Using Wikipedia and Wiktionary in Domain-Specific Information Retrieval

Christof Müller and Iryna Gurevych

Ubiquitous Knowledge Processing Lab, Computer Science Department,
Technische Universität Darmstadt,
Hochschulstr. 10, D-64289 Darmstadt, Germany
{mueller,gurevych}@tk.informatik.tu-darmstadt.de
http://www.ukp.tu-darmstadt.de

Abstract. The main objective of our experiments in the domain-specific track at CLEF 2008 is utilizing semantic knowledge from collaborative knowledge bases such as Wikipedia and Wiktionary to improve the effectiveness of information retrieval. While Wikipedia has already been used in IR, the application of Wiktionary in this task is new. We evaluate two retrieval models, i.e. SR-Text and SR-Word, based on semantic relatedness by comparing their performance to a statistical model as implemented by Lucene. We refer to Wikipedia article titles and Wiktionary word entries as concepts and map query and document terms to concept vectors which are then used to compute the document relevance. In the bilingual task, we translate the English topics into the document language, i.e. German, by using machine translation. For SR-Text, we alternatively perform the translation process by using cross-language links in Wikipedia, whereby the terms are directly mapped to concept vectors in the target language. The evaluation shows that the latter approach especially improves the retrieval performance in cases where the machine translation system incorrectly translates query terms.

Key words: Information Retrieval, Semantic Relatedness, Collaborative Knowledge Bases, Cross-Language Information Retrieval

1 Introduction

Statistical models are most frequently used in domain-specific information retrieval (**IR**). One of the disadvantages of these models is their lack of flexibility concerning synonymy, i.e. expressing a concept with different terms. There exist several approaches of tackling the problem of synonymy divided into local and global methods.

Local methods like relevance and pseudo-relevance feedback try to refine the representation of the user's information need by using either manual or automatic feedback about already returned documents. However, these methods require that the relevant documents show a significant term overlap, and that the term overlap between relevant and irrelevant documents is small. Also they are not able to close the gap between the vocabulary used in queries and in documents, i.e. query terms which do not explicitly occur in the document collection cannot be expanded with related terms.

Global methods expand the query with related terms using either automatically built thesauri based on the document collection or external linguistic knowledge bases like WordNet [1]. Using thesauri which are based on the document collection also suffers from the inability to close the vocabulary gap, if query terms do not occur in the document collection. The use of linguistic knowledge bases for query expansion has shown inconclusive results so far. Voorhees [2] could improve retrieval performance only in some cases even for manually selected expansion terms, while Mandala *et al.* [3] improved the performance on several test collections by combining a linguistic knowledge base with different types of thesauri built from the underlying text collections. The general problem of query expansion is that in fact it is able to improve recall in certain situations, but at the same time precision degrades as also irrelevant terms are added to the query.

Another knowledge-based approach to tackle the problem of synonymy is to use retrieval models which are based on semantic relatedness (**SR**) between query and document terms computed by using linguistic knowledge bases. Although first results of employing SR in IR were inconclusive [4], there have also been several promising results, e.g., [5, 6]. One of the main problems with using linguistic knowledge bases for semantically enhanced IR is the low coverage, especially of domain-specific vocabulary.

A new form of resources, so called collaborative knowledge bases [7] have the potential to overcome these limitations. Enabled by Web 2.0 technologies which simplify the editing and annotation process of web content, collaborative knowledge bases are constructed by volunteers on the web and have reached a size which makes them promising for improving IR performance. The most widely used and probably largest collaborative knowledge base is Wikipedia[1] which contains encyclopedic knowledge in a broad range of domains.

For our experiments in the domain-specific track at CLEF 2008 [8], we employ Wikipedia and for the first time Wiktionary[2] as knowledge bases for SR-based IR models. We compare their performance to a statistical model and also combine all three models by adding their respective relevance scores for each document. We perform the experiments for the languages English, German, and Russian. For bilingual IR experiments using English topics on a German document collection, we use (i) machine translation methods for statistical and semantic IR models, and (ii) cross-language links in Wikipedia for one of the semantic IR models.

