Domain Adaptation with Active Learning for Word Sense Disambiguation

Yee Seng Chan and **Hwee Tou Ng**
Department of Computer Science
National University of Singapore
3 Science Drive 2, Singapore 117543
{chanys, nght}@comp.nus.edu.sg

Abstract

When a word sense disambiguation (WSD) system is trained on one domain but applied to a different domain, a drop in accuracy is frequently observed. This highlights the importance of domain adaptation for word sense disambiguation. In this paper, we first show that an active learning approach can be successfully used to perform domain adaptation of WSD systems. Then, by using the predominant sense predicted by expectation-maximization (EM) and adopting a count-merging technique, we improve the effectiveness of the original adaptation process achieved by the basic active learning approach.

1 Introduction

In natural language, a word often assumes different meanings, and the task of determining the correct meaning, or sense, of a word in different contexts is known as word sense disambiguation (WSD). To date, the best performing systems in WSD use a corpus-based, supervised learning approach. With this approach, one would need to collect a text corpus, in which each ambiguous word occurrence is first tagged with its correct sense to serve as training data.

The reliance of supervised WSD systems on annotated corpus raises the important issue of domain dependence. To investigate this, Escudero et al. (2000) and Martinez and Agirre (2000) conducted experiments using the DSO corpus, which contains sentences from two different corpora, namely Brown Corpus (BC) and Wall Street Journal (WSJ). They found that training a WSD system on one part (BC or WSJ) of the DSO corpus, and applying it to the other, can result in an accuracy drop of more than 10%, highlighting the need to perform domain adaptation of WSD systems to new domains. Escudero et al. (2000) pointed out that one of the reasons for the drop in accuracy is the difference in sense priors (i.e., the proportions of the different senses of a word) between BC and WSJ. When the authors assumed they knew the sense priors of each word in BC and WSJ, and adjusted these two datasets such that the proportions of the different senses of each word were the same between BC and WSJ, accuracy improved by 9%.

In this paper, we explore domain adaptation of WSD systems, by adding training examples from the new domain as additional training data to a WSD system. To reduce the effort required to adapt a WSD system to a new domain, we employ an active learning strategy (Lewis and Gale, 1994) to select examples to annotate from the new domain of interest. To our knowledge, our work is the first to use active learning for domain adaptation for WSD. A similar work is the recent research by Chen et al. (2006), where active learning was used successfully to reduce the annotation effort for WSD of 5 English verbs using *coarse-grained* evaluation. In that work, the authors only used active learning to reduce the annotation effort and did not deal with the porting of a WSD system to a new domain.

Domain adaptation is necessary when the training and target domains are different. In this paper,

Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics, pages 49–56,
Prague, Czech Republic, June 2007.

we perform domain adaptation for WSD of a set of nouns using *fine-grained* evaluation. The contribution of our work is not only in showing that active learning can be successfully employed to reduce the annotation effort required for domain adaptation in a *fine-grained* WSD setting. More importantly, our main focus and contribution is in showing how we can improve the effectiveness of a basic active learning approach when it is used for domain adaptation. In particular, we explore the issue of different sense priors across different domains. Using the sense priors estimated by expectation-maximization (EM), the predominant sense in the new domain is predicted. Using this predicted predominant sense and adopting a count-merging technique, we *improve* the effectiveness of the adaptation process.

In the next section, we discuss the choice of corpus and nouns used in our experiments. We then introduce active learning for domain adaptation, followed by count-merging. Next, we describe an EM-based algorithm to estimate the sense priors in the new domain. Performance of domain adaptation using active learning and count-merging is then presented. Next, we show that by using the predominant sense of the target domain as predicted by the EM-based algorithm, we improve the effectiveness of the adaptation process. Our empirical results show that for the set of nouns which have different predominant senses between the training and target domains, we are able to reduce the annotation effort by 71%.

2 Experimental Setting

In this section, we discuss the motivations for choosing the particular corpus and the set of nouns to conduct our domain adaptation experiments.

