Arabic-French, English-French, and so on), we actually ran 2,850 ($114 \times 5 \times 5$) queries for each set of results. Each 'query' was one of the 114 documents, in one of the 5 languages; for each of these there were five sets of retrieval results (one set from each language). To assess the aggregate performance of the framework, we used two measures: average precision at 0 (the maximum precision at any level of recall), and average precision at 1 document. Both measures were based on the ranking of the actual translation of the query in the retrieved results (which we know a priori). In our case, precision at 1 document is 1 if the translation was retrieved first, and 0 otherwise; and since we consider there to be only one relevant document, precision at 0 is the inverse of the ranking of the translation in the results. Given the number of tests we ran, we achieved statistically significant results (the levels of confidence are shown where applicable).

4 Results and Discussion

Four different tests were performed. In the first case, to provide a benchmark, we measured average precision at 0 and at 1 document without any form of parsing. This involved including all available parallel versions (including those in the five languages shown in Table 1, but excluding the parsed version of Koine Greek). The results are presented below by language pair in Tables 3 and 4 (in each case here and below, the language of the source document is shown at left, and the target language is shown at the top). The overall average precision at 1 document is 0.813, and overall average precision at 0 is 0.872.

	AR	EN	ES	FR	RU
AR	1.000	0.623	0.570	0.596	0.526
EN	0.798	1.000	0.965	0.991	0.886
ES	0.553	0.939	1.000	0.965	0.702
FR	0.605	0.982	0.939	1.000	0.868
RU	0.579	0.746	0.754	0.746	1.000

Table 3. Precision at 1 document: benchmark

	AR	EN	ES	FR	RU
AR	1.000	0.729	0.707	0.724	0.658
EN	0.853	1.000	0.981	0.996	0.932
ES	0.681	0.958	1.000	0.982	0.815
FR	0.700	0.991	0.962	1.000	0.921
RU	0.686	0.834	0.842	0.833	1.000

Table 4. Precision at 0: benchmark

Before we go on to describe the remaining three tests, some comments should be made about the standard multilingual LSI framework. First, it does not 'know' which terms come from which language, because the document-by-term matrix is formed from the *combination* of text in all languages; the vector for each verse lists the frequencies of all terms in *any* language in that verse. Of course, in our example there is a clue in that the alphabets of Arabic, Russian, Koine Greek, and the remaining languages respectively come from distinct Unicode codepages. However, this is largely irrelevant here, as the SVD is ultimately performed on a matrix in which the row and column indices are numeric, not character-based (each number stands for a distinct term or text chunk). Furthermore, it is clear that different languages may use different numbers of terms to express the same concepts, and this is reflected in the various statistics for the different translations of the Bible as shown in Table 5.

Translation	Types	Tokens
English (King James)	12,335	789,744
Spanish (Reina Valera 1909)	28,456	704,004
Russian (Synodal 1876)	47,226	560,524
Arabic (Smith Van Dyke)	55,300	440,435
French (Darby)	20,428	812,947

Table 5. Type/token statistics for translations of the Bible into 5 languages

As a result, standard LSI (and specifically the log-entropy weighting scheme) will overweight terms in some languages and underweight terms in others. This is the primary explanation for the relatively wide differences in precision shown in Tables 3 and 4; generally, the more morphologically complex a language is, the less well standard cross-language LSI (as we have implemented it) performs. We have attempted a number of modifications to the standard weighting scheme to correct for this, but so far we have not found an alternative which outperforms standard log-entropy.

This brings us to the second point about standard multilingual LSI. As already stated, some languages are more morphologically complex than others. The most common concession to linguistics made by vector-based approaches to IR is to implement some form of stemming (but not full parsing), and this has been shown to lead to a definite improvement in precision (see for example Larkey et al 2002). Generally, documents from both the training set and the test set are subjected to stemming. As already stated, the reason that most IR systems do not progress beyond a higher level of linguistic sophistication than stemming is that anything more is too slow in practice.

One final point about standard multilingual LSI is that each document (or text chunk) used in training is more or less independent of every other. This explains why it is possible to use an incomplete parallel text (such as the Greek one we use, which covers only the New Testament) without fear of significantly impairing the overall precision; we have confirmed that this is the case in other tests not discussed further here. There is, however, a disadvantage to this independence, which is that the relationships between related (for example, adjacent) text chunks are not used in any way by LSI.

In short, there is much linguistic richness in text which perhaps most approaches to IR fail to exploit. This is true in particular for parallel aligned text and CLIR. Furthermore, we have listed just three such unused structural relationships in the text we use for training, without even touching on the many aspects of semantic or syntactic structure which could also be formalized and used. The challenge, then, is to exploit this richness within an IR framework without compromising the framework's performance. As we shall show, CL can measure up to this challenge.