2 Information Retrieval Models

Besides applying standard preprocessing steps like tokenization and stopword removal, we use the TreeTagger [9] for lemmatization. For the German test data,

[1] `http://www.wikipedia.org`

[2] `http://www.wiktionary.org`

we also split compounds into their constituents [10], and we use both constituents and compounds in the retrieval process. As baseline IR model we use Lucene[3] which is based on the vector space model. We also use Lucene for combining it with the semantic models.

2.1 Semantic Models

In our experiments, we adapt a method proposed by Gabrilovich and Markovitch [11] where article titles in Wikipedia are referred to as concepts and the article texts as textual representation of these concepts. The concept vector of a term consists of its tf value in the respective Wikipedia articles. In order to map a document or a query to its concept vector, we first build the concept vectors for all its terms. We then sum up the concept vectors after normalizing each vector and scaling it with the respective term's tf and idf values. Given the concept vector of a query and a document, we use the cosine of the angle between the two vectors as relevance score. We refer to this model as **SR-Text**.

Additionally, we employ a retrieval model proposed in [12], which we refer to as **SR-Word**. We extended the model by also taking into account the idf value of document terms and the tf value of query and document terms. This model is represented by the following equation:

$$r_{SR}(d,q) = \frac{\sum_{i=1}^{n_d} \sum_{j=1}^{n_q} tf(t_{d,i},d) \cdot idf(t_{d,i}) \cdot tf(t_{q,j},q) \cdot idf(t_{q,j}) \cdot s(t_{d,i},t_{q,j})}{(1+n_{nsm}) \cdot (1+n_{nr})} \tag{1}$$

where n_d is the number of unique terms in the document, n_q the number of unique terms in the query, $t_{d,i}$ the i-th unique document term, $t_{q,j}$ the j-th unique query term, $s(t_{d,i},t_{q,j})$ the SR score for the respective document and query term (using the cosine of the respective terms' concept vectors as score analog to SR-Text), n_{nsm} the number of unique query terms not literally found in the document, and n_{nr} the number of unique query terms which do not contribute a SR score above a predefined threshold. For SR-Text and SR-Word, we compute tf and idf as follows:

$$tf(t) = 1 + \log f(t) \tag{2}$$

where $f(t)$ is the frequency of term t in the corresponding document or query, and

$$idf(t) = \log \frac{n_{docs}}{df(t)} \tag{3}$$

where n_{docs} is the number of documents in the collection and $df(t)$ is the number of documents in the collection containing term t.

Besides Wikipedia we use Wiktionary as a knowledge base for the IR models. Thereby, we refer to each word entry in Wiktionary as a distinct concept, and use the entry's information as the textual representation of a concept analogous to

[3] `http://lucene.apache.org`

the text of Wikipedia articles (for details see [13]). In order to improve retrieval effectiveness, we combine the concept space of Wikipedia and Wiktionary, so that the concept vector of one term consists of concepts from both knowledge bases. For Wikipedia we remove concepts where the respective Wikipedia articles have less than 100 words or fewer than 5 in- or outlinks. For both Wikipedia and Wiktionary, we remove concepts from a term's concept vector if their normalized values are below the predefined threshold of *0.01*. The pruning methods are applied with the goal of noise reduction and better performance. For accessing the collaborative knowledge bases we use freely available Java-based APIs described in [7].

2.2 Combination of Models

As the statistical and semantic models use different types of information represented in queries, documents and possibly external knowledge, we hypothesize that a combination of the models might increase the retrieval effectiveness. We therefore combine their relevance scores computed separately into one relevance score for each document per query. For computing the combined relevance score, we use the *CombSUM* method which was introduced by Fox and Shaw [14] where the combined relevance score is set to the sum of the individual relevance scores. Before combining the scores, they are normalized using the formula:

$$r_{norm} = \frac{r_{orig} - r_{min}}{r_{max} - r_{min}} \tag{4}$$

where r_{orig} is the original relevance score, r_{min} is the minimal and r_{max} is the maximal occurring score for the query.

