Addressing the MFS Bias in WSD Systems

Marten Postma*, Ruben Izquierdo*, Eneko Agirre†, German Rigau†, Piek Vossen*
Vrije Universiteit Amsterdam*; University of the Basque Country†
m.c.postma@vu.nl, ruben.izquierdobevia@vu.nl, e.agirre@ehu.eus, german.rigau@ehu.eus, piek.vossen@vu.nl

Abstract

Word Sense Disambiguation (WSD) systems tend to have a strong bias towards assigning the Most Frequent Sense (MFS), which results in high performance on the MFS but in a very low performance on the less frequent senses. We addressed the MFS bias in WSD systems by combining the output from a WSD system with a set of mostly static features to create a MFS classifier to decide when to and not to choose the MFS. The output from this MFS classifier, which is based on the Random Forest algorithm, is then used to modify the output from the original WSD system. We applied our classifier to one of the state-of-the-art supervised WSD systems, i.e. IMS, and to of the best state-of-the-art unsupervised WSD systems, i.e. UKB. Our main finding is that we are able to improve the system output in terms of choosing between the MFS and the less frequent senses. When we apply the MFS classifier to fine-grained WSD, we observe an improvement on the less frequent sense cases, whereas we maintain the overall recall.

Keywords: WSD, MFS, Random Forest

1. Introduction

Word Sense Disambiguation (WSD) is generally defined as a classification task. The goal is to determine which sense of a word or multi-word expression is used in a linguistic context (Agirre and Edmonds, 2007). In order to be able to compare the performance of the techniques that have been applied to try to solve this task, WSD evaluation tasks have been organized. Izquierdo (2015) conducted an error analysis of five all-words tasks to define the problem space of WSD systems.[1]

The most striking results came from the analysis of comparing the average recall of WSD approaches on instances in which the Most Frequent Sense (MFS) was the gold sense versus when one of the less frequent senses (LFS) was the gold sense (Figure 1). We observe a clear trend that systems excel on MFS cases, exceeding the average recall by far, whereas the recall drops dramatically for all LFS cases, far below the average recall. The average performance of all systems is therefore largely influenced by a very skewed and limited capacity to recognize senses in contexts. The question we address here in this paper is: how can we increase the performance for the LFS part of the task?

The poor recall of WSD systems on the LFS can be explained by at least two observations. Firstly, the systems are biased towards the MFS sense, which is why they opt for the MFS when one of the LFS applies. Secondly in those cases where the WSD systems are correctly distinguishing between the MFS and the LFS, they opt for the wrong sense from the set of the LFS, probably due to lack of training data for the LFS.

We conducted a simple experiment in order to establish what the maximum performance on the LFS is, assuming that we perfectly know when to choose and not to choose the MFS. We ran the UKB system (Agirre et al., 2014) on SemEval–2013 Task 12: Multilingual Word Sense Disambiguation (Navigli et al., 2013). We assumed a perfect distinction between the MFS and the LFS. We then focused on the LFS. When the system wrongly opted for a MFS, we chose from the LFS the sense with the highest confidence. In all other cases, we used the original system output. By doing this, the recall on the LFS jumped from 25% to 57%, which would be a dramatic improvement compared to current WSD systems. The overall recall improved from 66% to 78%. In this paper, we focus specifically on improving the distinction between the MFS and the LFS mainly in order to improve the recall on the LFS, while maintaining or even improving the overall recall.

The outline of this paper is as follows. In Section 2., we describe the previous research on WSD, followed by the system description in Section 3.. Consequently, we present the results in Section 4., which are discussed in Section 5., followed by the conclusion in Section 6..

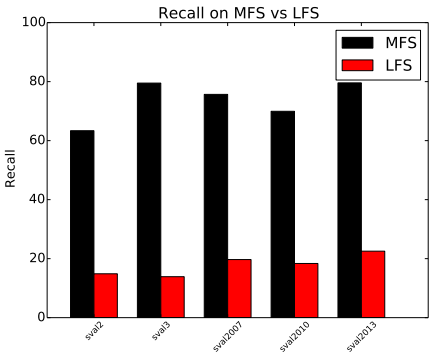

Figure 1: The average recall of all systems is shown on the instances, which includes monosemous instances, in which the gold sense is the MFS compared to those instances in which the gold sense is among the LFS.