2.1 Choice of Corpus

The DSO corpus (Ng and Lee, 1996) contains 192,800 annotated examples for 121 nouns and 70 verbs, drawn from BC and WSJ. While the BC is built as a balanced corpus, containing texts in various categories such as religion, politics, humanities, fiction, etc, the WSJ corpus consists primarily of business and financial news. Exploiting the difference in coverage between these two corpora, Escudero et al. (2000) separated the DSO corpus into its BC and WSJ parts to investigate the domain dependence of several WSD algorithms. Following the setup of (Escudero et al., 2000), we similarly made use of the DSO corpus to perform our experiments on domain adaptation.

Among the few currently available manually sense-annotated corpora for WSD, the SEMCOR (SC) corpus (Miller et al., 1994) is the most widely used. SEMCOR is a subset of BC which is sense-annotated. Since BC is a balanced corpus, and since performing adaptation from a general corpus to a more specific corpus is a natural scenario, we focus on adapting a WSD system trained on BC to WSJ in this paper. Henceforth, out-of-domain data will refer to BC examples, and in-domain data will refer to WSJ examples.

2.2 Choice of Nouns

The WordNet Domains resource (Magnini and Cavaglia, 2000) assigns domain labels to synsets in WordNet. Since the focus of the WSJ corpus is on business and financial news, we can make use of WordNet Domains to select the set of nouns having at least one synset labeled with a business or finance related domain label. This is similar to the approach taken in (Koeling et al., 2005) where they focus on determining the predominant sense of words in corpora drawn from finance versus sports domains.[1] Hence, we select the subset of DSO nouns that have at least one synset labeled with any of these domain labels: *commerce*, *enterprise*, *money*, *finance*, *banking*, and *economy*. This gives a set of 21 nouns: *book*, *business*, *center*, *community*, *condition*, *field*, *figure*, *house*, *interest*, *land*, *line*, *money*, *need*, *number*, *order*, *part*, *power*, *society*, *term*, *use*, *value*.[2]

For each noun, all the BC examples are used as out-of-domain training data. One-third of the WSJ examples for each noun are set aside as evaluation

[1] Note however that the coverage of the WordNet Domains resource is not comprehensive, as about 31% of the synsets are simply labeled with "factotum", indicating that the synset does not belong to a specific domain.

[2] 25 nouns have at least one synset labeled with the listed domain labels. In our experiments, 4 out of these 25 nouns have an accuracy of more than 90% before adaptation (i.e., training on just the BC examples) and accuracy improvement is less than 1% after all the available WSJ adaptation examples are added as additional training data. To obtain a clearer picture of the adaptation process, we discard these 4 nouns, leaving a set of 21 nouns.

Dataset	No. of senses		MFS acc. (%)	No. of training examples	No. of adaptation examples
	BC	WSJ			
21 nouns	6.7	6.8	61.1	310	406
9 nouns	7.9	8.6	65.8	276	416

Table 1: The average number of senses in BC and WSJ, average MFS accuracy, average number of BC training, and WSJ adaptation examples per noun.

data, and the rest of the WSJ examples are designated as in-domain adaptation data. The row *21 nouns* in Table 1 shows some information about these 21 nouns. For instance, these nouns have an average of 6.7 senses in BC and 6.8 senses in WSJ. This is slightly higher than the 5.8 senses per verb in (Chen et al., 2006), where the experiments were conducted using coarse-grained evaluation. Assuming we have access to an "oracle" which determines the predominant sense, or most frequent sense (MFS), of each noun in our WSJ test data perfectly, and we assign this most frequent sense to each noun in the test data, we will have achieved an accuracy of 61.1% as shown in the column *MFS accuracy* of Table 1. Finally, we note that we have an average of 310 BC training examples and 406 WSJ adaptation examples per noun.