3 Evaluation

We experiment with several query types by using different combinations of the topic fields. In our training runs using topics from the past CLEF workshops, we found that the retrieval effectiveness improved when query terms are weighted depending on the field in which they occur. We therefore use the following weights for query terms in all experiments: 1 for *title* (**T**), 0.8 for *description* (**D**), and 0.6 for *narrative* (**N**).

We set the threshold for SR values in SR-Word to the following values as they showed the best performance in the training runs: 0.25 for English, 0.11 for German, and 0.23 for Russian.

Besides the officially submitted runs, we performed several experiments where the concept vectors used in SR-Text were normalized again after removing some concepts that had values below the predefined threshold of 0.01. We found that the performance increased slightly for most experiments. We therefore report the results of these new experiments together with some other additional runs.

3.1 Monolingual Retrieval

Table 1 shows the mean average precision **(MAP)** of each model and the combination of all models over query length and language for the monolingual experiments. For English and German, we used the combination of Wikipedia and Wiktionary as knowledge base, for Russian we used only Wikipedia as the API for Wiktionary does not allow to parse the Russian Wiktionary edition.

For English, Lucene outperforms the semantic models for all query types. For German this is only the case for the longest query type TDN. Using query types T and TD, the SR-Word model outperforms Lucene. Except for the query type T where SR-Text performs best, the semantic models are outperformed on the Russian data set by Lucene. However, when Lucene is combined with the semantic models by using the CombSUM method, MAP increases for all languages and query types and outperforms the separate models. Compared to Lucene, the highest and statistically significant[4] increase of MAP is 9% for English and 15% for German. For Russian we receive the best result when Lucene using query type TDN is combined with SR-Word using query type T. This results in a MAP of 0.1491 which is an increase of 16% as compared to Lucene. However, the difference is not statistically significant.

The performance of Lucene almost consistently increases for longer query types on all three languages. For SR-Text we also observe a trend to perform better for longer queries except for Russian. For SR-Word the trend is opposite for German and Russian.

For English and German, we also performed experiments using either Wikipedia or Wiktionary separately as knowledge base. The results show that for German the combination of Wikipedia and Wiktionary slightly improves the performance in most cases. For English using only Wikipedia often performs better than using the combination of both knowledge bases. Using Wiktionary separately always performed worse than using Wikipedia or the combination of both.

Table 1. The MAP values of the monolingual runs. The highest value of the separate models is in bold for each query type.

	English			*German*			*Russian*		
	T	TD	TDN	T	TD	TDN	T	TD	TDN
Lucene	**0.2514**	**0.2983**	**0.2987**	0.3405	0.3318	**0.3536**	0.1194	**0.1254**	**0.1286**
SR-Text	0.2020	0.2220	0.2521	0.2761	0.3204	0.3302	**0.1277**	0.1096	0.0745
SR-Word	0.2351	0.2595	0.2526	**0.3605**	**0.3548**	0.3248	0.1211	0.1058	0.0930
Combination	0.2735	0.3104	0.3211	0.3719	0.3820	0.3922	0.1387	0.1383	0.1330

[4] We used a paired t-test to determine the statistical significance.

3.2 Bilingual Retrieval

In the bilingual retrieval, we use English topics with the German document collection. The English topics are translated into German using machine translation[5] (**MT**). For the SR-Text model, we additionally explore a different method using the cross-language links (**CLL**) between language specific editions of Wikipedia. A cross-language link points from an article in one language to the same article in a different language, e.g. an English article might point to its German counterpart. By using these links, we map a concept vector whose concepts are represented by articles in the English Wikipedia into a concept vector whose concepts are represented by articles in the German Wikipedia. Thus, by transforming the concept vector of an English query using cross-language links, the similarity between the English query and a German document is computed by the SR-Text model without actually translating the query.[6] As Wiktionary also has cross-language links and furthermore many of the word entries contain translations of the term into other languages, it is possible to apply the CLL method to both Wikipedia and Wiktionary. However, we only report the results for using CLLs in Wikipedia.

