Word Sense Disambiguation Using OntoNotes: An Empirical Study

Zhi Zhong and **Hwee Tou Ng** and **Yee Seng Chan**
Department of Computer Science
National University of Singapore
Law Link, Singapore 117590
{zhongzhi, nght, chanys}@comp.nus.edu.sg

Abstract

The accuracy of current word sense disambiguation (WSD) systems is affected by the fine-grained sense inventory of WordNet as well as a lack of training examples. Using the WSD examples provided through OntoNotes, we conduct the first large-scale WSD evaluation involving hundreds of word types and tens of thousands of sense-tagged examples, while adopting a coarse-grained sense inventory. We show that though WSD systems trained with a large number of examples can obtain a high level of accuracy, they nevertheless suffer a substantial drop in accuracy when applied to a different domain. To address this issue, we propose combining a domain adaptation technique using feature augmentation with active learning. Our results show that this approach is effective in reducing the annotation effort required to adapt a WSD system to a new domain. Finally, we propose that one can maximize the dual benefits of reducing the annotation effort while ensuring an increase in WSD accuracy, by only performing active learning on the set of most frequently occurring word types.

1 Introduction

In language, many words have multiple meanings. The process of identifying the correct meaning, or sense of a word in context, is known as word sense disambiguation (WSD). WSD is one of the fundamental problems in natural language processing and is important for applications such as machine translation (MT) (Chan et al., 2007a; Carpuat and Wu, 2007), information retrieval (IR), etc.

WSD is typically viewed as a classification problem where each ambiguous word is assigned a sense label (from a pre-defined sense inventory) during the disambiguation process. In current WSD research, WordNet (Miller, 1990) is usually used as the sense inventory. WordNet, however, adopts a very fine level of sense granularity, thus restricting the accuracy of WSD systems. Also, current state-of-the-art WSD systems are based on supervised learning and face a general lack of training data.

To provide a standardized test-bed for evaluation of WSD systems, a series of evaluation exercises called SENSEVAL were held. In the English all-words task of SENSEVAL-2 and SENSEVAL-3 (Palmer et al., 2001; Snyder and Palmer, 2004), no training data was provided and systems must tag all the content words (noun, verb, adjective, and adverb) in running English texts with their correct WordNet senses. In SENSEVAL-2, the best performing system (Mihalcea and Moldovan, 2001) in the English all-words task achieved an accuracy of 69.0%, while in SENSEVAL-3, the best performing system (Decadt et al., 2004) achieved an accuracy of 65.2%. In SemEval-2007, which was the most recent SENSEVAL evaluation, a similar English all-words task was held, where systems had to provide the correct WordNet sense tag for all the verbs and head words of their arguments in running English texts. For this task, the best performing system (Tratz et al., 2007) achieved an accuracy of 59.1%. Results of these evaluations showed that state-of-the-art English all-words WSD systems performed with an accuracy of 60%–70%, using the fine-grained sense inventory of WordNet.

The low level of performance by these state-of-the-art WSD systems is a cause for concern, since WSD is supposed to be an enabling technology to be incorporated as a module into applications

Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing, pages 1002–1010,
Honolulu, October 2008. ©2008 Association for Computational Linguistics

such as MT and IR. As mentioned earlier, one of the major reasons for the low performance is that these evaluation exercises adopted WordNet as the reference sense inventory, which is often too fine-grained. As an indication of this, inter-annotator agreement (ITA) reported for manual sense-tagging on these SENSEVAL English all-words datasets is typically in the mid-70s. To address this issue, a coarse-grained English all-words task (Navigli et al., 2007) was conducted during SemEval-2007. This task used a coarse-grained version of WordNet and reported an ITA of around 90%. We note that the best performing system (Chan et al., 2007b) of this task achieved a relatively high accuracy of 82.5%, highlighting the importance of having an appropriate level of sense granularity.

Another issue faced by current WSD systems is the lack of training data. We note that the top performing systems mentioned in the previous paragraphs are all based on supervised learning. With this approach, however, one would need to obtain a corpus where each ambiguous word occurrence is manually annotated with the correct sense, to serve as training data. Since it is time consuming to perform sense annotation of word occurrences, only a handful of sense-tagged corpora are publicly available. Among the existing sense-tagged corpora, the SEMCOR corpus (Miller et al., 1994) is one of the most widely used. In SEMCOR, content words have been manually tagged with WordNet senses. Current supervised WSD systems (which include all the top-performing systems in the English all-words task) usually rely on this relatively small manually annotated corpus for training examples, and this has inevitably affected the accuracy and scalability of current WSD systems.