[1]The following tasks were taken into consideration: SensEval–2 (sval2): All-Words task (Palmer et al., 2001) ; SensEval-3 (sval3): Task 1: The English all-words task (Snyder and Palmer, 2004) ; SemEval-2007 (sval2007): Task-17: English Lexical Sample, SRL and All Words (Pradhan et al., 2007) ; SemEval–2010 (sval2010): Task 17: All-Words Word Sense Disambiguation on a Specific Domain (Agirre et al., 2010); SemEval–2013 (sval2013): Task 12: Multilingual Word Sense Disambiguation (Navigli et al., 2013).

2. Related Work

The task of WSD is to create a function that maps a lexical expression (lemma or multi-word expression) to a sense given a context. In general, the lexical semantic resource WordNet (Fellbaum, 1998) is used to define what the possible senses of a lexical expression are.

Several properties of this mapping complicate obtaining a good performance in the automatic induction of the mapping. Firstly, the number of possible senses of a lexical expression can be high (as high as 75 in WordNet). This makes it relatively difficult to find training data, since knowledge must be found for each meaning of a lexical expression. Since it is expensive and time-consuming to create sense-labeled data, this often results in a knowledge bottleneck, because it is expensive to do it manually, and there is simply not enough training data to induce machine models to automatically gather new data. Finally, in language usage, the senses are not equally distributed either. In general, one sense is used more often than others, often called the most frequent sense (MFS). Past work has shown that WSD systems tend to have a strong bias towards the MFS (Preiss, 2006).

Unsupervised approaches overcome the knowledge bottleneck problem by not relying on training data, but on the knowledge in a lexical database, e.g. WordNet. Many of the unsupervised approaches are graph-based of which the UKB system (Agirre et al., 2014) is among the best performing ones. The UKB system represents senses as nodes and semantic relations as edges. Firstly, the node weights are initialized using the knowledge from the graph. Next, the node weights are updated with respect to the knowledge found in the local context of a target word, resulting in context-dependent PageRank. In general, unsupervised approaches do not suffer greatly from the knowledge bottleneck problem. However, recent work has shown that they also have a strong bias towards the MFS (Calvo and Gelbukh, 2015).

Supervised approaches attempt to maximize the performance of the mapping function by training word and sense experts using (mostly) sense-labeled training data. The *It makes sense* (IMS) system (Zhong and Ng, 2010) is one of the best performing supervised approaches, which makes use of linear support vector machines with mostly local contextual features. The biggest challenge for supervised approaches is the reliance on manually sense-tagged training data, which is expensive and time-consuming to create, especially for high polysemous words. One of the reasons why IMS is performing so well is that it partly overcomes the knowledge bottleneck problem by making use of parallel data from two different languages and thus generating more training data for the LFS.

Other supervised approaches focus on improving the performance of the mapping function by reducing the number of possible classes. The rationale behind this approach is to reduce the knowledge bottleneck problem by combining the training data from related senses. Good results have been reported for these approaches on WordNet Domains, Supersenses, and Base Level Concepts (Peh and Ng, 1997; Izquierdo et al., 2007).

In this paper, we propose an approach to modify the output from a WSD system using a MFS classifier. We do not attempt to overcome the knowledge bottleneck problem, but we try to correct the systems for their MFS bias by reducing the mapping function to MFS and LFS.

3. System Description

The starting point for our system is the output from a WSD system. We report the results for the UKB and the IMS systems. A feature set containing mostly static features, focusing predominantly on frequency and domain properties of lemmas, is combined with the WSD output and fed into a Random Forest algorithm (Breiman, 2001) to create a MFS classifier, of which the goal is to predict whether the gold sense for an instance is the MFS or among the LFS. Consequently, we use the MFS classifier output to modify the WSD output from the original system. A visual representation of the MFS classifier can be found in Figure 2.