3 Active Learning

For our experiments, we use naive Bayes as the learning algorithm. The knowledge sources we use include parts-of-speech, local collocations, and surrounding words. These knowledge sources were effectively used to build a state-of-the-art WSD program in one of our prior work (Lee and Ng, 2002). In performing WSD with a naive Bayes classifier, the sense s assigned to an example with features $f_1, \ldots, f_n$ is chosen so as to maximize:

$$p(s) \prod_{j=1}^{n} p(f_j|s)$$

In our domain adaptation study, we start with a WSD system built using training examples drawn from BC. We then investigate the utility of adding additional in-domain training data from WSJ. In the baseline approach, the additional WSJ examples are randomly selected. With active learning (Lewis and Gale, 1994), we use *uncertainty sampling* as shown in Figure 1. In each iteration, we train a WSD system on the available training data and apply it on the WSJ adaptation examples. Among these WSJ examples, the example predicted with the lowest confidence is selected and removed from the adaptation data. The correct label is then supplied for this example and it is added to the training data.

```
D_T ← the set of BC training examples
D_A ← the set of untagged WSJ adaptation examples
Γ ← WSD system trained on D_T
repeat
    p_min ← ∞
    for each d ∈ D_A do
        ŝ ← word sense prediction for d using Γ
        p ← confidence of prediction ŝ
        if p < p_min then
            p_min ← p, d_min ← d
        end
    end
    D_A ← D_A − d_min
    provide correct sense s for d_min and add d_min to D_T
    Γ ← WSD system trained on new D_T
end
```

Figure 1: Active learning

Note that in the experiments reported in this paper, all the adaptation examples are already pre-annotated before the experiments start, since all the WSJ adaptation examples come from the DSO corpus which have already been sense-annotated. Hence, the annotation of an example needed during each adaptation iteration is simulated by performing a lookup without any manual annotation.

4 Count-merging

We also employ a technique known as *count-merging* in our domain adaptation study. Count-merging assigns different weights to different examples to better reflect their relative importance. Roark and Bacchiani (2003) showed that weighted count-merging is a special case of maximum a posteriori (MAP) estimation, and successfully used it for probabilistic context-free grammar domain adaptation (Roark and Bacchiani, 2003) and language model adaptation (Bacchiani and Roark, 2003).

Count-merging can be regarded as scaling of counts obtained from different data sets. We let $\widetilde{c}$ denote the counts from out-of-domain training data, $\bar{c}$ denote the counts from in-domain adaptation data, and $\widehat{p}$ denote the probability estimate by count-merging. We can scale the out-of-domain and in-domain counts with different factors, or just use a single weight parameter β:

$$\widehat{p}(f_j|s_i) = \frac{\widetilde{c}(f_j, s_i) + \beta\bar{c}(f_j, s_i)}{\widetilde{c}(s_i) + \beta\bar{c}(s_i)} \tag{1}$$

Similarly,

$$\widehat{p}(s_i) = \frac{\widetilde{c}(s_i) + \beta\bar{c}(s_i)}{\widetilde{c} + \beta\bar{c}} \tag{2}$$

Obtaining an optimum value for β is not the focus of this work. Instead, we are interested to see if assigning a higher weight to the in-domain WSJ adaptation examples, as compared to the out-of-domain BC examples, will improve the adaptation process. Hence, we just use a β value of 3 in our experiments involving count-merging.

5 Estimating Sense Priors

In this section, we describe an EM-based algorithm that was introduced by Saerens et al. (2002), which can be used to estimate the sense priors, or a priori probabilities of the different senses in a new dataset. We have recently shown that this algorithm is effective in estimating the sense priors of a set of nouns (Chan and Ng, 2005).

Most of this section is based on (Saerens et al., 2002). Assume we have a set of labeled data D_L with *n* classes and a set of *N* independent instances $(\mathbf{x}_1, \ldots, \mathbf{x}_N)$ from a new data set. The likelihood of these *N* instances can be defined as:

$$\begin{aligned} L(\mathbf{x}_1, \ldots, \mathbf{x}_N) &= \prod_{k=1}^{N} p(\mathbf{x}_k) \\ &= \prod_{k=1}^{N} \left[\sum_{i=1}^{n} p(\mathbf{x}_k, \omega_i) \right] \\ &= \prod_{k=1}^{N} \left[\sum_{i=1}^{n} p(\mathbf{x}_k|\omega_i) p(\omega_i) \right] \end{aligned} \tag{3}$$