Related to the problem of a lack of training data for WSD, there is also a lack of *test* data. Having a large amount of test data for evaluation is important to ensure the robustness and scalability of WSD systems. Due to the expensive process of manual sense-tagging, the SENSEVAL English all-words task evaluations were conducted on relatively small sets of evaluation data. For instance, the evaluation data of SENSEVAL-2 and SENSEVAL-3 English all-words task consists of 2,473 and 2,041 test examples respectively. In SemEval-2007, the fine-grained English all-words task consists of only 465 test examples, while the SemEval-2007 coarse-grained English all-words task consists of 2,269 test examples.

Hence, it is necessary to address the issues of sense granularity, and the lack of both training and test data. To this end, a recent large-scale annotation effort called the OntoNotes project (Hovy et al., 2006) was started. Building on the annotations from the Wall Street Journal (WSJ) portion of the Penn Treebank (Marcus et al., 1993), the project added several new layers of semantic annotations, such as coreference information, word senses, etc. In its first release (LDC2007T21) through the Linguistic Data Consortium (LDC), the project manually sense-tagged more than 40,000 examples belonging to hundreds of noun and verb types with an ITA of 90%, based on a coarse-grained sense inventory, where each word has an average of only 3.2 senses. Thus, besides providing WSD examples that were sense-tagged with a high ITA, the project also addressed the previously discussed issues of a lack of training and test data.

In this paper, we use the sense-tagged data provided by the OntoNotes project to investigate the accuracy achievable by current WSD systems when adopting a coarse-grained sense inventory. Through our experiments, we then highlight that domain adaptation for WSD is an important issue as it substantially affects the performance of a state-of-the-art WSD system which is trained on SEMCOR but evaluated on sense-tagged examples in OntoNotes. To address this issue, we then show that by combining a domain adaptation technique using feature augmentation with active learning, one only needs to annotate a small amount of in-domain examples to obtain a substantial improvement in the accuracy of the WSD system which is previously trained on out-of-domain examples.

The contributions of this paper are as follows. To our knowledge, this is the first large-scale WSD evaluation conducted that involves hundreds of word types and tens of thousands of sense-tagged examples, and that is based on a coarse-grained sense inventory. The present study also highlights the practical significance of domain adaptation in word sense disambiguation in the context of a large-scale empirical evaluation, and proposes an effective method to address the domain adaptation problem.

In the next section, we give a brief description of our WSD system. In Section 3, we describe experiments where we conduct both training and evaluation using data from OntoNotes. In Section 4, we investigate the WSD performance when we train our system on examples that are gathered from a different domain as compared to the OntoNotes evaluation data. In Section 5, we perform domain adaptation experiments using a recently introduced feature augmentation technique. In Section 6, we investigate the use of active learning to reduce the annotation effort required to adapt our WSD system to the domain of the OntoNotes data, before concluding in Section 7.

2 The WSD System

For the experiments reported in this paper, we follow the supervised learning approach of (Lee and Ng, 2002), by training an individual classifier for each word using the knowledge sources of local collocations, parts-of-speech (POS), and surrounding words.

For local collocations, we use 11 features: $C_{-1,-1}$, $C_{1,1}$, $C_{-2,-2}$, $C_{2,2}$, $C_{-2,-1}$, $C_{-1,1}$, $C_{1,2}$, $C_{-3,-1}$, $C_{-2,1}$, $C_{-1,2}$, and $C_{1,3}$, where $C_{i,j}$ refers to the ordered sequence of tokens in the local context of an ambiguous word *w*. Offsets *i* and *j* denote the starting and ending position (relative to *w*) of the sequence, where a negative (positive) offset refers to a token to its left (right). For parts-of-speech, we use 7 features: P_{-3}, P_{-2}, P_{-1}, P_0, P_1, P_2, P_3, where P_0 is the POS of *w*, and P_{-i} (P_i) is the POS of the *i*th token to the left (right) of *w*. For surrounding words, we consider all unigrams (single words) in the surrounding context of *w*. These words can be in a different sentence from *w*. For our experiments reported in this paper, we use support vector machines (SVM) as our learning algorithm, which was shown to achieve good WSD performance in (Lee and Ng, 2002; Chan et al., 2007b).