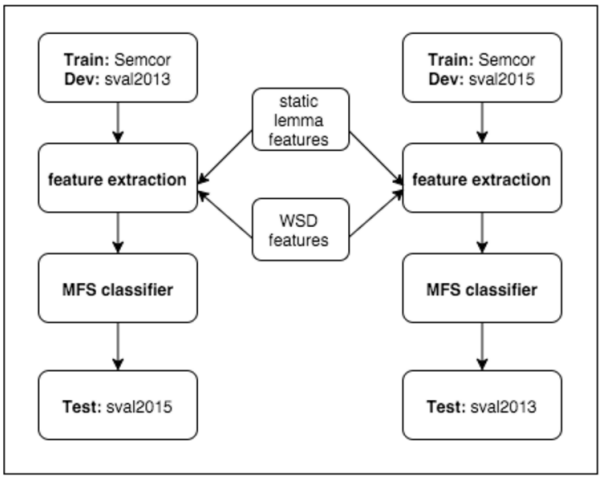

Figure 2: system architecture MFS classifier

Figure 2 presents the architecture for the MFS classifier. Firstly, SemCor (Miller et al., 1993) is used as training corpus. We evaluate on two testing corpora, which are SemEval 2013 task 12: Multilingual Word Sense Disambiguation (Navigli et al., 2013), and SemEval 2015 task 13: Multilingual All-Words Sense Disambiguation and Entity Linking (Moro and Navigli, 2015), or *sval2013*, and *sval2015*, respectively. We evaluate on the non multiword instances from these competitions. sval2013 contains 1514 noun instances, whereas sval2015 consists of 815 instances and contains nouns, verbs, adjectives, and adverbs. In the development phase, sval2013 is used as the development corpus for testing sval2015 and vice versa.

The feature set for the MFS classifier contains features extracted from the output from the WSD system and mostly static features. In the development phase, the features are selected using regression. Each run of the Random Forest classifier contains at least one feature containing information from the WSD system. The output of the WSD system is a set of senses. A confidence value is attributed to each sense in the set. There are three features that use information from the WSD output, which are the *system confidence on the MFS*, the *entropy of the sense ranking*, and the *cor-*

relation between the system sense entropy and the sense entropy in Semcor. In addition, we use mostly static features, which are the same in training, development, and test for a particular lemma. These features include *TF-IDF*, *part of speech*, *number of senses* and *system sense entropy*, as well as *WordNet domains*, and the *WordNet Supersense*. Finally, we use one feature that is dependent on the corpus used in training, development, and test. This feature makes use of the domain classifier JRC EuroVoc Indexer JEX (Steinberger et al., 2012). We compare the domain distribution of Semcor to the domain distribution of the instances of a lemma.

Finally, the output from the MFS classifier and a WSD system are combined to obtain the final sense assignment. The algorithm is visualized in Algorithm 1.

Algorithm 1 Sense assignment

```
for each instance i ∈ test_corpus do
    if output_mfs_classifier[i] == 0
    AND
    1st_system_sense == mfs_sense then
        return 2nd_system_sense
    else
        return 1st_system_sense
    end if
end for
```

For every instance in a testing corpus, the output from the MFS classifier is checked. This output is either 0 (LFS) or 1 (MFS). If the MFS classifier classifies the instance as part of the LFS and the system did originally assign the MFS, the system sense with the second highest confidence is chosen. In all other cases, the sense with the highest confidence as assigned by the system is selected.

4. Results

In this Section, we study the impact of the MFS classifier on the IMS and UKB systems, respectively, with respect to the binary task of choosing between the MFS and the LFS (in Table 1) , and fine-grained WSD (in Table 2) .

Table 1 presents the results for the task of predicting for each instance in a testing corpus whether the gold sense is the MFS or among the LFS. The MFS baseline for sval2013 was 60.6% and 64% for sval2015, respectively. In addition, a True Positive (TP) is defined as correctly predicting the MFS, whereas a True Negative (TN) is defined as correctly identifying a LFS.

Table 1 presents the results for the task of correctly choosing between the MFS and the LFS. All experiments are able to beat the MFS baseline. The results from the four measures provide a clear insight into the influence of the MFS classifier on the sense assignment of the WSD systems. We observe a clear improvement of the Accuracy with an average of 3.1 points. The MFS classifier alters the sense assignment such that it is more focused on assigning the LFS, which results in higher TNrate and Precision. The cost of this can be found in the results of the Recall, which drops.

The logical next step is to use the MFS classifier in fine-grained WSD, of which the results can be found in Table 2.

