Combining Probabilistic and Translation-Based Models for Information Retrieval Based on Word Sense Annotations

Elisabeth Wolf[1], Delphine Bernhard[2,⋆], and Iryna Gurevych[1]

[1] Ubiquitous Knowledge Processing (UKP) Lab, Computer Science Department,
Technische Universität Darmstadt, Germany
http://www.ukp.tu-darmstadt.de
[2] ILES group, LIMSI-CNRS,
Orsay, France
delphine.bernhard@limsi.fr

Abstract. The objective of our experiments in the monolingual robust word sense disambiguation (WSD) track at CLEF 2009 is twofold. On the one hand, we intend to increase the precision of WSD by a heuristic-based combination of the annotations of the two WSD systems. For this, we provide an extrinsic evaluation on different levels of word sense accuracy. On the other hand, we aim at combining an often used probabilistic model, namely the Divergence From Randomness BM25 model, with a monolingual translation-based model. Our best performing system with and without utilizing word senses ranked 1st overall in the monolingual task. However, we could not observe any improvement by applying the sense annotations compared to the retrieval settings based on tokens or lemmas only.

1 Introduction

The CLEF robust WSD track 2009 follows the same design as in 2008, when runs by different systems were submitted varying in the pre-processing steps, indexing procedures, ranking functions, the application of query expansion methods, and the integration of word senses. In 2008, the best performance could be achieved by a combination of different probabilistic models (**PM**s) [7], namely the BM25 model, a Divergence From Randomness version of the BM25 model, and a statistical language model introduced by Hiemstra. In our experiments, we combine an often used PM with a monolingual translation-based model (**TM**), which was trained on definitions and glosses provided by different lexical semantic resources, namely WordNet, Wiktionary, Wikipedia, and Simple Wikipedia. This TM was successfully used for the task of answer finding by Bernhard and Gurevych [4]. Further, as all participants in 2008 took only one of the two systems for WSD into account when selecting the word sense annotations, we intend to increase the precision of WSD by an heuristic-based combination of the annotations of

[⋆] This work was done while the author was at the UKP Lab, Darmstadt, Germany.

C. Peters et al. (Eds.): CLEF 2009 Workshop, Part I, LNCS 6241, pp. 120–127, 2010.

the two WSD systems. We provide an extrinsic evaluation on different levels of word sense accuracy. The task description and detailed information about the data collection can be found in the track overview paper [1].

2 Indexing and Retrieval Models

2.1 Indexing

We used Terrier (TERabyte RetrIEveR) [9], version 2.1 for indexing the documents. Each of the 169,000 documents is represented by its tokens. Each token is assigned a lemma and multiple word senses. Two different word sense disambiguation systems were applied, namely the UBC-ALM [2] and NUS-PT [5] system (abbreviated as UBC and NUS, respectively, in the remainder of the paper). In total, the document collection consists of approximately 100 million tokens including stop words. The NUS annotated corpus comes with around 199 million sense annotations including the sense probability scores, i.e. on average 2 senses per token. The UBC annotated corpus even consists of around 275 million sense annotations and probability scores, i.e. on average 2.75 senses per token. The accuracy of word sense annotations can highly influence the retrieval performance when utilizing word senses (see e.g. Sanderson [10]). Preliminary experiments on the training topics have shown that restricting the incorporated senses to the highest scored sense for each token increases the MAP of retrieval.