3 Training and Evaluating on OntoNotes

The annotated data of OntoNotes is drawn from the Wall Street Journal (WSJ) portion of the Penn Treebank corpus, divided into sections 00-24. These WSJ documents have been widely used in various NLP tasks such as syntactic parsing (Collins, 1999) and semantic role labeling (SRL) (Carreras and Marquez, 2005). In these tasks, the practice is to use documents from WSJ sections 02-21 as training data and WSJ section 23 as test data. Hence for our experiments reported in this paper, we follow this convention and use the annotated instances from WSJ sections 02-21 as our training data, and instances in WSJ section 23 as our test data.

Section	No. of word types	No. of word tokens	
		Individual	Cumulative
02	248	425	425
03	79	107	532
04	186	389	921
05	287	625	1546
06	224	446	1992
07	270	549	2541
08	177	301	2842
09	308	677	3519
10	648	3048	6567
11	724	4071	10638
12	740	4296	14934
13	749	4577	19511
14	710	3900	23411
15	748	4768	28179
16	306	576	28755
17	219	398	29153
18	266	566	29719
19	219	389	30108
20	288	536	30644
21	262	470	31114
23	685	3755	-

Table 1: Size of the sense-tagged data in the various WSJ sections.

As mentioned in Section 1, the OntoNotes data provided WSD examples for a large number of nouns and verbs, which are sense-tagged according to a coarse-grained sense inventory. In Table 1, we show the amount of sense-tagged data available from OntoNotes, across the various WSJ sections.[1] In the table, for each WSJ section, we list the number of word types, the number of sense-tagged examples, and the cumulative count on the number of

[1] We removed erroneous examples which were simply tagged with 'XXX' as sense-tag, or tagged with senses that were not found in the sense-inventory provided. Also, since we will be comparing against training on SEMCOR later (which was tagged using WordNet senses), we removed examples tagged with OntoNotes senses which were not mapped to WordNet senses. On the whole, about 7% of the original OntoNotes examples were removed as a result.

sense-tagged examples. From the table, we see that sections 02-21, which will be used as training data in our experiments, contain a total of slightly over 31,000 sense-tagged examples.

Using examples from sections 02-21 as training data, we trained our WSD system and evaluated on the examples from section 23. In our experiments, if a word type in section 23 has no training examples from sections 02-21, we randomly select an OntoNotes sense as the answer. Using these experimental settings, our WSD system achieved an accuracy of 89.1%. We note that this accuracy is much higher than the 60%–70% accuracies achieved by state-of-the-art English all-words WSD systems which are trained using the fine-grained sense inventory of WordNet. Hence, this highlights the importance of having an appropriate level of sense granularity.

Besides training on the entire set of examples from sections 02-21, we also investigated the performance achievable from training on various subsections of the data and show these results as "ON" in Figure 1. From the figure, we see that WSD accuracy increases as we add more training examples.

The fact that current state-of-the-art WSD systems are able to achieve a high level of performance is important, as this means that WSD systems will potentially be more usable for inclusion in end-applications. For instance, the high level of performance by syntactic parsers allows it to be used as an enabling technology in various NLP tasks. Here, we note that the 89.1% WSD accuracy we obtained is comparable to state-of-the-art syntactic parsing accuracies, such as the 91.0% performance by the statistical parser of Charniak and Johnson (2005).

4 Building WSD Systems with Out-of-Domain Data

Although our WSD system had achieved a high accuracy of 89.1%, this was achieved by training on a large amount (about 31,000) of manually sense annotated examples from sections 02-21 of the OntoNotes data. Further, all these training data and test data are gathered from the same domain of WSJ. In reality, however, since manual sense annotation is time consuming, it is not feasible to collect such a large amount of manually sense-tagged data for every domain of interest. Hence, in this section, we investigate the performance of our WSD system when it is trained on out-of-domain data.

In the English all-words task of the previous SENSEVAL evaluations (SENSEVAL-2, SENSEVAL-3, SemEval-2007), the best performing English all-words task systems with the highest WSD accuracy were trained on SEMCOR (Mihalcea and Moldovan, 2001; Decadt et al., 2004; Chan et al., 2007b). Hence, we similarly trained our WSD system on SEMCOR and evaluated on section 23 of the OntoNotes corpus. For those word types in section 23 which do not have training examples from SEMCOR, we randomly chose an OntoNotes sense as the answer. In training on SEMCOR, we have also ensured that there is a domain difference between our training and test data. This is because while the OntoNotes data was gathered from WSJ, which contains mainly business related news, the SEMCOR corpus is the sense-tagged portion of the Brown Corpus (BC), which is a mixture of several genres such as scientific texts, fictions, etc.