Assuming the within-class densities $p(\mathbf{x}_k|\omega_i)$, i.e., the probabilities of observing $\mathbf{x}_k$ given the class ω_i, do not change from the training set D_L to the new data set, we can define: $p(\mathbf{x}_k|\omega_i) = p_L(\mathbf{x}_k|\omega_i)$. To determine the a priori probability estimates $\widehat{p}(\omega_i)$ of the new data set that will maximize the likelihood of (3) with respect to $p(\omega_i)$, we can apply the iterative procedure of the EM algorithm. In effect, through maximizing the likelihood of (3), we obtain the a priori probability estimates as a by-product.

Let us now define some notations. When we apply a classifier trained on D_L on an instance $\mathbf{x}_k$ drawn from the new data set D_U, we get $\widehat{p}_L(\omega_i|\mathbf{x}_k)$, which we define as the probability of instance $\mathbf{x}_k$ being classified as class ω_i by the classifier trained on D_L. Further, let us define $\widehat{p}_L(\omega_i)$ as the a priori probability of class ω_i in D_L. This can be estimated by the class frequency of ω_i in D_L. We also define $\widehat{p}^{(s)}(\omega_i)$ and $\widehat{p}^{(s)}(\omega_i|\mathbf{x}_k)$ as estimates of the new a priori and a posteriori probabilities at step *s* of the iterative EM procedure. Assuming we initialize $\widehat{p}^{(0)}(\omega_i) = \widehat{p}_L(\omega_i)$, then for each instance $\mathbf{x}_k$ in D_U and each class ω_i, the EM algorithm provides the following iterative steps:

$$\widehat{p}^{(s)}(\omega_i|\mathbf{x}_k) = \frac{\widehat{p}_L(\omega_i|\mathbf{x}_k)\frac{\widehat{p}^{(s)}(\omega_i)}{\widehat{p}_L(\omega_i)}}{\sum_{j=1}^{n} \widehat{p}_L(\omega_j|\mathbf{x}_k)\frac{\widehat{p}^{(s)}(\omega_j)}{\widehat{p}_L(\omega_j)}} \tag{4}$$

$$\widehat{p}^{(s+1)}(\omega_i) = \frac{1}{N}\sum_{k=1}^{N} \widehat{p}^{(s)}(\omega_i|\mathbf{x}_k) \tag{5}$$

where Equation (4) represents the expectation E-step, Equation (5) represents the maximization M-step, and *N* represents the number of instances in D_U. Note that the probabilities $\widehat{p}_L(\omega_i|\mathbf{x}_k)$ and $\widehat{p}_L(\omega_i)$ in Equation (4) will stay the same throughout the iterations for each particular instance $\mathbf{x}_k$ and class ω_i. The new a posteriori probabilities $\widehat{p}^{(s)}(\omega_i|\mathbf{x}_k)$ at step *s* in Equation (4) are simply the a posteriori probabilities in the conditions of the labeled data, $\widehat{p}_L(\omega_i|\mathbf{x}_k)$, weighted by the ratio of the new priors $\widehat{p}^{(s)}(\omega_i)$ to the old priors $\widehat{p}_L(\omega_i)$. The denominator in Equation (4) is simply a normalizing factor.

The a posteriori $\widehat{p}^{(s)}(\omega_i|\mathbf{x}_k)$ and a priori probabilities $\widehat{p}^{(s)}(\omega_i)$ are re-estimated sequentially during each iteration *s* for each new instance $\mathbf{x}_k$ and each class ω_i, until the convergence of the estimated probabilities $\widehat{p}^{(s)}(\omega_i)$, which will be our estimated sense priors. This iterative procedure will increase the likelihood of (3) at each step.