		P	R	Acc	TNrate
sval2013	UKB	70.1	92.3	71.5	39.5
	UKB+C	77.7	85.4	76.3	62.3
	IMS	70.2	86.4	69.6	43.7
	IMS+C	76.9	77.4	72.3	64.3
sval2015	UKB	73.9	92.4	74.2	41.9
	UKB+C	78.0	90.3	77.5	54.7
	IMS	72.9	89.5	72.0	40.8
	IMS+C	78.2	81.5	73.6	59.5

Table 1: In this Table, the results are presented for the competitions sval2013 and sval2015, respectively, in which the WSD task has been reduced to choosing between the MFS and the LFS. Four measures are used to show the performance of the UKB and the IMS WSD systems, respectively, with (+C) and without the MFS classifier. The following measures are used: Precision (P) = TP/(TP+FP), Recall (R) = TP/(TP+FN), Accuracy (Acc) = (TP+TN)/N, and TNrate = TN/(FP+TN).

		Pwsd	Rwsd	Rwsd lfs
sval2013	UKB	65.9	65.9	25.3
	UKB+C	66.1	66.1	36.3
	IMS	60.6	60.6	20.9
	IMS+C	59.4	59.4	31.5
sval2015	UKB	68.5	67.1	20.8
	UKB+C	69.5	68.1	27.3
	IMS	67.1	65.8	17.3
	IMS+C	64.8	63.6	23.5

Table 2: In this Table, the WSD results are presented for the competitions sval2013 and sval2015, respectively. Three measures are used to show the performance of UKB and IMS WSD systems with the MFS classifier (+C) and without. Precision (Pwsd) and Recall (Rwsd) refer to the precision and recall of the official scorers of the competitions. The recall on the LFS cases (Rwsd lfs) is calculated using our own scorer.

Table 2 presents the WSD results for the competitions sval2013 and sval2015, respectively.[2] Overall, we observe that we improve the results on the LFS by an average of 8.6 points. In addition, we slightly improve the overall recall for the UKB, but decrease the recall for the IMS system. These observations are valid for both sval2013 (only nouns) as well as for sval2015, which, besides nouns, also contains verbs, adjectives, and adverbs.

5. Discussion

In this Section, we discuss our results, study the errors with respect to their sense rank, and address future work.

The main contribution of the MFS classifier is that it partly removes the MFS bias from a WSD system. This improvement is observed in a higher performance on the LFS, while

[2]The precision and recall are not the same for sval2015, because the UKB did not provide an answer for two instances of sval2015.

maintaining the overall recall. Two main issues arise from our experiments.[3] Firstly, our classifier improves the recall for the unsupervised UKB system, while it drops for the supervised IMS system. In addition, while we improve the recall on the LFS by 8.5 points on average, the performance is still far from the performance on MFS cases.

In order to get a better understanding of the two main issues, we discuss the performance of the systems per sense rank. Figure 3 presents the distribution of the gold sense ranks per competition. What stands out is the skewness of the graph, originating mainly from the fact that 80% of the gold keys are in the first three sense ranks for sval2015, whereas this percentage is even 90% for sval2013. Naturally, the main contributor to this skewness is the fact that between 55-60% of the gold keys are the MFS, i.e. sense rank 1.

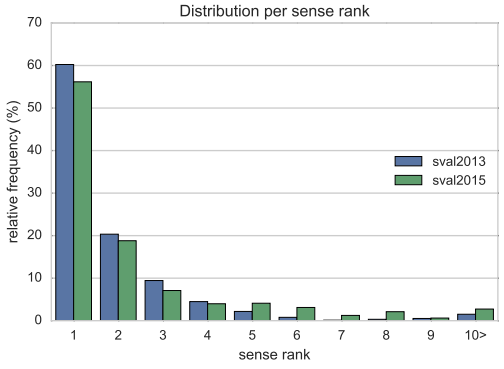

Figure 3: Sense rank distributions of gold keys from the competitions sval2013 and sval2015, respectively.

The main cause of the discrepancy between the results for IMS and UKB when we apply the MFS classifier can be found in the results for Recall and Accuracy in Table 2. For both competitions, the improvement in Precision is similar to the UKB, but the drop in Recall is bigger, resulting in a lower improvement in Accuracy. The core features of the MFS classifier are static lemma based features. It might be the case that the added value of these features are complementary to the UKB system, whereas this is to a lesser extent the case for IMS.

Figures 4 and 5 show the recall of the IMS and UKB system per sense rank, with and without the classifier. A clear pattern arises from this analysis. The performance on MFS cases drops for both systems for both competitions. However, we increase the recall mainly for the sense ranks two, three, and four, respectively. The impact of our classifier on sense ranks higher than 4 is very limited, although the recall does not decrease for those cases. Although we managed to improve the recall on LFS cases by an average of 8.5 points, there is still room for improvement into getting the LFS performance as high as the MFS performance.

