Alicante at CLEF 2009 Robust-WSD Task

Javi Fernández, Rubén Izquierdo, and José M. Gómez

University of Alicante, Department of Software and Computing Systems
San Vicente del Raspeig Road, 03690 Alicante, Spain
javier.fernandez@eltallerdigital.com, ruben@dlsi.ua.es, jmgomez@ua.es
http://www.dlsi.ua.es

Abstract. In this paper we explore the use of semantic classes in an Information Retrieval system in order to improve the results in the Robust-WSD task at CLEF 2009. We use two ontologies of semantic classes (WordNet domain and Basic Level Concepts) to re-rank the retrieved documents and obtain better recall and precision. Finally, we implement an innovative method to weight the expanded terms taking into account the ones of the original query terms and their relations in WordNet with respect to the new ones which have demonstrated to improve the results.

1 Introduction

The two main goals of the Robust-WSD last year's edition task were to measure the robustness of the retrieval systems (good stable performance over all queries), and test the benefits of the use of Word Sense Disambiguation (WSD) on this kind of systems.

For our participation in the second edition, we decided to employ a system already implemented and evaluated in the last year edition as starting point, the one of Universidad Complutense de Madrid [10] due to its good results, availability and the possibility of easily adjusting the code to our objectives.

Our main strategy consists of experimenting the benefits semantic classes in Information Retrieval (IR) systems. Moreover, we propose an innovative and flexible way of weighting terms for the query expansion based on WordNet relations.

WSD, can be defined as the task of assigning the correct sense to words depending on the context in which they appear. It is a challenging task, difficult to be addressed and, despite the long time it is been studied, the results of state-of-art WSD systems are still a long way to be useful in other Natural Language Procesing applications, as shown in last international evaluations [15, 12]. Generally, supervised systems obtain better results than the unsupervised ones on previous cited international evaluations. The annotated corpora used commonly in supervised approaches are tagged manually by lexicographers with word senses taken from a particular lexical semantic resource (most commonly WordNet [4].) This tool has been widely criticized for being a sense repository that often provides too fine–grained sense distinctions. On the one hand, too fine–grained senses are not useful for higher level applications like Machine Translation or Question

Answering. On the other hand, it seems that many word–sense distinctions are too subtle to be captured by automatic systems with the current small volumes of word–sense annotated examples. This could be a reason of the poor results of current WSD systems.

A possible solution that has been explored is the use of semantic classes instead of word senses. The task of WSD consists of assigning the proper semantic class to each ambiguous word, instead of its word sense. The use of semantic classes has several advantages. Firstly, they provide richer and more useful information than word senses. For example, for IR could be more informative that the word church belongs to the semantic class BUILDING, instead of knowing that the correct sense to that word is the 1. Secondly, the average polysemy of texts is decreased with the use of semantic classes; in fact they can group in the same class several senses of a concrete word. Therefore, the classification task is simplified, and finally, the amount of training data for each classifier is increased because semantic classes can group together senses of different words, senses of the same word, and also senses of word of different morphological categories. As a consequence, the number of examples to train each classifier is increased in semantic class approaches, and the problem of the lack of data in alleviated.

In [5] they empirically explored on the supervised WSD task the performnace of different levels of abstraction provided by WordNet Domains [8], SUMO labels [9], Lexicographer Files of WordNet [4] and Basic Level Concepts [6]. They referred to this approach as class–based WSD since the classifiers were created at a class level instead of at a sense level. As we abovementioned, class–based WSD cluster senses of different words into the same explicit and comprehensive grouping. Only thoses cases belonging to the same semantic class are grouped to train the classifier. For example, the coarser word grouping obtained in [14] only has one remaining sense for "church". Using a set of Base Level Concepts [6], the three senses of "church" are still represented by *faith.n#3*, *building.n#1* and *religious_ceremony.n#1.*

We are convinced that IR could take advantage of the use of word sense disambiguation, from a semantic class point of view instead from the traditional word sense point of view. Due to the data of the robust adhoc IR task has been processed automatically by two WSD systems, and the information of word senses is available, we do not run any class–based WSD system over the data.