6 Experimental Results

For each adaptation experiment, we start off with a classifier built from an initial training set consisting

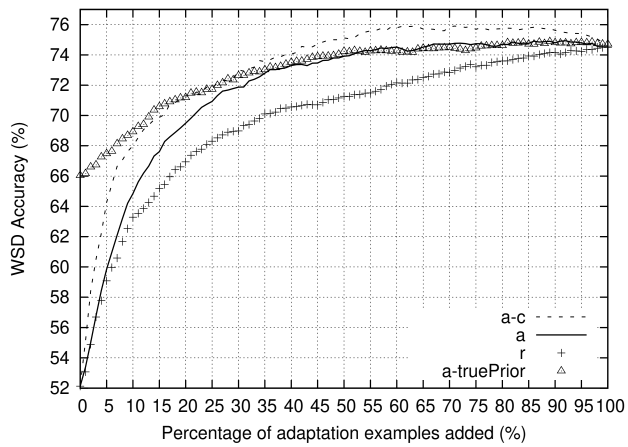

Figure 2: Adaptation process for all 21 nouns.

of the BC training examples. At each adaptation iteration, WSJ adaptation examples are selected *one at a time* and added to the training set. The adaptation process continues until all the adaptation examples are added. Classification accuracies averaged over 3 random trials on the WSJ test examples at each iteration are calculated. Since the number of WSJ adaptation examples differs for each of the 21 nouns, the learning curves we will show in the various figures are plotted in terms of different percentage of adaptation examples added, varying from 0 to 100 percent in steps of 1 percent. To obtain these curves, we first calculate for each noun, the WSD accuracy when different percentages of adaptation examples are added. Then, for each percentage, we calculate the macro-average WSD accuracy over all the nouns to obtain a single learning curve representing all the nouns.

6.1 Utility of Active Learning and Count-merging

In Figure 2, the curve *r* represents the adaptation process of the baseline approach, where additional WSJ examples are randomly selected during each adaptation iteration. The adaptation process using active learning is represented by the curve *a*, while applying count-merging with active learning is represented by the curve *a-c*. Note that random selection *r* achieves its highest WSD accuracy after *all* the adaptation examples are added. To reach the same accuracy, the *a* approach requires the addition of only 57% of adaptation examples. The *a-c* approach is even more effective and requires only 42% of adaptation examples. This demonstrates the effectiveness of count-merging in further reducing the annotation effort, when compared to using only active learning. To reach the MFS accuracy of 61.1% as shown earlier in Table 1, *a-c* requires just 4% of the adaptation examples.

To determine the utility of the out-of-domain BC examples, we have also conducted three active learning runs using only WSJ adaptation examples. Using 10%, 20%, and 30% of WSJ adaptation examples to build a classifier, the accuracy of these runs is lower than the active learning *a* curve and paired t-tests show that the difference is statistically significant at the level of significance 0.01.

6.2 Using Sense Priors Information

As mentioned in section 1, research in (Escudero et al., 2000) noted an improvement in accuracy when they adjusted the BC and WSJ datasets such that the proportions of the different senses of each word were the same between BC and WSJ. We can similarly choose BC examples such that the sense priors in the BC training data adhere to the sense priors in the WSJ evaluation data. To gauge the effectiveness of this approach, we first assume that we know the *true* sense priors of each noun in the WSJ evaluation data. We then gather BC training examples for a noun to adhere as much as possible to the sense priors in WSJ. Assume sense s_i is the predominant sense in the WSJ evaluation data, s_i has a sense prior of p_i in the WSJ data and has n_i BC training examples. Taking n_i examples to represent a sense prior of p_i, we proportionally determine the number of BC examples to gather for other senses s according to their respective sense priors in WSJ. If there are insufficient training examples in BC for some sense s, whatever available examples of s are used.

This approach gives an average of 195 BC training examples for the 21 nouns. With this new set of training examples, we perform adaptation using active learning and obtain the *a-truePrior* curve in Figure 2. The *a-truePrior* curve shows that by ensuring that the sense priors in the BC training data adhere as much as possible to the sense priors in the WSJ data, we start off with a higher WSD accuracy. However, the performance is no different from the *a* curve after 35% of adaptation examples are added. A possible reason might be that by strictly adhering to the sense priors in the WSJ data, we have removed too many BC training examples, from an average of 310 examples per noun as shown in Table 1, to an average of 195 examples.