Further, we hypothesize that combining the NUS and UBC sense assignments increases the precision of annotated word senses. Therefore, we created several indices for our experiments. Each index consists of three fields, namely token, lemma, and sense. The indexed senses vary in the way they are selected. Four different indices were created: (i) an index with the highest scored UBC sense for each token (**UBCBest**), (ii) an index with the highest scored NUS sense for each token (**NUSBest**), (iii) an index with senses that were assigned by both systems and have the greatest sum of scores (**CombBest**), and finally (iv) an index with senses as in (iii), but where we chose the sense with the highest score from the UBC or NUS corpus when the intersection of the set of senses that were assigned by both systems is empty (**CombBest$^+$**). The construction of **CombBest** can be formally described by:

$$sense(t) = \underset{s \in S(t)}{\operatorname{argmax}} \quad score^{UBC}(s) + score^{NUS}(s) \tag{1}$$

with $S(t) = S^{UBC}(t) \cap S^{NUS}(t)$, where $S^{UBC}(t)$ is the set of senses of token t obtained from the UBC system and $S^{NUS}(t)$ is the sense set accordingly obtained from the NUS system. Thus, $S(t)$ is the intersection of the senses of token t annotated from the UBC and NUS systems. Further, $score^{UBC}(s)$ and $score^{NUS}(s)$ is the probability score assigned to sense s from the UBC and NUS system. If no probability score is assigned $score^{UBC}(s)$ and $score^{NUS}(s)$ returns 0, respectively. Accordingly, **CombBest$^+$** is defined as:

$$sense(t) = \begin{cases} \operatorname{argmax}_{s \in S(t)} \quad score^{UBC}(s) + score^{NUS}(s) & \text{if } S(t) \neq \emptyset \\ \operatorname{argmax}_{s \in S^+(t)} \quad score^{UBC}(s) + score^{NUS}(s) & \text{otherwise} \end{cases} \tag{2}$$

where $S^+(t) = S^{UBC}(t) \cup S^{NUS}(t)$ is the union of the sense sets of token t from the UBC and NUS systems.

Prior to indexing, we applied standard stopword removal. Without stopwords, all indices consists of approximately 40.7 million tokens. As shown in the third column of Table 1 the UBCBest index contains around 34.1 million senses, the NUSBest index contains around 34.5 million senses, i.e. 6.6 million and 6.2 million tokens are not annotated with any sense in the UBCBest and NUSBest index, respectively. The CombBest index contains only 31.7 million senses, while the CombBest$^+$ index consists of 35.1 million senses.

2.2 Retrieval Models

We carried out several retrieval experiments using the Divergence From Randomness BM25 model (DFR_BM25). Often, such PMs have problems dealing with synonymy. This problem, also called *lexical gap*, arises from alternative ways of expressing a concept using different terms. Query expansion models try to overcome the lexical gap problem by reformulating the original query to increase the retrieval performance. We chose the the Kullback-Leibler (KL) query expansion model [6], since it performed best on the training data. In our experiments the original query is expanded by up to 10 most informative (highest weighted) terms from the 3 top ranked documents.

A further solution to the lexical gap problem is the integration of monolingual TMs first introduced by Berger and Lafferty [3]. These models encode statistical word associations which are trained on parallel monolingual document collections such as question-answer pairs. Recently, Bernhard and Gurevych [4] successfully applied TMs for the task of answer finding. In order to automatically train the TMs, they used the definitions and glosses provided for the same term by different lexical semantic resources, namely WordNet, Wiktionary, Wikipedia, and Simple Wikipedia yielding domain-independent TMs. The authors have shown that their models significantly perform better than baseline approaches for answer finding. In our experiments we employed the model defined by Xue et al. [11] and used by Bernhard and Gurevych [4]:

$$P(q|D) = \prod_{w \in q} P(w|d)\,, \tag{3}$$

where

$$P(w|d) = (1 - \lambda)P_{mx}(w|d) + \lambda P(w|D)\,, \tag{4}$$

$$P_{mx}(w|d) = (1 - \beta)P_{ml}(w|d) + \beta \sum_{t \in d} P(w|t)P_{ml}(t|d)\,, \tag{5}$$

q is the query, d the document, λ the smoothing parameter for the document collection D and $P(w|t)$ is the probability of translating a document term t to the query term w. The parameter β was set to 0.8 and λ to 0.5.

We applied the TM trained for the answer finding task, though it was not particularly trained for our task. As the TM was trained on tokens, we apply it on the indexed token field exclusively.