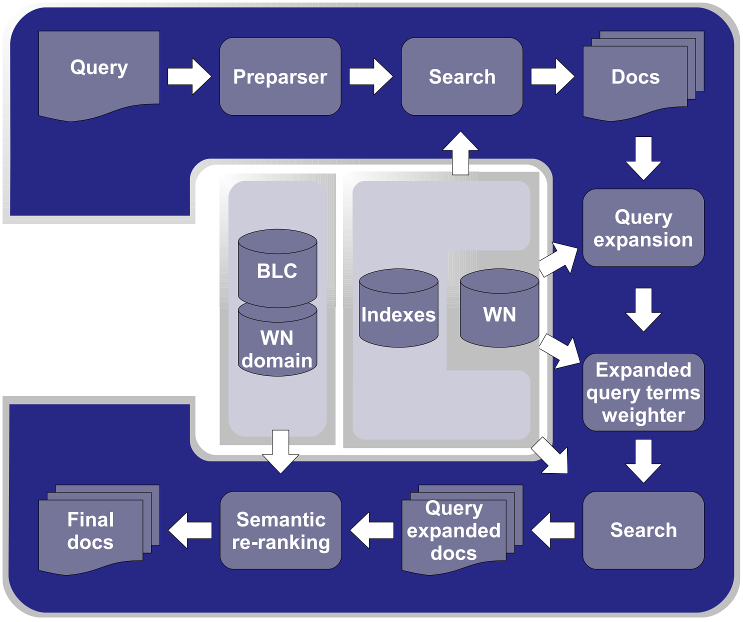
This paper is organized as follows: the next section describes the architecture of our system. In section 3 we discuss the results of this system at CLEF 2009 Robust-WSD Task. Finally, in section 4 we draw the conclusions and future works.

2 Description of the System

The system architecture is shown in Figure 1.

As we can see in Figure 1, the user query is pre-parsed to obtain a set of terms without stopwords and any special symbol. Next, a ranked list of relevant docu-

Fig. 1. Architecture of the system



ments are retrieved using Lucene search engine[1]. With the retrieved documents, the initial query, the relations of the external resource WordNet and state-of-art query expansions methods an expanded query is obtained. The terms of this new query are weighted taking into account the weights of the original query words, their relations in WordNet with respect to the new ones, the weight assigned by the WSD system to each sense and the weight returned by the expansion method. Once we obtain a new list of weighted terms, we do, again, a search but this time using the expanded query instead of the original one in order to retrieved a new ranked list of documents. Finally, we employ the semantic class information from two semantic resources (WordNet Domains and Base Level Concepts) in order to obtain a re-ranked document list as result.

The following sections present each of this processes in more detail.

2.1 Search engine and query expansion

As we mentioned previously, the search engine we employ is the one provided by the Universidad Complutense de Madrid . Its implementation is a modified version of Lucene which uses the BM25 probabilistic model [13] for the document retrieval. They have also implemented two state-of-art query expansion methods: Kullback-Liebler Divergence [2, 3] (an information-theoretic approach) and the Bo1 model [7, 11] (based on Divergence From Randomness [1]). We selected the

[1] http://lucene.apache.org

Bo1 model since this approach obtains the best results. We also decided to use the same constant values than they used in the last CLEF Robust-WSD edition in order to compare the effectiveness of our methods of semantic classes.

As we can see in the Figure 1, we make two search processes. For the first retrieval process, query terms are lemmatized and stemmed in order to increase the system recall. The first *Search* module gets these terms as input and returns a list of relevant documents using the BM25 probabilistic model. Furthermore, in the *Query expansion module*, we expand the original query obtaining new terms by means of the Bo1 model.

Even if [10] proposed a method for weighting the expanded query terms based on WordNet, we preferred to use our own in fact they do not use all senses of each term but the weightest one. We decided to use all senses retrieved by the WSD system in order to improve the recall. In this way, the system searches all expanded terms in the relations of WordNet with respect to the synonyms, hyperonyms and hyponyms until a certain level or *distance*. For example, if the distance is 2, we search any expanded term among the hyperonym and hyponym synsets of the original terms but, also, the hyperonyms of the hyperonyms and the hyponyms of the hyponyms. The distance constant marks the allowed jumps to reach in the WordNet relations from the synsets of the query terms. We use all senses supplied for the WSD system for each query term taking into account the score given by these systems to each sense in order to calculate the weight of the expanded terms. Thefore, this distance factor is calculated by the following equation:

$$weight(synset_{i,d}) = weight(synset_{i,d-1}) * \alpha^{d} \tag{1}$$

We defined $synset_{i,1}$ as a WordNet synset given and $synset_{i,d}$ as another WordNet synset which is related to the $synset_{i,1}$ of a distance of d jumps (taken into acount only hyperonym and hyponym relations). Thus, $weight(synset_{i,d})$ is the weight of the synset i,d and $weight(synset_{i,1})$ is the score given by the WSD system to the synset i,d. α is a constant whose value is between 0 and 1 and d the distance of $synset_{i,d}$ to the $synset_{i,1}$.