Evaluating on the section 23 test data, our WSD system achieved only 76.2% accuracy. Compared to the 89.1% accuracy achievable when we had trained on examples from sections 02-21, this is a substantially lower and disappointing drop of performance and motivates the need for domain adaptation.

The need for domain adaptation is a general and important issue for many NLP tasks (Daume III and Marcu, 2006). For instance, SRL systems are usually trained and evaluated on data drawn from the WSJ. In the CoNLL-2005 shared task on SRL (Carreras and Marquez, 2005), however, a task of training and evaluating systems on different domains was included. For that task, systems that were trained on the PropBank corpus (Palmer et al., 2005) (which was gathered from the WSJ), suffered a 10% drop in accuracy when evaluated on test data drawn from BC, as compared to the performance achievable when evaluated on data drawn from WSJ. More recently, CoNLL-2007 included a shared task on dependency parsing (Nivre et al., 2007). In this task, systems that were trained on Penn Treebank (drawn from WSJ), but evaluated on data drawn from a different domain (such as chemical abstracts and parent-child dialogues) showed a similar drop in performance. For research involving training and eval-

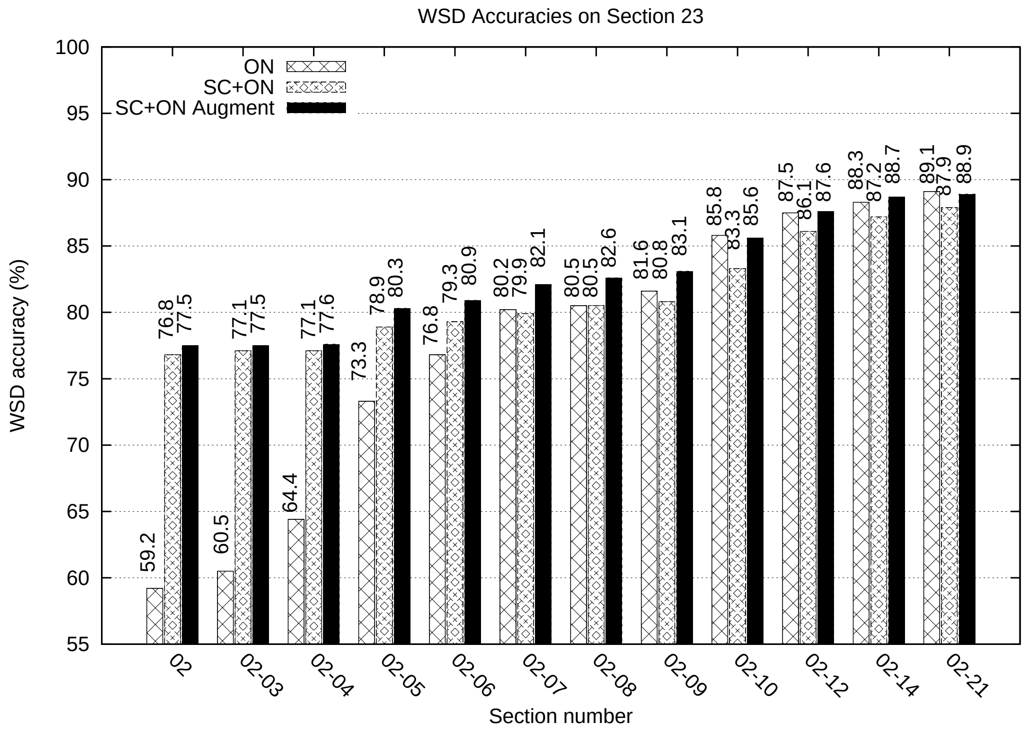

Figure 1: WSD accuracies evaluated on section 23, using SEMCOR and different OntoNotes sections as training data. ON: only OntoNotes as training data. SC+ON: SEMCOR and OntoNotes as training data, SC+ON Augment: Combining SEMCOR and OntoNotes via the Augment domain adaptation technique.

uating WSD systems on data drawn from different domains, several prior research efforts (Escudero et al., 2000; Martinez and Agirre, 2000) observed a similar drop in performance of about 10% when a WSD system that was trained on the BC part of the DSO corpus was evaluated on the WSJ part of the corpus, and vice versa.

In the rest of this paper, we perform domain adaptation experiments for WSD, focusing on domain adaptation methods that use in-domain annotated data. In particular, we use a feature augmentation technique recently introduced by Daume III (2007), and active learning (Lewis and Gale, 1994) to perform domain adaptation of WSD systems.