The results of the bilingual runs are shown in Table 2. Generally, the MAP values in our bilingual runs are much lower compared to the monolingual German runs as both methods, MT and CLL, add noise to the retrieval process. For the query types T and TD, SR-Word is the best performing model. For the query type TDN, Lucene performs slightly better than SR-Word. At first sight, SR-Text using MT seems to yield better results than SR-Text using the CLL method. When combined with the Lucene model, SR-Text-CLL outperforms SR-Text-MT. When we use the respective best performing query type for each model, the combination of Lucene with query type TDN, SR-Text-CLL with query type TD and SR-Word with query type T results in a MAP of 0.2350 which is the best performance of our bilingual runs. Compared to using Lucene alone, this is a significant increase of 35%. This run is not shown in Table 2.

Analyzing the results of individual queries, we found that the CLL method is especially beneficial in cases of substantial translation errors for the query terms. In topic no. 209 where the English title field contains the terms *Doping and sports* the correct German translation of *Doping* would be the same term *Doping*. Instead, it is incorrectly translated by the machine translation system to *Lackieren* which has the meaning of *painting* or *varnishing*. As the Lucene model relies on the translation with the MT system, the combination with SR-Text using the CLL method especially improves the retrieval in these cases. The lower performance of SR-Text-CLL compared to SR-Text-MT when not combined with Lucene might result from missing cross-language links between articles in the German and English Wikipedia. Not even half of the articles in the German Wikipedia link to the respective articles in the English Wikipedia.

[5] `http://babelfish.yahoo.com/` which is based on the Systran Translator.

[6] As we do not actually translate the query terms, we have no information about the document frequency of a query term to compute its idf value. Therefore, we use the term's document frequency in Wikipedia for computing its idf value.

Table 2. The MAP values of the bilingual runs. The highest value of the separate models and the combinations is in bold for each query type.

	T	TD	TDN
Lucene	0.1490	0.1638	**0.1746**
SR-Text-MT	0.1173	0.1519	0.1547
SR-Text-CLL	0.1193	0.1288	0.1225
SR-Word	**0.1806**	**0.1760**	0.1688
Lucene + SR-Text-MT	0.1476	0.1783	0.1925
Lucene + SR-Text-CLL	0.1963	**0.2139**	**0.2205**
Lucene + SR-Text-MT + SR-Word	0.1687	0.1891	0.1976
Lucene + SR-Text-CLL + SR-Word	**0.2003**	0.2117	0.2162
Lucene + SR-Text-MT + SR-Text-CLL + SR-Word	0.1944	0.2089	0.2128

4 Conclusions

In our experiments, we have explored the integration of semantic knowledge from collaborative knowledge bases into IR. For the first time, we have employed Wiktionary in combination with Wikipedia for this task. We have evaluated two IR models (SR-Text and SR-Word) based on semantic relatedness by comparing their performance to a statistical model as implemented by Lucene. In these semantic models, the articles in Wikipedia and the word entries in Wiktionary are employed as textual representations of concepts. The SR-Text model computes the similarity of a query and document by summing up the concept vectors of the query and document terms respectively and then computing the cosine of the angle between the query's and the document's concept vector. The SR-Word model combines individual similarities of each query and document term pair that are above a predefined threshold and then applies a set of heuristics to compute the final relevance score.

In the monolingual task, the combination of Lucene and the semantic models increases the MAP by 9% for English, 15% for German, and 16% for Russian as compared to Lucene. In the bilingual task, we translated the English topics into the document language, i.e. German, by using machine translation. For SR-Text, we additionally explored a different method using the cross-language links between different language editions of Wikipedia. This approach especially improved the retrieval performance in cases where the machine translation system incorrectly translated terms. When Lucene was combined with SR-Text-CLL and SR-Word, the MAP increased by 35%. In our future work, we will additionally use the cross-language links in Wiktionary to further improve the IR effectiveness. We also plan to integrate the cross-language links into the SR-Word model.