Once we calculated the previous synset weight, we combine this weight with the weight assigned by the expanded method bo1 in order to calculate the final term weight using the following equation:

$$weight(term_t) = \frac{weight(synset_{i,d}) + weight_0(term_t)}{2} \tag{2}$$

Where $weight(term_t)$ is the weight of the expanded term t which is grouped in the WordNet synset i,d, and $weight_0(term_t)$ is the weight assigned by bo1 to the term t.

Using these equations, we give importance to those expanded terms closely related with the original query terms and, in addition, we include the score given by the WSD system for each query term in the final term weight. Thus, we include all senses of a term in the search giving more importance those terms which are relationed with more likely senses and closer to the original query terms.

2.2 Semantic classes

Our approach consists on mapping the assigned word senses to semantic classes, specifically to WordNet Domains labels and Basic Level Concepts.

WordNet Domains [8] is a hierarchy of 165 Domain Labels used to label all the WordNet synsets. Information brought by Domain Labels is complementary to what is already in WordNet. First of all, a Domain Label can include synsets of different syntactic categories: for instance MEDICINE groups together with senses from nouns, such as doctor or hospital, and from verbs, such as to operate. Second, a Domain Label may also contains senses from different WordNet sub hierarchies. For example, SPORT contains senses such as athlete, deriving from life form, game equipment from physical object, sport from act and playing field from location.

Basic Level Concepts [6] are a set of concepts that result from the compromise between two conflicting principles of characterization:

- Represent as many concepts as possible;
- Represent as many features as possible;

As a result, Basic Level Concepts typically occurs in the middle of hierarchies and less than the maximum number of relations.

The authors developed a method for the automatic selection of BLC from WordNet. They use a very simple method for deriving a small set of appropriate meanings using basic structural properties of WordNet. The approach considers:

- The total number of relations of every synset or just the hyponymy relations
- Discard those BLCs that do not represent at least a number of synsets.
- Optionally, the frequency of the synsets (summing up the frequency of the senses provided by WordNet).

The process of automatic selection of BLC follows a bottom-up approach using the chain of hypernym relations. For each synset in WN, the process selects as its Base Level Concept the first local maximum according to the relative number of relations. For synsets having multiple hypernyms, the path having the local maximum with higher number of relations is selected. Usually, this process ends with a number of fake Base Level Concepts. That is, synsets having no descendants (or with a small number) but being the first local maximum according to the number of relations considered. Thus, the process concludes checking if the number of concepts subsumed by the preliminary list of BLC is higher than a certain threshold. For those BLC not representing enough concepts according to a certain threshold, the process selects the next local maximum following the hypernym hierarchy.

Thus, depending on the type of relations considered to be counted and the threshold established, different sets of BLC can be easily obtained for each WN version. For our work, we selected the set of BLC built using all kind of relations and a threshold of 20 as the minimum number of synsets that each BLC must subsume.

We explain now the **representation of documents or queries with semantic classes** of the words contained on them. In the task data, each ambiguous word is annotated with their possible senses, each one with a certain probability. Starting from this information, we create a domain vector, for a query or for a document, containing all the semantic classes information of the query or document. The domain vector consists in a vector which, each element, represents a WordNet Domain or a Basic Level Concept and its associated weight. Note that there are 165 Domain Labels in WordNet Domains and 558 Basic Level Concepts for nouns. The way to build this vector is: each word has annotated several senses, with the associated probability; each word sense is mapped to its proper semantic class, and the element of the vector corresponding to this domain is increased with the probability associated to the word sense. After processing all terms, we obtain a domain vector representing the semantic information of the document or query. Finally to compare two documents, or a document and a query, and obtain their similarity in terms of their semantic content, we use the value of the cosine defined by the two domain vectors.