5 Combining In-Domain and Out-of-Domain Data for Training

In this section, we will first introduce the AUGMENT technique of Daume III (2007), before showing the performance of our WSD system with and without using this technique.

5.1 The AUGMENT technique for Domain Adaptation

The AUGMENT technique introduced by Daume III (2007) is a simple yet very effective approach to performing domain adaptation. This technique is applicable when one has access to training data from the source domain and a small amount of training data from the target domain.

The technique essentially augments the feature space of an instance. Assuming $\boldsymbol{x}$ is an instance and its original feature vector is $\Phi(\boldsymbol{x})$, the augmented feature vector for instance $\boldsymbol{x}$ is

$$\Phi'(\boldsymbol{x}) = \begin{cases} < \Phi(\boldsymbol{x}), \Phi(\boldsymbol{x}), \mathbf{0} > & \text{if } \boldsymbol{x} \in D_s \\ < \Phi(\boldsymbol{x}), \mathbf{0}, \Phi(\boldsymbol{x}) > & \text{if } \boldsymbol{x} \in D_t \end{cases},$$

where $\mathbf{0}$ is a zero vector of size $|\Phi(\boldsymbol{x})|$, D_s and D_t are the sets of instances from the source and target domains respectively. We see that the technique essentially treats the first part of the augmented feature space as holding general features that are not meant to be differentiated between different domains. Then, different parts of the augmented feature space are reserved for holding source domain specific, or target domain specific features. Despite its relative simplicity, this AUGMENT technique has been shown to outperform other domain adaptation techniques on various tasks such as named entity recognition, part-of-speech tagging, etc.

5.2 Experimental Results

As mentioned in Section 4, training our WSD system on SEMCOR examples gave a relatively low accuracy of 76.2%, as compared to the 89.1% accuracy obtained from training on the OntoNotes section 02-21 examples. Assuming we have access to some in-domain training data, then a simple method to potentially obtain better accuracies is to train on both the out-of-domain and in-domain examples. To investigate this, we combined the SEMCOR examples with various amounts of OntoNotes examples to train our WSD system and show the resulting "SC+ON" accuracies obtained in Figure 1. We also performed another set of experiments, where instead of simply combining the SEMCOR and OntoNotes examples, we applied the AUGMENT technique when combining these examples, treating SEMCOR examples as out-of-domain (source domain) data and OntoNotes examples as in-domain (target domain) data. We similarly show the resulting accuracies as "SC+ON Augment" in Figure 1.

Comparing the "SC+ON" and "SC+ON Augment" accuracies in Figure 1, we see that the AUGMENT technique *always* helps to improve the accuracy of our WSD system. Further, notice from the first few sets of results in the figure that when we have access to limited in-domain training examples from OntoNotes, incorporating additional out-of-domain training data from SEMCOR (either using the strategies "SC+ON" or "SC+ON Augment") achieves better accuracies than "ON". Significance tests using one-tailed paired t-test reveal that these accuracy improvements are statistically significant at the level of significance 0.01 (all significance tests in the rest of this paper use the same level of significance 0.01). These results validate the contribution of the SemCor examples. This trend continues till the result for sections 02-06.

The right half of Figure 1 shows the accuracy trend of the various strategies, in the unlikely event that we have access to a large amount of in-domain training examples. Although we observe that in this scenario, "ON" performs better than "SC+ON", "SC+ON Augment" continues to perform better than "ON" (where the improvement is statistically significant) till the result for sections 02-09. Beyond that, as we add more OntoNotes examples, significance testing reveals that the "SC+ON Augment" and "ON" strategies give comparable performance. This means that the "SC+ON Augment" strategy, besides giving good performance when one has few in-domain examples, does continue to perform well even when one has a large number of in-domain examples.

```
$\mathrm{D}_S \leftarrow$ the set of SEMCOR training examples
$\mathrm{D}_A \leftarrow$ the set of OntoNotes sections 02-21 examples
$\mathrm{D}_T \leftarrow$ empty
while $\mathrm{D}_A \neq \phi$
    $\mathrm{p}_{min} \leftarrow \infty$
    $\Gamma \leftarrow$ WSD system trained on $\mathrm{D}_S$ and $\mathrm{D}_T$ using AUGMENT technique
    for each $d \in \mathrm{D}_A$ do
        $\widehat{s} \leftarrow$ word sense prediction for $d$ using $\Gamma$
        $p \leftarrow$ confidence of prediction $\widehat{s}$
        if $p < \mathrm{p}_{min}$ then
            $\mathrm{p}_{min} \leftarrow p$, $\mathrm{d}_{min} \leftarrow d$
        end
    end
    $\mathrm{D}_A \leftarrow \mathrm{D}_A - \{\mathrm{d}_{min}\}$
    provide correct sense $s$ for $\mathrm{d}_{min}$ and add $\mathrm{d}_{min}$ to $\mathrm{D}_T$
end
```

Figure 2: The active learning algorithm.