For the second test, to determine the effect of morphological parsing of one language upon precision, we added parsed Greek to the training set. Each morphological element was treated as a separate 'term' for IR purposes. To take an example from Table 2, every time 'αποληται' occurs in a verse, the terms 'G622', 'G5643' 'V', '2AMS', and '3S' will also occur. Each of these terms represents a separate morphological property of 'αποληται'. In effect, this allows LSI to make the association between different wordforms of a common lexeme, or even (for example) different cases where the third person singular occurs.

One might wonder, with the additional information which must now be processed, how it can be claimed that IR is still as efficient. The reason is that, since our test languages do not include Koine Greek, the size of the matrix output by SVD and used subsequently is exactly the same as it was for the first test (to be precise, 300 dimensions × 160,396 terms for Arabic, English, French, Russian and Spanish). The effect of the parsed Koine Greek is simply to make changes to the values in the cells for the 160,396 terms, as Koine Greek terms can be discarded after computing the SVD if they are not needed for testing. It is true that there is some additional processing on the front-end, but this processing has to take place only once to produce the SVD output, and has no effect on run-time. Since the SVD output is no larger than in the previous test, the requirements for run-time processing (the processing which is used ultimately to compute the similarities of unseen documents and queries) are unchanged from before. Note in particular that the parsing needs to take place only for the training data; there is no requirement for parsing of the unseen test documents.

The addition of parsed Greek raised overall precision at 1 document from 0.813 to 0.820, and overall precision at 0 from 0.871 to 0.874 - relatively modest increases, but nonetheless significant ($p \approx 0.034$). The greatest boost in precision was for Russian (Arabic-Russian +0.044, English-Russian +0.053, Russian-English +0.026; overall, averaging +0.016)[2]. Again, the increases for Russian are significant given the size of the test set ($p \approx 0.003$). The overall results by language pair are shown in Tables 6 and 7.

	AR	EN	ES	FR	RU
AR	1.000	0.623	0.509	0.614	0.570
EN	0.781	1.000	0.965	0.991	0.939
ES	0.588	0.930	1.000	0.965	0.702
FR	0.623	0.991	0.947	1.000	0.851
RU	0.596	0.772	0.772	0.763	1.000

Table 6. Precision at 1 document with morphology

	AR	EN	ES	FR	RU
AR	1.000	0.719	0.679	0.722	0.688
EN	0.839	1.000	0.981	0.996	0.960
ES	0.705	0.954	1.000	0.982	0.809
FR	0.706	0.996	0.966	1.000	0.914
RU	0.700	0.848	0.851	0.846	1.000

Table 7. Precision at 0 with morphology

We think that Greek parsing helped Russian in particular because of the similarities between Russian and Koine Greek morphology, for example in the nominal case system. Given these results, there appears to be prima facie evidence that precision would be raised still further by the addition to the training data of, for example, parsed Russian and Arabic – and there are many morphological parsers which could perform the necessary pre-processing. However, testing this was beyond the scope of this experiment.

With this in perspective, we can now explain why, although we are using pre-parsed text, we can make the claim that CL can benefit IR. Granted, we did not have to implement a CL grammar to achieve improvements in precision, but it is easy to see that CL systems can produce output such as the Greek morphological tokenization shown in Table 2 above. Furthermore, there are CL systems which produce much richer analyses than the ones we use here. For example, a single phrase-structure grammar (PSG) can parse text, deducing morphological, syntactic, and even phonological structure. In PSG, the output is typically a parse tree in which each node represents some linguistic entity, either subword (such as the morphological elements above) or super-word (such as a phrase or sentence). Analogously to the morphological elements from Koine Greek, these phrase-structure elements can be treated as terms in the document-by-term matrix.

For the third test, we wanted to investigate the effect of including 'discourse' structure in training.

[2] For some language pairs, there were small decreases in precision, but these were more than offset by increases for other pairs.

The Bible text we used consists of 66 books; in some cases there is debate among theologians about authorship, and it is perhaps a simplistic assumption that each book has a single author, but there is wider theological agreement that each book has definite distinctive themes. Thus, it does not seem unreasonable to assume that all verses within a book are part of a common discourse structure. Similarly, the boundaries between chapters tend to fall in places corresponding to thematic transitions. For IR purposes, therefore, we need to associate verses by use of a new set of terms, each of which, linguistically speaking, represents a distinct 'element of discourse', or – to put it in phrase-structure terms – a high-level node in the parse tree. Since standard LSI treats each text chunk separately, any thematic associations between text chunks are lost, but the addition of these terms allows LSI to make associations that it would otherwise be unable to make.