5 Acknowledgement

This work was supported by the Volkswagen Foundation as part of the Lichtenberg-Professorship Program under grant No. I/82806 and by the German Research Foundation under grant No. GU 798/1-3.

References

1. Fellbaum, C., ed.: WordNet: An Electronic Lexical Database. MIT Press, Cambridge, MA (1998)
2. Voorhees, E.M.: Query expansion using lexical-semantic relations. In: SIGIR '94: Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval, New York, NY, USA, Springer-Verlag New York, Inc. (1994) 61–69
3. Mandala, R., Tokunaga, T., Tanaka, H.: The Use of WordNet in Information Retrieval. In Harabagiu, S., ed.: Proceedings of the COLING-ACL workshop on Usage of WordNet in Natural Language Processing. Association for Computational Linguistics, Somerset, New Jersey (1998) 31–37
4. Smeaton, A.: Using NLP or NLP Resources for Information Retrieval Tasks. In Strzalkowski, T., ed.: Natural Language Information Retrieval. Kluwer Academic Publishers (1999) 99–111
5. Lytinen, S., Tomuro, N., Repede, T.: The use of WordNet sense tagging in FAQFinder. In: Proceedings of the AAAI-2000 workshop on AI and Web Search, Austin, TX (2000)
6. Müller, C., Gurevych, I., Mühlhäuser, M.: Integrating Semantic Knowledge into Text Similarity and Information Retrieval. In: Proceedings of the First IEEE International Conference on Semantic Computing (ICSC), Irvine, CA, USA (2007) 257–264
7. Zesch, T., Müller, C., Gurevych, I.: Extracting Lexical Semantic Knowledge from Wikipedia and Wiktionary. In: Proceedings of the Conference on Language Resources and Evaluation (LREC). (2008)
8. Petras, V., Baerisch, S.: The Domain-Specific Track at CLEF 2008. In Peters, C., Deselaers, T., Ferro, N., Gonzalo, J., J.F.Jones, G., Kurimo, M., Mandl, T., Peas, A., Petras, V., eds.: Evaluating Systems for Multilingual and Multimodal Information Access 9th Workshop of the Cross-Language Evaluation Forum, CLEF 2008. LNCS, Heidelberg, Springer (2009)
9. Schmid, H.: Probabilistic part-of-speech tagging using decision trees. In: Proceedings of Conference on New Methods in Language Processing. (1994)
10. Langer, S.: Zur Morphologie und Semantik von Nominalkomposita. In: Tagungsband der Konferenz zur Verarbeitung natürlicher Sprache, KONVENS. (1998) 83–97
11. Gabrilovich, E., Markovitch, S.: Computing Semantic Relatedness using Wikipedia-based Explicit Semantic Analysis. In: Proceedings of The Twentieth International Joint Conference for Artificial Intelligence, Hyderabad, India (2007) 1606–1611
12. Müller, C., Gurevych, I.: Exploring the Potential of Semantic Relatedness in Information Retrieval. In Schaaf, M., Althoff, K.D., eds.: LWA 2006 Lernen - Wissensentdeckung - Adaptivität, 9.-11.10.2006 in Hildesheim. Hildesheimer Informatikberichte, Hildesheim, Germany, GI-Fachgruppe Information Retrieval, Universität Hildesheim (2006) 126–131
13. Zesch, T., Müller, C., Gurevych, I.: Using Wiktionary for Computing Semantic Relatedness. In: Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence, AAAI 2008, Chicago, Illinois, USA (2008) (861–867)
14. Fox, E., Shaw, J.: Combination of multiple searches. In: Proceedings of the 2nd Text REtrieval Conference (TREC-2). (1994) 243–252