6.3 Using Predominant Sense Information

Research by McCarthy et al. (2004) and Koeling et al. (2005) pointed out that a change of predominant sense is often indicative of a change in domain. For example, the predominant sense of the noun *interest* in the BC part of the DSO corpus has the meaning "a sense of concern with and curiosity about someone or something". In the WSJ part of the DSO corpus, the noun *interest* has a different predominant sense with the meaning "a fixed charge for borrowing money", which is reflective of the business and finance focus of the WSJ corpus.

Instead of restricting the BC training data to adhere strictly to the sense priors in WSJ, another alternative is just to ensure that the predominant sense in BC is the same as that of WSJ. Out of the 21 nouns, 12 nouns have the same predominant sense in both BC and WSJ. The remaining 9 nouns that have different predominant senses in the BC and WSJ data are: *center*, *field*, *figure*, *interest*, *line*, *need*, *order*, *term*, *value*. The row *9 nouns* in Table 1 gives some information for this set of 9 nouns. To gauge the utility of this approach, we conduct experiments on these nouns by first assuming that we know the *true* predominant sense in the WSJ data. Assume that the WSJ predominant sense of a noun is s_i and s_i has n_i examples in the BC data. We then gather BC examples for a noun to adhere to this WSJ predominant sense, by gathering only up to n_i BC examples for each sense of this noun. This approach gives an average of 190 BC examples for the 9 nouns. This is higher than an average of 83 BC examples for these 9 nouns if BC examples are selected to follow the sense priors of WSJ evaluation data as described in the last subsection 6.2.

For these 9 nouns, the average KL-divergence between the sense priors of the original BC data and WSJ evaluation data is 0.81. This drops to 0.51 after ensuring that the predominant sense in BC is the same as that of WSJ, confirming that the sense priors in the newly gathered BC data more closely follow

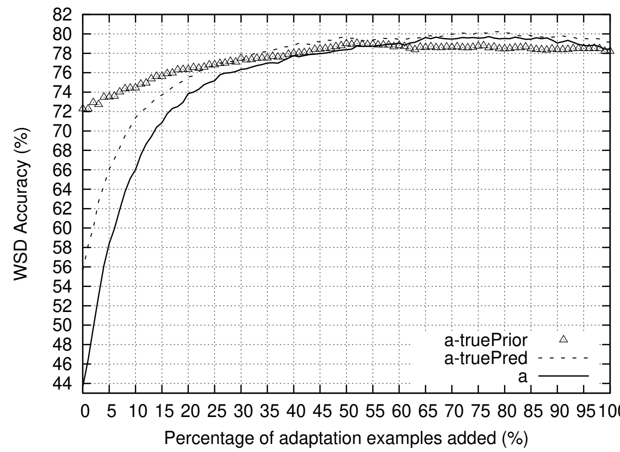

Figure 3: Using true predominant sense for the 9 nouns.

the sense priors in WSJ. Using this new set of training examples, we perform domain adaptation using active learning to obtain the curve *a-truePred* in Figure 3. For comparison, we also plot the curves *a* and *a-truePrior* for this set of 9 nouns in Figure 3. Results in Figure 3 show that *a-truePred* starts off at a higher accuracy and performs consistently better than the *a* curve. In contrast, though *a-truePrior* starts at a high accuracy, its performance is lower than *a-truePred* and *a* after 50% of adaptation examples are added. The approach represented by *a-truePred* is a compromise between ensuring that the sense priors in the training data follow as closely as possible the sense priors in the evaluation data, while retaining enough training examples. These results highlight the importance of striking a balance between these two goals.

In (McCarthy et al., 2004), a method was presented to determine the predominant sense of a word in a corpus. However, in (Chan and Ng, 2005), we showed that in a supervised setting where one has access to some annotated training data, the EM-based method in section 5 estimates the sense priors more effectively than the method described in (McCarthy et al., 2004). Hence, we use the EM-based algorithm to estimate the sense priors in the WSJ evaluation data for each of the 21 nouns. The sense with the highest estimated sense prior is taken as the predominant sense of the noun.