For the third test, therefore, we included two extra terms per verse, one to denote the book of the Bible that the verse came from, and another to denote the chapter. The results of this test are shown in Tables 8 and 9.

	AR	EN	ES	FR	RU
AR	1.000	0.623	0.509	0.614	0.570
EN	0.781	1.000	0.965	0.991	0.939
ES	0.588	0.930	1.000	0.965	0.702
FR	0.623	0.991	0.947	1.000	0.851
RU	0.596	0.772	0.772	0.763	1.000

Table 8. Precision at 1 document with morphology and discourse structure

	AR	EN	ES	FR	RU
AR	1.000	0.719	0.679	0.722	0.688
EN	0.839	1.000	0.981	0.996	0.960
ES	0.705	0.954	1.000	0.982	0.809
FR	0.706	0.996	0.966	1.000	0.914
RU	0.700	0.848	0.851	0.846	1.000

Table 9. Precision at 0 with morphology and discourse structure

For sure, there are likely to be more sophisticated approaches to discourse analysis than the simple one we have employed, and doubtless the results of using these approaches would produce superior results. The point is, however, that even the simple approach that we took improves IR precision; how much more then would a more sophisticated approach contribute?

The final test involved exploiting the multilingual structure of the parallel text, taking into account the fact that the contribution of terms to each verse is not equal for all languages. Just as in the previous test we included a small number of additional terms to represent linguistic metadata – specifically, that relating to the discourse structure – we now include a few more additional terms, one to represent each distinct language used in training. As already mentioned, standard multilingual LSI cannot know how many terms each language contributes to each verse. For a linguist, however, this is straightforward to determine. For example, the language of a term could be output by a parsing algorithm, and there are also methods in IR for deducing the language of text – an example is the SILC system developed at RALI, University of Montreal[3]. Parsing would be essential if the language of the training text were unknown, but with our training data, the parsing stage can be bypassed altogether since we know the language of each text chunk in the parallel aligned corpus.

A slight variation was made to the standard log-entropy weighting scheme just for the new terms added in this test. For these terms, the weighted frequency was given by:

$$W = F \times (1 + H_t / \log_2 (N))$$

In other words, $\log_2(F)$ is replaced by F. The rationale for $\log_2(F)$ in the standard weighting scheme is the well-known principle that linguistic tokens generally follow a Zipf distribution, where ranked frequency is inversely proportional to log(frequency). The effect of using log(frequency) is to dampen down highly frequent terms. However, this is inappropriate in the case of the new 'language' terms. For example, if a particular text chunk contains 10 English tokens and 5 Arabic tokens, it should be the case that each English token holds 5/10 (not $\log_2 5$ / $\log_2 10$) as much 'information' (in the information theoretic sense) as each Arabic token, on average. (Strictly speaking, incidentally, the same modification should be made for the 'discourse' terms in the previous test, although in practice it is immaterial whether standard or modified weighting is used in that case,

[3] http://rali.iro.umontreal.ca/

since there is only ever one of each type of term per text chunk, and, if F=1, $\log_2(F+1) = F$. This will be true for any linguistic structure which spans across text chunks.)

The results of the final test are shown in Tables 10 and 11.

	AR	EN	ES	FR	RU
AR	1.000	0.623	0.509	0.614	0.570
EN	0.781	1.000	0.965	0.991	0.939
ES	0.588	0.930	1.000	0.965	0.702
FR	0.623	0.991	0.947	1.000	0.851
RU	0.596	0.772	0.772	0.763	1.000

Table 10. Precision at 1 document with morphology, discourse structure, and cues for language

	AR	EN	ES	FR	RU
AR	1.000	0.719	0.679	0.722	0.688
EN	0.839	1.000	0.981	0.996	0.960
ES	0.705	0.954	1.000	0.982	0.809
FR	0.706	0.996	0.966	1.000	0.914
RU	0.700	0.848	0.851	0.846	1.000

Table 11. Precision at 0 with morphology, discourse structure, and cues for language

5 Conclusion

In this paper, we have shown that, with each addition of three different types of linguistic metadata to the training data used for multilingual LSI, there is a small, but definite and significant, increase in two separate measures of precision. Linguistic metadata is essentially the linguist's stock-in-trade; one of the useful functions of generative grammar is to assign hierarchical structure (or metadata) to unparsed linguistic strings. Clearly, computational linguistics has a special role to play here, as it offers many ways to automate the assignment of such hierarchical structure –something which is beyond the reach of most standard IR systems.