6 Active Learning with AUGMENT Technique

So far in this paper, we have seen that when we have access to some in-domain examples, a good strategy is to combine the out-of-domain and in-domain examples via the AUGMENT technique. This suggests that when one wishes to apply a WSD system to a new domain of interest, it is worth the effort to annotate a small number of examples gathered from the new domain. However, instead of randomly selecting in-domain examples to annotate, we could use active learning (Lewis and Gale, 1994) to help select in-domain examples to annotate. By doing so, we could minimize the manual annotation effort needed.

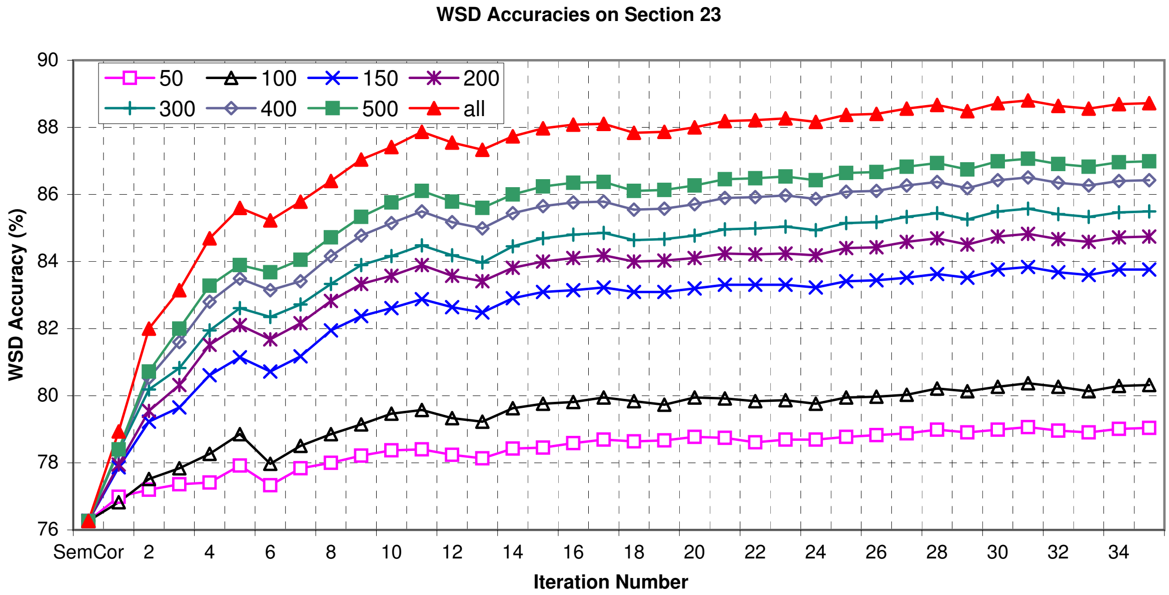

Figure 3: Results of applying active learning with the AUGMENT technique on different number of word types. Each curve represents the adaptation process of applying active learning on a certain number of most frequently occurring word types.

In WSD, several prior research efforts have successfully used active learning to reduce the annotation effort required (Zhu and Hovy, 2007; Chan and Ng, 2007; Chen et al., 2006; Fujii et al., 1998). With the exception of (Chan and Ng, 2007) which tried to adapt a WSD system trained on the BC part of the DSO corpus to the WSJ part of the DSO corpus, the other researchers simply applied active learning to reduce the annotation effort required and did not deal with the issue of adapting a WSD system to a new domain. Also, these prior research efforts only experimented with a few word types. In contrast, we perform active learning experiments on the hundreds of word types in the OntoNotes data, with the aim of adapting our WSD system trained on SEMCOR to the WSJ domain represented by the OntoNotes data.