In the present set-up of the MFS classifier, we only change the sense assignment of a system when the classifier determines that an instance belongs to the LFS set. We also experimented with the output from the classifier when it assigns the MFS, but this did not improve the results. It might be the case that this is related to the strategy used to alter the sense assignment, i.e. we currently opt for the sense with the second highest system confidence. For future work, one might consider incorporating sense clusters (Agirre and López de Lacalle, 2003), e.g. by selecting the sense with the highest confidence that is not in the sense cluster of the MFS. In addition, in recent years, there has been an emergence of WSD approaches that are based on word embeddings (Chen et al., 2014; Rothe and Schütze, 2015), which directly address the knowledge-acquisition bottleneck by finding new examples of usage of senses through unsupervised learning. It would be interesting to examine the impact of the MFS classifier on these systems or investigate how the acquired examples can be used to get more balanced and less skewed models.

[3] We also conducted an error analysis with respect to polysemy. However, as this analysis did not result in more insight into the workings of the MFS classifier, this is left out of the discussion of this paper.

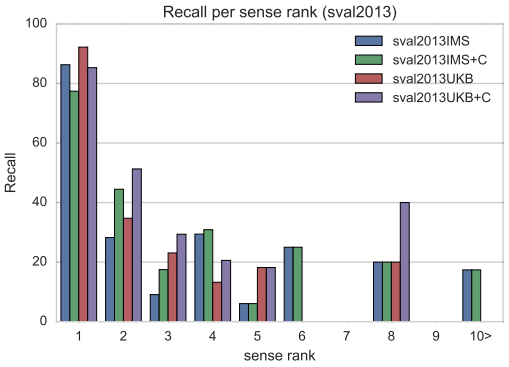

Figure 4: Recall per sense rank sval2013.

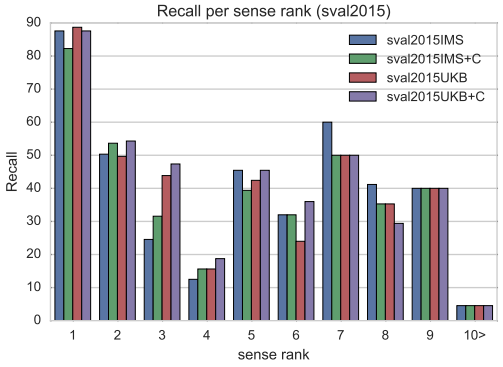

Figure 5: Recall per sense rank sval2015.

6. Conclusion

In sum, we start with the observation that WSD systems perform poorly on the LFS. Assuming that we perfectly know when to choose and not to choose the MFS (so the MFS classifier would perform with an accuracy of 100%), we observed that for example for the UKB, it would be possible to improve the overall recall by 12 points and the recall on the LFS with 32 points. In this paper, we introduced a MFS classifier, based on the Random Forest algorithm, which alters the sense assignment from a

WSD system to favor the LFS. We managed to improve the recall on the LFS by an average of 8.5 points, while maintaining the overall recall. The scripts to replicate the results reported in this paper can be found at: `https://github.com/cltl/MFS_classifier`. In future work, we will attempt to improve the MFS classifier by focusing on unsupervised approaches. Finally, several resources have been made available in the process of making this paper:

1. Wsd_corpora: sense annotated corpora in a common XML format (`https://github.com/rubenIzquierdo/wsd_corpora`)

2. WordNetMapper: this repository provides the possibility to map senses between different WordNet versions. (`https://github.com/MartenPostma/WordNetMapper`)

3. semantic_class_manager: this repository allows the users to easily access the different kinds of WordNet domains (Base Level Concepts, Domains, Supersenses). (`https://github.com/rubenIzquierdo/semantic_class_manager`)

7. Acknowledgements

This research was funded through the SPINOZA prize of Piek Vossen, awarded by the Netherlands Organisation for Scientific Research (NWO). We would also like to acknowledge Ander Barrena and Oier Lopez de Lacalle for their valuable input.

8. Bibliographical References

Agirre, E. and Edmonds, P. G. (2007). *Word sense disambiguation: Algorithms and Applications.*, volume 33. Springer Science & Business Media.