Although our work did not include the implementation of a computational grammar, it is quite possible to see how data analogous to our preparsed training data could have been output by such a grammar. Moreover, since all three cases where we added structural information resulted in improved precision in the IR tests, there is certainly prima facie evidence that 'more of the same' would result in further increases in precision. The structural information we added was morphological or related to discourse, but in reality any type of structure that CL deals with (particularly those that have some connection to the text's meaning, so possibly not phonetic or phonological structure) could be used in such a system – syntax, semantics, and so on. We also added sub-word (morphological) structure only for one language, Koine Greek, whereas similar structure could have been added for all training languages, given sufficient time.

It should also be emphasized that there is no reason that linguistic metadata for different languages, or different types of structure, must be produced within a single framework (for example, a single phrase-structure grammar). The Koine Greek morphological structure follows the Strong's system, but if Russian morphological structure were added, it would not necessarily have to follow a system based on the foundational concepts of morphology. After all, the five languages we used for testing differ from one another considerably in morphological structure, and even without any parsing LSI is able to find cross-language correlations. (Essentially, each translation 'parses' every other translation.) The only requirement is that the additional structural information should clarify, not obfuscate, the semantic relationships between words and text chunks.

To conclude, there is good reason to suppose that computational linguistics has an important role to play in information retrieval, and that there are many ways this role could be played. The evidence we have presented in this paper shows that with each successive addition to the training data of linguistic metadata – of the type that computational linguistics is well-equipped to produce – there is a small but significant increase in precision when the training data is used for cross-language information retrieval. Since the role for CL can be confined to the preparation of training data but still result in a benefit to the test results, we have shown that it is possible to achieve improvement in IR precision without in any way compromising IR performance at run-time, when speed matters most.

Acknowledgement

Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy's Na-

tional Nuclear Security Administration under contract DE-AC04-94AL85000.

References

Association for Computational Linguistics. 2006. *How Can Computational Linguistics Improve Information Retrieval?: Proceedings of the Workshop*. Sydney: Association for Computational Linguistics.

Ricardo Baeza-Yates and Berthier Ribeiro-Neto. 1999. *Modern Information Retrieval*. New York: ACM Press.

Michael Berry, Theresa Do, Gavin O'Brien, Vijay Krishna, and Sowmimi Varadhan. 1996. SVDPACKC (Version 1.0) User's Guide. Knoxville, TN: University of Tennessee.

Biola University. 2005-2006. *The Unbound Bible*. Accessed at http://www.unboundbible.com/ on Jan. 5, 2007.

Peter Chew, Stephen Verzi, Travis Bauer and Jonathan McClain. 2006. Evaluation of the Bible as a Resource for Cross-Language Information Retrieval. In *Proceedings of the Workshop on Multilingual Language Resources and Interoperability*, 68-74. Sydney: Association for Computational Linguistics.

Susan Dumais. 1991. Improving the Retrieval of Information from External Sources. *Behavior Research Methods, Instruments, and Computers* 23(2):229-236.

Julio Gonzalo. 2001. Language Resources in Cross-Language Text Retrieval: a CLEF Perspective. In Carol Peters (ed.). *Cross-Language Information Retrieval and Evaluation: Workshop of the Cross-Language Evaluation Forum, CLEF 2000*: 36-47. Berlin: Springer-Verlag.

Leah Larkey, Lisa Ballesteros, and Margaret Connell. 2002. Improving stemming for Arabic information retrieval: light stemming and co-occurrence analysis. In Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 275-282. Tampere: ACM.

Dragos Munteanu and Daniel Marcu. 2006. Improving Machine Translation Performance by Exploiting Non-Parallel Corpora. *Computational Linguistics* 31(4):477-504.

Jian-Yun Nie and Fuman Jin. 2002. A Multilingual Approach to Multilingual Information Retrieval. *Proceedings of the Cross-Language Evaluation Forum*, 101-110. Berlin: Springer-Verlag.

Jian-Yun Nie, Michel Simard, Pierre Isabelle, and Richard Durand. 1999. Cross-Language Retrieval based on Parallel Texts and Automatic Mining of Parallel Texts from the Web. *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 74-81, August 15-19, 1999, Berkeley, CA.

Carol Peters (ed.). 2001. *Cross-Language Information Retrieval and Evaluation: Workshop of the Cross-Language Evaluation Forum, CLEF 2000*. Berlin: Springer-Verlag.

Philip Resnik, Mari Broman Olsen, and Mona Diab. 1999. The Bible as a Parallel Corpus: Annotating the "Book of 2000 Tongues". *Computers and the Humanities*, 33: 129-153.