For our active learning experiments, we use the *uncertainty sampling* strategy (Lewis and Gale, 1994), as shown in Figure 2. For our experiments, the SEMCOR examples will be our initial set of training examples, while the OntoNotes examples from sections 02-21 will be used as our pool of adaptation examples, from which we will select examples to annotate via active learning. Also, since we have found that the AUGMENT technique is useful in increasing WSD accuracy, we will apply the AUGMENT technique during each iteration of active learning to combine the SEMCOR examples and the selected adaptation examples.

As shown in Figure 2, we train an initial WSD system using only the set D_S of SEMCOR examples. We then apply our WSD system on the set D_A of OntoNotes adaptation examples. The example in D_A which is predicted with the lowest confidence will be removed from D_A and added to the set D_T of in-domain examples that have been selected via active learning thus far. We then use the AUGMENT technique to combine the set of examples in D_S and D_T to train a new WSD system, which is then applied again on the set D_A of remaining adaptation examples, and this active learning process continues until we have used up all the adaptation examples. Note that because we are using OntoNotes sections 02-21 (which have already been sense-tagged beforehand) as our adaptation data, the annotation of the selected example during each active learning iteration is simply simulated by referring to its tagged sense.

6.1 Experimental Results

As mentioned earlier, we use the examples in OntoNotes sections 02-21 as our adaptation examples during active learning. Hence, we perform active learning experiments on *all* the word types that have sense-tagged examples from OntoNotes sections 02-21, and show the evaluation results on OntoNotes section 23 as the topmost "all" curve in Figure 3. Since our aim is to reduce the human annotation effort required in adapting a WSD system to a new domain, we may not want to perform active learning on all the word types in practice. Instead, we can maximize the benefits by performing active learning only on the more frequently occurring word types. Hence, in Figure 3, we also show via various curves the results of applying active learning only to various sets of word types, according to their frequency, or number of sense-tagged examples in OntoNotes sections 02-21. Note that the various accuracy curves in Figure 3 are plotted in terms of evaluation accuracies over all the test examples in OntoNotes section 23, hence they are directly comparable to the results reported thus far in this paper. Also, since the accuracies for the various curves stabilize after 35 active learning iterations, we only show the results of the first 35 iterations.

From Figure 3, we note that by performing active learning on the set of 150 most frequently occurring word types, we are able to achieve a WSD accuracy of 82.6% after 10 active learning iterations. Note that in Section 4, we mentioned that training only on the out-of-domain SEMCOR examples gave an accuracy of 76.2%. Hence, we have gained an accuracy improvement of 6.4% (82.6% − 76.2%) by just using 1,500 in-domain OntoNotes examples. Compared with the 12.9% (89.1% − 76.2%) improvement in accuracy achieved by using all 31,114 OntoNotes sections 02-21 examples, we have obtained half of this maximum increase in accuracy, by requiring only about 5% (1,500/31,114) of the total number of sense-tagged examples. Based on these results, we propose that when there is a need to apply a previously trained WSD system to a different domain, one can apply the AUGMENT technique with active learning on the most frequent word types, to greatly reduce the annotation effort required while obtaining a substantial improvement in accuracy.

7 Conclusion

Using the WSD examples made available through OntoNotes, which are sense-tagged according to a coarse-grained sense inventory, we show that our WSD system is able to achieve a high accuracy of 89.1% when we train and evaluate on these examples. However, when we apply a WSD system that is trained on SEMCOR, we suffer a substantial drop in accuracy, highlighting the need to perform domain adaptation. We show that by combining the AUGMENT domain adaptation technique with active learning, we are able to effectively reduce the amount of annotation effort required for domain adaptation.

References

M. Carpuat and D. Wu. 2007. Improving Statistical Machine Translation Using Word Sense Disambiguation. In *Proc. of EMNLP-CoNLL07*, pages 61–72.

X. Carreras and L. Marquez. 2005. Introduction to the CoNLL-2005 Shared Task: Semantic Role Labeling. In *Proc. of CoNLL-2005*, pages 152–164.

Y. S. Chan and H. T. Ng. 2007. Domain Adaptation with Active Learning for Word Sense Disambiguation. In *Proc. of ACL07*, pages 49–56.

Y. S. Chan, H. T. Ng, and D. Chiang. 2007a. Word Sense Disambiguation Improves Statistical Machine Translation. In *Proc. of ACL07*, pages 33–40.

Y. S. Chan, H. T. Ng, and Z. Zhong. 2007b. NUS-PT: Exploiting Parallel Texts for Word Sense Disambiguation in the English All-Words Tasks. In *Proc. of SemEval-2007*, pages 253–256.