2.3 Integration of Semantic classes in Robust Ad hoc

Once the final list of documents from the expanded query is retrieved, the *Semantic re-ranking* module re-arrange this list taking into account both the similarity returned by the BM25 probabilistic model and the similarity calculated by semantic class system. In order to do this, we studied the following 5 different equations:

$$semsim_1(i,j) = sim_{ij} * sem_{ij} \tag{3}$$

$$semsim_2(i,j) = \begin{cases} simmax_i + sim_{ij} & \text{if } sem_{ij} > h \\ \\ sim_{ij} & \text{otherwise} \end{cases} \tag{4}$$

$$semsim_3(i,j) = \begin{cases} (simmax_i + sim_{ij}) * sem_{ij} & \text{if } sem_{ij} > h \\ \\ sim_{ij} & \text{otherwise} \end{cases} \tag{5}$$

$$semsim_4(i,j) = \begin{cases} simmax_i + sim_{ij} * sem_{ij} & \text{if } sem_{ij} > h \\ \\ sim_{ij} & \text{otherwise} \end{cases} \tag{6}$$

$$semsim_5(i,j) = simmax_i * sem_{ij} + sim_{ij} \tag{7}$$

Where $semsim_x(i,j)$ is the final similarity between the query i and the document j using the method x, sim_{ij} is the similarity of the query i with respect to the document j returned by the search engine, sem_{ij} is the same similarity but returned by the semantic class system, $simmax_i$ is the greatest value of similarity returned by the search engine for the query i and h is a constant which determines a semantic similarity threshold defined empirically.

As both similarity values (sim_ij and sem_ij) are normalized ones, our first approximation was the equation 3. In this equation we simply multiply both values (the similarity returned by the search engine and the similarity obtained by the semantic class system) in order to obtain a new similarity value between the query i and the document j. With this equation both values have the same importance.

The equation 4 was thought in order to put those documents with a certain level of semantic relation with the query above other ones. Therefore, if the similarity obtained by the search engine will be summed to the greatest similarity if, and only if, the semantic class similarity between the document and the query is greater than a threshold h, otherwise only the BM25 similarity will be taken into account.

The next equation (5) is based on the previous one, however the semantic similarity multiplies the sum of the BM25 similarity for the document j and the greatest similarity. The main idea of this equation is to overlap some of the less semantic related documents which exceed the threshold with those which do not exceed.

In order to give more relevance those documents with high semantic similarity but taking into account the semantic class score in the final similarity value, the equation 6 was used.

Finally, the equation 7 multiplies the greatest similarity value with the document semantic similarity and, next, sums this result to the search engine document similarity. This equation tries to improve the search engine similarty value using the semantic similarity as reduction factor of maximum similarity value.

3 Evaluation

In this section we report the results of each one of our proposals.

For the evaluation of the *Expanded query terms weighter*, we set the value for two variables: α and d (distance). In order to get the best values for these variables, we experimented with several values for them. Table 1 presents two of the best results of those experiments. With $\alpha = 0.8$ and $d = 1$ we improve the baseline GMAP in a 9.97%. With $\alpha = 0.92$ and $d = 6$ we improve both the baseline MAP in a 0.02% and the baseline GMAP in a 8.19%. The results of *BM25+Bo1+WD (*$\alpha = 0.92$*,* $d = 6$*)* correspond to our CLEF experiment named *ali02wsd*.

The results of our second proposal, the *Semantic re-ranking module*, depend on the function used for the integration of the documents' weights given by the semantic classes and the search engine. In addition, for some of the integration functions, the variable h (threshold) has to be set too. We experimented with those functions and different values for the threshold. For each integration func-

Table 1. Evaluation of the *Expanded query terms weighter* module

	MAP	GMAP	R-Prec	P@5	P@10
BM25 + Bo1 (Baseline)	.3737	.1294	.0.3585	.4475	**.3825**
BM25 + Bo1 + WD ($\alpha = 0.8$, $d = 1$)	.3706	**.1423**	.3624	.4500	.3750
BM25 + Bo1 + WD ($\alpha = 0.92$, $d = 6$)	**.3738**	.1400	**.3655**	**.4513**	.3775

tion we obtained its best threshold value (when needed), as can be seen in Tables 2 and 3.