Table 1. Number of indexed word senses and MAP on retrieval (model: DFR_BM25 + KL) for different index types

index type	# senses	MAP
UBCBest	34.1 million	0.2636
NUSBest	34.5 million	0.3473
CombBest	31.7 million	0.3313
CombBest$^+$	35.1 million	**0.3551**

2.3 Combination of Retrieval Models

Our hypothesis is that TMs retrieve different documents for some queries than PMs. Therefore, we compute a combined relevance score to improve the retrieval performance. First, we normalize the scores resulting from each model applying standard normalization:

$$r_{norm}(i) = \frac{r_{orig}(i) - r_{min}}{r_{max} - r_{min}} , \tag{6}$$

where $r_{orig}(i)$ is the original score, r_{min} is the minimum, and r_{max} is the maximum occurring score for a query. Second, we combine the normalized relevance scores computed for individual models into a final score using the CombSUM method introduced by Fox and Shaw [8]. This method ranks the documents based on the sum of the (normalized) similarity scores of individual runs. Each run can be assigned a different weight.

3 Retrieval Results

3.1 Preliminary Experiments on Word Senses

As stated in Section 2.1 we created four indices which differ in the way word senses assigned by the UBC and NUS systems are selected. Table 1 shows the number of indexed word senses for the total number of 40.7 million tokens and the MAP values of different retrieval experiments applying the DFR_BM25 ranking model with KL query expansion. Retrieval on the UBCBest index shows a MAP value of 0.2636. For retrieval based on the NUSBest index the MAP value increases by 24.1%. According to this extrinsic evaluation, the NUS system clearly outperforms the UBC system. While CombBest does not increase the retrieval performance measured by MAP (0.3313), we were able to significantly[1] increase the MAP value using the CombBest$^+$ index up to 0.3551. In the remainder of this paper, we use the indices CombBest and CombBest$^+$ as our intention was to analyze the performance of the heuristic-based combination approach. The runs that we officially submitted are based on the CombBest index only.

[1] We used a two-tailed paired t-test ($\alpha < 0.05$) to determine the statistical significance.

Table 2. MAP values of the different retrieval models and index fields

retrieval model	token	lemma	sense CombBest	sense CombBest$^+$
TM	0.3616	-	-	-
DFR_BM25	0.3741	0.4054	0.2867	0.3096
DFR_BM25 + KL	0.4223	**0.4451**	0.3313	0.3551

3.2 Stand-Alone Retrieval Models

Table 2 shows the MAP of the different models. The TM is always restricted to the indexed tokens; the PM can use all different fields. We did not perform any fine-tuning on the parameters. The TM and the DFR_BM25 model without any query expansion show similar MAP values. However, when applying query expansion the DFR_BM25 approach outperforms the TM. The DFR_BM25 model with query expansion on tokens yields a MAP value of 0.4223 while we get a MAP value of 0.4451 on lemmas, which is an improvement of 5.1%. Experiments on senses achieve the lowest performance ranging from 0.2867 up to 0.3551. Applying query expansion on the CombBest and CombBest$^+$ index outperforms the runs without query expansion. In the following, we focus on experiments applying the DFR_BM25 model with query expansion (hereafter referred to as PM) and the TM.

3.3 Combination of Retrieval Models

We extensively experimented on the training data with different combination weights for the PM and TM using the CombSUM method described in Section 2.3. The combination achieves best performance when the PMs based on tokens and lemmas were assigned a higher weight (due to their higher MAP values) than the model based on senses or the TM. Table 3 illustrates the results obtained on the test topics by different combinations, with and without the integration of word senses. The presented weight combinations yielded best performance on the training data.