E. Charniak and M. Johnson. 2005. Coarse-to-Fine n-Best Parsing and MaxEnt Discriminative Reranking. In *Proc. of ACL05*, pages 173–180.

J. Y. Chen, A. Schein, L. Ungar, and M. Palmer. 2006. An Empirical Study of the Behavior of Active Learning for Word Sense Disambiguation. In *Proc. of HLT/NAACL06*, pages 120–127.

M. Collins. 1999. *Head-Driven Statistical Model for Natural Language Parsing*. PhD dissertation, University of Pennsylvania.

H. Daume III and D. Marcu. 2006. Domain Adaptation for Statistical Classifiers. *Journal of Artificial Intelligence Research*, 26:101–126.

H. Daume III. 2007. Frustratingly Easy Domain Adaptation. In *Proc. of ACL07*, pages 256–263.

B. Decadt, V. Hoste, and W. Daelemans. 2004. GAMBL, Genetic Algorithm Optimization of Memory-Based WSD. In *Proc. of SENSEVAL-3*, pages 108–112.

G. Escudero, L. Marquez, and G. Riagu. 2000. An Empirical Study of the Domain Dependence of Supervised Word Sense Disambiguation Systems. In *Proc. of EMNLP/VLC00*, pages 172–180.

A. Fujii, K. Inui, T. Tokunaga, and H. Tanaka. 1998. Selective Sampling for Example-based Word Sense Disambiguation. *Computational Linguistics*, 24(4).

E. Hovy, M. Marcus, M. Palmer, L. Ramshaw, and R. Weischedel. 2006. OntoNotes: The 90% solution. In *Proc. of HLT-NAACL06*, pages 57–60.

Y. K. Lee and H. T. Ng. 2002. An Empirical Evaluation of Knowledge Sources and Learning Algorithms for Word Sense Disambiguation. In *Proc. of EMNLP02*, pages 41–48.

D. D. Lewis and W. A. Gale. 1994. A Sequential Algorithm for Training Text Classifiers. In *Proc. of SIGIR94*.

M. Marcus, B. Santorini, and M. A. Marcinkiewicz. 1993. Building a Large Annotated Corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2):313–330.

D. Martinez and E. Agirre. 2000. One Sense per Collocation and Genre/Topic Variations. In *Proc. of EMNLP/VLC00*, pages 207–215.

R. Mihalcea and D. Moldovan. 2001. Pattern Learning and Active Feature Selection for Word Sense Disambiguation. In *Proc. of SENSEVAL-2*, pages 127–130.

G. A. Miller, M. Chodorow, S. Landes, C. Leacock, and R. G. Thomas. 1994. Using a Semantic Concordance for Sense Identification. In *Proc. of ARPA Human Language Technology Workshop*, pages 240–243.

G. A. Miller. 1990. WordNet: An On-line Lexical Database. *International Journal of Lexicography*, 3(4):235–312.

R. Navigli, K. C. Litkowski, and O. Hargraves. 2007. SemEval-2007 Task 07: Coarse-Grained English All-Words Task. In *Proc. of SemEval-2007*, pages 30–35.

J. Nivre, J. Hall, S. Kubler, R. McDonald, J. Nilsson, S. Riedel, and D. Yuret. 2007. The CoNLL 2007 Shared Task on Dependency Parsing. In *Proc. of EMNLP-CoNLL07*, pages 915–932.

M. Palmer, C. Fellbaum, S. Cotton, L. Delfs, and H. T. Dang. 2001. English Tasks: All-Words and Verb Lexical Sample. In *Proc. of SENSEVAL-2*, pages 21–24.

M. Palmer, D. Gildea, and P. Kingsbury. 2005. The Proposition Bank: An Annotated Corpus of Semantic Roles. *Computational Linguistics*, 31(1):71–105.

B. Snyder and M. Palmer. 2004. The English All-Words Task. In *Proc. of SENSEVAL-3*, pages 41–43.

S. Tratz, A. Sanfilippo, M. Gregory, A. Chappell, C. Posse, and P. Whitney. 2007. PNNL: A Supervised Maximum Entropy Approach to Word Sense Disambiguation. In *Proc. of SemEval-2007*, pages 264–267.

J. B. Zhu and E. Hovy. 2007. Active Learning for Word Sense Disambiguation with Methods for Addressing the Class Imbalance Problem. In *Proc. of EMNLP-CoNLL07*, pages 783–790.