Agirre, E. and López de Lacalle, O. (2003). Clustering WordNet word senses. In *Recent Advances in Natural Language Processing (RANLP-2003)*, volume 260, pages 121–130.

Agirre, E., López de Lacalle, O., Fellbaum, C., Hsieh, S.-K., Tesconi, M., Monachini, M., Vossen, P., and Segers, R. (2010). SemEval-2010 Task 17: All-Words Word Sense Disambiguation on a Specific Domain. In *Proceedings of the 5th International Workshop on Semantic Evaluation (SemEval-2010)*, pages 75–80, Uppsala, Sweden, July. Association for Computational Linguistics.

Agirre, E., de Lacalle, O. L., and Soroa, A. (2014). Random Walks for Knowledge-Based Word Sense Disambiguation. *Computational Linguistics*, 40(1):57–84.

Breiman, L. (2001). Random Forests. *Machine learning*, 45(1):5–32.

Calvo, H. and Gelbukh, A. (2015). Is the Most Frequent Sense of a Word Better Connected in a Semantic Network? In *Advanced Intelligent Computing Theories and Applications*, pages 491–499. Springer.

Chen, X., Liu, Z., and Sun, M. (2014). A Unified Model for Word Sense Representation and Disambiguation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP-2014)*, pages 1025–1035.

Christiane Fellbaum, editor. (1998). *WordNet An Electronic Lexical Database*. The MIT Press, Cambridge, MA ; London, May.

Izquierdo, R., Suárez, A., and Rigau, G. (2007). A Proposal of Automatic Selection of Coarsegrained Semantic Classes for WSD. *Procesamiento del Lenguaje Natural*, 39:189–196.

Izquierdo, R. (2015). Error analysis of Word Sense Disambiguation. *presented at: The 25th Meeting of Computational Linguistics in the Netherlands (CLIN25)*.

Miller, G. A., Leacock, C., Tengi, R., and Bunker, R. T. (1993). A semantic concordance. In *Proceedings of the workshop on Human Language Technology*, pages 303–308. Association for Computational Linguistics.

Moro, A. and Navigli, R. (2015). SemEval-2015 task 13: Multilingual All-Words Sense Disambiguation and Entity Linking. *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval-2015)*.

Navigli, R., Jurgens, D., and Vannella, D. (2013). SemEval-2013 Task 12: Multilingual Word Sense Disambiguation. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 222–231, Atlanta, Georgia, USA, June. Association for Computational Linguistics.

Palmer, M., Fellbaum, C., Cotton, S., Delfs, L., and Dang, H. T. (2001). English Tasks: All-Words and Verb Lexical Sample. In *Proceedings of the Second International Workshop on Evaluating Word Sense Disambiguation Systems (SENSEVAL-2)*, pages 21–24, Toulouse, France, July. Association for Computational Linguistics.

Peh, L. S. and Ng, H. T. (1997). Domain-Specific Semantic Class Disambiguation using WordNet. In *Proceedings of the Fifth Workshop on Very Large Corpora*, pages 56–64.

Pradhan, S., Loper, E., Dligach, D., and Palmer, M. (2007). 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. Association for Computational Linguistics.

Preiss, J. (2006). A Detailed Comparison of WSD Systems: An Analysis of the System Answers for the SENSEVAL-2 English All Words Task. *Natural Language Engineering*, 12(3):209–228, September.

Rothe, S. and Schütze, H. (2015). Autoextend: Extending word embeddings to embeddings for synsets and lexemes. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1793–1803. Association for Computational Linguistics.

Snyder, B. and Palmer, M. (2004). The English All-Words Task. In Rada Mihalcea et al., editors, *SensEval-3: Third International Workshop on the Evaluation of Systems for*

the Semantic Analysis of Text., pages 41–43, Barcelona, Spain, July. Association for Computational Linguistics.

Steinberger, R., Ebrahim, M., and Turchi, M. (2012). JRC Eurovoc Indexer JEX - A freely available multi-label categorisation tool. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC-2012)*. European Language Resources Association (ELRA).

Zhong, Z. and Ng, T. H. (2010). It Makes Sense: A Wide-Coverage Word Sense Disambiguation System for Free Text. In *Proceedings of the ACL 2010 System Demonstrations*, pages 78–83. Association for Computational Linguistics.