Table 2. Evaluation of the *Semantic re-ranking* module with WND and different integration functions

	MAP	GMAP	R-Prec	P@5	P@10
BM25 + Bo1 + WND + RR1	**.3752**	**.1298**	**.3638**	**.4462**	**.3862**
BM25 + Bo1 + WND + RR2 ($h = 0.2$)	.3737	.1294	.3585	.4475	.3825
BM25 + Bo1 + WND + RR3 ($h = 1.0$)	.3737	.1294	.3585	.4475	.3825
BM25 + Bo1 + WND + RR4 ($h = 0.5$)	**.3752**	**.1298**	**.3638**	**.4462**	**.3862**
BM25 + Bo1 + WND + RR5	.3746	.1296	.3592	.4463	.3856

As we can see in Table 2, some of the results of different integration functions are the same. In the case of RR2, RR3 and RR5, this occurs because the best results they can reach are the same as the baseline results. This integration functions do not improve the system. In the case of RR1 and RR4, they mathematical function are the same, except the second one, that can be affected by the threshold. Both of them obtain the best results for the WND model for all measures.

Table 3. Evaluation of the *Semantic re-ranking* module with BLC20 and different integration functions

	MAP	GMAP	R-Prec	P@5	P@10
BM25 + Bo1 + BLC20 + RR1	.3533	.1231	.3375	.4337	.3619
BM25 + Bo1 + BLC20 + RR2 ($h = 0.8$)	**.3776**	**.1317**	**.3609**	**.4437**	**.3806**
BM25 + Bo1 + BLC20 + RR3 ($h = 0.8$)	**.3776**	**.1317**	**.3609**	**.4437**	**.3806**
BM25 + Bo1 + BLC20 + RR4 ($h = 0.8$)	**.3776**	**.1317**	**.3609**	**.4437**	**.3806**
BM25 + Bo1 + BLC20 + RR5	.3375	.1170	.3229	.4213	.3625

Table 3 presents the results of the integration with the BLC20 model. The best results are obtained by the RR2, RR3 and RR4 integration functions. These are the functions that use a threshold. Note the threshold that gives the best results is the same for all them. Thus, we can deduce that the threshold $h = 0.8$ is the ideal for determining if a semantic class is relevant or not.

For our final comparison, we chose the best integration function for each model (WND, BLC20), as shown on Table 4 and Figure 2. The results of *BM25+Bo1+WND* correspond to our CLEF experiment named *ali01wnd*.

Fig. 2. Results of MAP and GMAP for models WND and BLC20 and each integration function

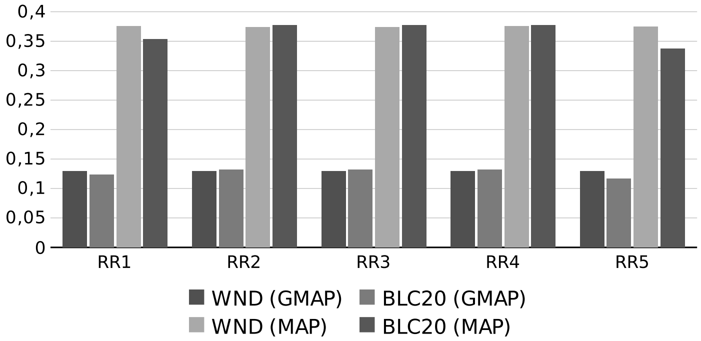


Table 4. Evaluation of the *Semantic re-ranking* module

	MAP	GMAP	R-Prec	P@5	P@10
BM25 + Bo1 (Baseline)	.3737	.1294	.3585	**.4475**	.3825
BM25 + Bo1 + WND	.3752	.1298	**.3638**	.4462	**.3862**
BM25 + Bo1 + BLC20	**.3776**	**.1317**	.3609	.4437	.3806

The integration of the semantic classes to the search engine improves the baseline results. With WND we improve both the baseline MAP in a 0.4% and the baseline GMAP in a 0.31%. With BLC20 we improve both the baseline MAP in a 0.64% and the baseline GMAP in a 1.77%.

4 Conclusions

The results of the experiments with our two proposals demonstrate improvements to the initial IR system. Regarding the *Expanded query terms weighter* module, we experimented with the weights of the terms in a probabilistic IR system. We have applied a smoothing function based on the WordNet distance to the weights given by the IR system. The experiments show GMAP improvements of nearly 10% but not significant MAP improvements.