For the set of 12 nouns where the predominant

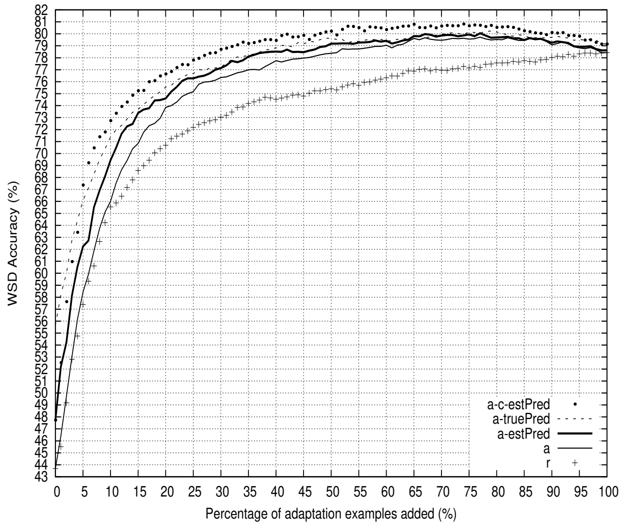

Figure 4: Using estimated predominant sense for the 9 nouns.

Accuracy	% adaptation examples needed			
	r	a	a-estPred	a-c-estPred
50%: 61.1	8	7 (0.88)	5 (0.63)	4 (0.50)
60%: 64.5	10	9 (0.90)	7 (0.70)	5 (0.50)
70%: 68.0	15	12 (0.80)	9 (0.60)	6 (0.40)
80%: 71.5	23	16 (0.70)	12 (0.52)	9 (0.39)
90%: 74.9	46	24 (0.52)	21 (0.46)	15 (0.33)
100%: 78.4	100	51 (0.51)	38 (0.38)	29 (0.29)

Table 2: Annotation savings and percentage of adaptation examples needed to reach various accuracies.

sense remains unchanged between BC and WSJ, the EM-based algorithm is able to predict that the predominant sense remains unchanged for *all* 12 nouns. Hence, we will focus on the 9 nouns which have different predominant senses between BC and WSJ for our remaining adaptation experiments. For these 9 nouns, the EM-based algorithm correctly predicts the WSJ predominant sense for 6 nouns. Hence, the algorithm is able to predict the correct predominant sense for 18 out of 21 nouns overall, representing an accuracy of 86%.

Figure 4 plots the curve *a-estPred*, which is similar to *a-truePred*, except that the predominant sense is now estimated by the EM-based algorithm. Employing count-merging with *a-estPred* produces the curve *a-c-estPred*. For comparison, the curves *r*, *a*, and *a-truePred* are also plotted. The results show that *a-estPred* performs consistently better than *a*, and *a-c-estPred* in turn performs better than *a-estPred*. Hence, employing the predicted predominant sense and count-merging, we further improve the effectiveness of the active learning-based adaptation process.

With reference to Figure 4, the WSD accuracies of the *r* and *a* curves before and after adaptation are 43.7% and 78.4% respectively. Starting from the mid-point 61.1% accuracy, which represents a 50% accuracy increase from 43.7%, we show in Table 2 the percentage of adaptation examples required by the various approaches to reach certain levels of WSD accuracies. For instance, to reach the final accuracy of 78.4%, *r*, *a*, *a-estPred*, and *a-c-estPred* require the addition of 100%, 51%, 38%, and 29% adaptation examples respectively. The numbers in brackets give the ratio of adaptation examples needed by *a*, *a-estPred*, and *a-c-estPred* versus random selection *r*. For instance, to reach a WSD accuracy of 78.4%, *a-c-estPred* needs only 29% adaptation examples, representing a ratio of 0.29 and an annotation saving of 71%. Note that this represents a more effective adaptation process than the basic active learning *a* approach, which requires 51% adaptation examples. Hence, besides showing that active learning can be used to reduce the annotation effort required for domain adaptation, we have further improved the effectiveness of the adaptation process by using the predicted predominant sense of the new domain and adopting the count-merging technique.