Two combinational aspects are of particular interest. The combination of the PMs based on tokens and lemmas yields no improvement (as no sense annotations are used CombBest and CombBest$^+$ yield equal performance). In contrast, the combinations of the PM with the TM always leads to an improvement. Even if the impact of the TM, i.e. its weight, is low (here: 0.2), the MAP values significantly increase when compared to the results obtained by the PM alone, on the token and lemma index fields. This fact corroborates our hypothesis that the PM and the TM retrieve different sets of relevant documents for some queries and that those different sets are effectively combined applying the CombSUM approach.

The second interesting aspect concerns the integration of word sense information. As listed in Table 1 retrieval based on senses from the CombBest index yields a MAP of 0.3313, while retrieval based on senses of the CombBest$^+$ index

Table 3. MAP values and weights for the combination of different models, using the CombBest and CombBest$^+$ indices. The settings marked with a '*' were submitted.

model:field	weights for combinations without word sense annotations				weights for combinations with word sense annotations				
TM:token	-	0.2	0.2	0.2	–	–	0.1	0.1	0.1
PM:token	0.5	0.8	-	0.4	0.8	–	0.8	–	0.4
PM:lemma	0.5	-	0.8	0.4	–	0.8	–	0.8	0.4
PM:sense	-	-	–	–	0.2	0.2	0.1	0.1	0.1
index type	MAP values				MAP values				
CombBest	0.4409	0.4316*	**0.4509***	0.4500*	0.4303	0.4461*	0.4330*	**0.4500***	0.4481*
CombBest$^+$					0.4327	0.4473	0.4331	**0.4507**	0.4480

shows a MAP of 0.3551. We attribute the difference to the fact that CombBest loses information about the documents due to the smaller amount of indexed senses. However, all combinations either with the CombBest or the CombBest$^+$ senses end up with a very similar performance. The reason could be that the loss of information when using the CombBest index is compensated by querying the tokens or lemmas as well.

In some combinational variations, the integration of word senses could achieve a higher MAP value than retrieval settings without word senses. For example, the MAP value corresponding to the retrieval based on tokens alone is 0.4223 (see Table 1), while the combination with senses obtains a MAP value of 0.4303 for the CombBest index and even 0.4327 for the CombBest$^+$ index. However, for the combination based on lemmas and senses, the difference is not significant. Overall, the best performance is obtained by the combination of the TM and the PM based on lemmas and senses, applying weights of 0.1, 0.8, and 0.1, respectively.

3.4 Discussion

In the previous section, we described all our experiments carried out on the document collection disambiguated with word senses. We submitted five runs without the integration of word senses and five further runs utilizing the annotated word senses. According to the MAP values our runs without word senses ended up in the 1st place out of 10 participants. Our highest MAP value could be achieved with the combination of the TM and the PM based on lemmas and the assigned weights of 0.2 and 0.8, respectively. When utilizing word senses, the combination of the TM and the PM based on both lemmas and senses obtains the 1st place according to the MAP as well. We mistakingly submitted runs on the CombBest index, even though we planned to focus on the CombBest$^+$ index. However, we have shown that the differences between the combinational approaches are minimal. Our best performing submitted retrieval setting achieved a MAP value of 0.4500, whereas the second top scoring system in the official challenge obtains a MAP value of 0.4346.

We increased the precision of WSD annotations through a heuristic-based combination of the UBC and NUS annotated senses, which we evaluated extrinsically. This evaluation has shown that the accuracy of annotated word senses highly influences the outcome of retrieval systems. However, we could not observe any improvement by applying the sense annotations compared to the retrieval settings based on tokens or lemmas only. This observation is consistent with the conclusion of last years' challenge.

Regarding the performance of the TM, the results on the combination are promising given that we merely applied a TM built for a previous application in the field of answer finding. The main drawback of the straightforward use is the discrepancy in the tokenization scheme. The tokenization of the document collection is not always compatible with the tokenization of the parallel corpora used for training the TM. In addition, the TM we used contains only tokens and thus cannot deal with indexed multiword expressions. For instance, the phrase "public transport" is indexed as "public_transport". In the TM the two terms "public" and "transport" appear, but not the phrase "public_transport". We quickly analyzed the amount of multiword expressions in the test topic collection. In fact, 61 queries out of the 160 test queries contain at least one multiword expression. This analysis shows that the TM was not particularly trained for this task and motivates further improvements. In addition, the TM could be trained on lemmas and senses. The latter option, however, requires a word sense disambiguated monolingual parallel corpus.