As future work we propose to continue with the experiments on this module. For the propagation function 2, the search of the best values for α and d can be more exhaustive, finding better values for this variables. Moreover, new relations can be explored in WordNet (not only hyponyms and hyperonyms), in order to improve recall. Even new weight propagation functions can be proposed to better exploit the concept of *distance* in WordNet.

Regarding our second proposal, the *Semantic re-ranking* module, we have integrated the semantic classes to a IR system. We carried out this integration recalculating the weight of the documents retrieved depending on the similarity between the semantic class of each document and the semantic class of the query. The results of the experiments made reveal that the semantic classes resources can be effectively be integrated to the IR systems.

This module can also be led to new levels. We only used five simple integration functions for the search engine and the semantic classes weights. More functions can be studied with the purpose of finding the best way to integrate the available resources of semantic classes.

Acknowledgements

This paper has been partially supported by the next projects: "Question Answering Learning technologies in a multiLingual and Multimodal Environment (QALL-ME)" (FP6 IST-033860), "Intelligent, Interactive and Multilingual Text Mining based on Human Language Technologies (TEXT-MESS)" (TIN2006-15265-C06-01) and "Virtual Observatory for Technology Transfer". We also want to thank the support given by "El Taller Digital".

References

1. Gianni Amati and Cornelis Joost Van Rijsbergen. Probabilistic models of information retrieval based on measuring the divergence from randomness. *ACM Trans. Inf. Syst.*, 20(4):357–389, 2002.
2. Claudio Carpineto, Renato de Mori, Giovanni Romano, and Brigitte Bigi. An information-theoretic approach to automatic query expansion. *ACM Trans. Inf. Syst.*, 19(1):1–27, 2001.
3. Thomas M. Cover and Joy A. Thomas. *Elements of information theory.* Wiley-Interscience, New York, NY, USA, 1991.
4. C. Fellbaum, editor. *WordNet. An Electronic Lexical Database.* The MIT Press, 1998.
5. R. Izquierdo, A.Suárez, and G. Rigau. An empirical study on class-based word sense disambiguation. In *EACL*, pages 389–397. The Association for Computer Linguistics, 2009.
6. R. Izquierdo, A. Suarez, and G. Rigau. Exploring the automatic selection of basic level concepts. In Galia Angelova et al., editor, *International Conference Recent Advances in Natural Language Processing*, pages 298–302, Borovets, Bulgaria, 2007.
7. Craig Macdonald, Ben He, and Vassilis Plachouras and Iadh Ounis. University of glasgow at trec 2005: Experiments in terabyte and enterprise tracks with terrier. In *TREC*, 2005.

8. B. Magnini and G. Cavaglià. Integrating subject field codes into wordnet. In *Proceedings of LREC*, Athens. Greece, 2000.
9. I. Niles and A. Pease. Towards a standard upper ontology. In *Proceedings of the 2nd International Conference on Formal Ontology in Information Systems (FOIS-2001)*, pages 17–19. Chris Welty and Barry Smith, eds, 2001.
10. José R. Peréz Agüera and Hugo Zaragoza. Ucm-y!r at clef 2008 robust and wsd tasks. In *CLEF*, 2008.
11. Vassilis Plachouras, Ben He, and Iadh Ounis. University of glasgow at trec 2004: Experiments in web, robust, and terabyte tracks with terrier. In Ellen M. Voorhees, Lori P. Buckland, Ellen M. Voorhees, and Lori P. Buckland, editors, *TREC*, volume Special Publication 500-261. National Institute of Standards and Technology (NIST), 2004.
12. Sameer Pradhan, Edward Loper, Dmitriy Dligach, and Martha Palmer. Semeval-2007 task-17: English lexical sample, srl and all words. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 87–92, Prague, Czech Republic, June 2007. Association for Computational Linguistics.
13. S. E. Robertson and S. Walker. Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In *SIGIR '94: Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 232–241, New York, NY, USA, 1994. Springer-Verlag New York, Inc.
14. R. Snow, Prakash S., Jurafsky D., and Ng A. Learning to merge word senses. In *Proceedings of Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 1005–1014, 2007.
15. B. Snyder and M. Palmer. The english all-words task. In Rada Mihalcea and Phil Edmonds, editors, *Senseval-3: Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text*, pages 41–43, Barcelona, Spain, July 2004. Association for Computational Linguistics.