7 Related Work

In applying active learning for domain adaptation, Zhang et al. (2003) presented work on sentence boundary detection using generalized Winnow, while Tur et al. (2004) performed language model adaptation of automatic speech recognition systems. In both papers, out-of-domain and in-domain data were simply mixed together without MAP estimation such as count-merging. For WSD, Fujii et al. (1998) used selective sampling for a Japanese language WSD system, Chen et al. (2006) used active learning for 5 verbs using coarse-grained evaluation, and H. T. Dang (2004) employed active learning for another set of 5 verbs. However, their work only investigated the use of active learning to reduce the annotation effort necessary for WSD, but did not deal with the porting of a WSD system to a different domain. Escudero et al. (2000) used the DSO corpus to highlight the importance of the issue of domain dependence of WSD systems, but did not propose methods such as active learning or count-merging to address the specific problem of how to perform domain adaptation for WSD.

8 Conclusion

Domain adaptation is important to ensure the general applicability of WSD systems across different domains. In this paper, we have shown that active learning is effective in reducing the annotation effort required in porting a WSD system to a new domain. Also, we have successfully used an EM-based algorithm to detect a change in predominant sense between the training and new domain. With this information on the predominant sense of the new domain and incorporating count-merging, we have shown that we are able to improve the effectiveness of the original adaptation process achieved by the basic active learning approach.

Acknowledgement

Yee Seng Chan is supported by a Singapore Millennium Foundation Scholarship (ref no. SMF-2004-1076).

References

M. Bacchiani and B. Roark. 2003. Unsupervised language model adaptation. In *Proc. of IEEE ICASSP03*.

Y. S. Chan and H. T. Ng. 2005. Word sense disambiguation with distribution estimation. In *Proc. of IJCAI05*.

J. 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*.

H. T. Dang. 2004. *Investigations into the Role of Lexical Semantics in Word Sense Disambiguation*. PhD dissertation, University of Pennsylvania.

G. Escudero, L. Marquez, and G. Rigau. 2000. An empirical study of the domain dependence of supervised word sense disambiguation systems. In *Proc. of EMNLP/VLC00*.

A. Fujii, K. Inui, T. Tokunaga, and H. Tanaka. 1998. Selective sampling for example-based word sense disambiguation. *Computational Linguistics*, 24(4).

R. Koeling, D. McCarthy, and J. Carroll. 2005. Domain-specific sense distributions and predominant sense acquisition. In *Proc. of Joint HLT-EMNLP05*.

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*.

D. D. Lewis and W. A. Gale. 1994. A sequential algorithm for training text classifiers. In *Proc. of SIGIR94*.

B. Magnini and G. Cavaglia. 2000. Integrating subject field codes into WordNet. In *Proc. of LREC-2000*.

D. Martinez and E. Agirre. 2000. One sense per collocation and genre/topic variations. In *Proc. of EMNLP/VLC00*.

D. McCarthy, R. Koeling, J. Weeds, and J. Carroll. 2004. Finding predominant word senses in untagged text. In *Proc. of ACL04*.

G. A. Miller, M. Chodorow, S. Landes, C. Leacock, and R. G. Thomas. 1994. Using a semantic concordance for sense identification. In *Proc. of HLT94 Workshop on Human Language Technology*.

H. T. Ng and H. B. Lee. 1996. Integrating multiple knowledge sources to disambiguate word sense: An exemplar-based approach. In *Proc. of ACL96*.

B. Roark and M. Bacchiani. 2003. Supervised and unsupervised PCFG adaptation to novel domains. In *Proc. of HLT-NAACL03*.

M. Saerens, P. Latinne, and C. Decaestecker. 2002. Adjusting the outputs of a classifier to new a priori probabilities: A simple procedure. *Neural Computation*, 14(1).

D. H. Tur, G. Tur, M. Rahim, and G. Riccardi. 2004. Unsupervised and active learning in automatic speech recognition for call classification. In *Proc. of IEEE ICASSP04*.

T. Zhang, F. Damerau, and D. Johnson. 2003. Updating an NLP system to fit new domains: an empirical study on the sentence segmentation problem. In *Proc. of CONLL03*.