4 Conclusions

We have described a combinational approach to information retrieval on word sense disambiguated data, which combines a PM and a monolingual TM. For the PM we have used the DFR_BM25 model with the KL query expansion method. For the TM we have applied a model which was already trained for an answer finding task. Our aim was to assess the benefits of the combination of both models. We have shown that the combinational approach always achieves better performance than the stand-alone models.

Our second goal was to analyse different methods for selecting word senses from annotated corpora in order to increase their accuracy. We have discovered that our heuristic-based approach CombBest$^+$ increases the retrieval performance based on word senses by 2.2% when compared to NUSBest and even 25.8% when compared to UBCBest. The huge difference between NUSBest and UBCBest demonstrates that WSD accuracy is essential for utilizing word sense information. However, the experiments on the CombBest$^+$ index have shown that we could only increase the retrieval performance in one specific case: by combining the PM based on tokens with the same model based on senses. Nevertheless, other combinations without word senses outperformed this setting easily.

Acknowledgements

This work has been supported by the Emmy Noether Program of the German Research Foundation (DFG) under the grant No. GU 798/3-1, and by the Volkswagen Foundation as part of the Lichtenberg-Professorship Program under the grant No. I/82806.

References

1. Agirre, E., Di Nunzio, G.M., Mandl, T., Otegi, A.: CLEF 2009 Ad Hoc Track Overview: Robust-WSD Task. In: Multilingual Information Access Evaluation Text Retrieval Experiments. LNCS, vol. 1, Springer, Heidelberg (2010)
2. Agirre, E., Lopez de Lacalle, O.: UBC-ALM: Combining k-NN with SVD for WSD. In: Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007), Prague, Czech Republic, pp. 342–345 (2007)
3. Berger, A., Lafferty, J.: Information Retrieval as Statistical Translation. In: Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 222–229 (1999)
4. Bernhard, D., Gurevych, I.: Combining Lexical Semantic Resources with Question & Answer Archives for Translation-Based Answer Finding. In: Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP, Suntec, Singapore, pp. 728–736 (2009)
5. Chan, Y.S., Ng, H.T., Zhong, Z.: NUS-PT: Exploiting Parallel Texts for Word Sense Disambiguation in the English All-Words Tasks. In: Proceedings of the Fourth International Workshop on Semantic Evaluations (Sem Eval-2007), Prague, Czech Republic, pp. 253–256 (2007)
6. Cover, T.M., Thomas, J.A.: Elements of Information Theory. Wiley-Interscience, New York (1991)
7. Dolamic, L., Fautsch, C., Savoy, J.: UniNE at CLEF 2008: TEL, and Persian IR. In: Peters, C., et al. (eds.) CLEF 2008. LNCS, vol. 5706, pp. 178–185. Springer, Heidelberg (2009)
8. Fox, E.A., Shaw, J.A.: Combination of Multiple Searches. In: Proceedings of the 2nd Text REtrieval Conference (TREC-2), pp. 243–252 (1994)
9. Ounis, I., Amati, G., Plachouras, V., He, B., Macdonald, C., Lioma, C.: Terrier: A High Performance and Scalable Information Retrieval Platform. In: Proceedings of the ACM SIGIR Workshop on Open Source Information Retrieval (2006)
10. Sanderson, M.: Word Sense Disambiguation and Information Retrieval. In: Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval, New York, NY, USA, pp. 142–151 (1994)
11. Xue, X., Jeon, J., Croft, W.B.: Retrieval Models for Question and Answer Archives. In: Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval, New York, NY, USA, pp. 475–482 (2008)