Supervised Domain Adaption for WSD

Eneko Agirre and Oier Lopez de Lacalle
IXA NLP Group
University of the Basque Country
Donostia, Basque Contry
{e.agirre,oier.lopezdelacalle}@ehu.es

Abstract

The lack of positive results on supervised domain adaptation for WSD have cast some doubts on the utility of hand-tagging general corpora and thus developing generic supervised WSD systems. In this paper we show for the first time that our WSD system trained on a general source corpus (BNC) and the target corpus, obtains up to 22% error reduction when compared to a system trained on the target corpus alone. In addition, we show that as little as 40% of the target corpus (when supplemented with the source corpus) is sufficient to obtain the same results as training on the full target data. The key for success is the use of unlabeled data with SVD, a combination of kernels and SVM.

1 Introduction

In many Natural Language Processing (NLP) tasks we find that a large collection of manually-annotated text is used to train and test supervised machine learning models. While these models have been shown to perform very well when tested on the text collection related to the training data (what we call the **source** domain), the performance drops considerably when testing on text from other domains (called **target** domains).

In order to build models that perform well in new (target) domains we usually find two settings (Daumé III, 2007). In the semi-supervised setting, the training hand-annotated text from the source domain is supplemented with unlabeled data from the target domain. In the supervised setting, we use training data from both the source and target domains to test on the target domain.

In (Agirre and Lopez de Lacalle, 2008) we studied semi-supervised Word Sense Disambiguation (WSD) adaptation, and in this paper we focus on supervised WSD adaptation. We compare the performance of similar supervised WSD systems on three different scenarios. In the **source to target scenario** the WSD system is trained on the source domain and tested on the target domain. In the **target scenario** the WSD system is trained and tested on the target domain (using cross-validation). In the **adaptation scenario** the WSD system is trained on both source and target domain and tested in the target domain (also using cross-validation over the target data). The source to target scenario represents a weak baseline for domain adaptation, as it does not use any examples from the target domain. The target scenario represents the hard baseline, and in fact, if the domain adaptation scenario does not yield better results, the adaptation would have failed, as it would mean that the source examples are not useful when we do have hand-labeled target examples.

Previous work shows that current state-of-the-art WSD systems are not able to obtain better results on the adaptation scenario compared to the target scenario (Escudero et al., 2000; Agirre and Martínez, 2004; Chan and Ng, 2007). This would mean that if a user of a generic WSD system (i.e. based on hand-annotated examples from a generic corpus) would need to adapt it to a specific domain, he would be better off throwing away the generic examples and hand-tagging domain examples directly. This paper will show that domain adaptation is feasible, even for difficult domain-related words, in the sense that generic corpora can be reused when deploying WSD systems in specific domains. We will also show that, given the source corpus, our technique can save up to 60% of effort when tagging domain-related occurrences.

We performed on a publicly available corpus which was designed to study the effect of domains in WSD (Koeling et al., 2005). It comprises 41

nouns which are highly relevant in the SPORTS and FINANCES domains, with 300 examples for each. The use of two target domains strengthens the conclusions of this paper.

Our system uses Singular Value Decomposition (SVD) in order to find correlations between terms, which are helpful to overcome the scarcity of training data in WSD (Gliozzo et al., 2005). This work explores how this ability of SVD and a combination of the resulting feature spaces improves domain adaptation. We present two ways to combine the reduced spaces: kernel combination with Support Vector Machines (SVM), and k Nearest-Neighbors (k-NN) combination.

The paper is structured as follows. Section 2 reviews prior work in the area. Section 3 presents the data sets used. In Section 4 we describe the learning features, including the application of SVD, and in Section 5 the learning methods and the combination. The experimental results are presented in Section 6. Section 7 presents the discussion and some analysis of this paper and finally Section 8 draws the conclusions.

2 Prior work

Domain adaptation is a practical problem attracting more and more attention. In the supervised setting, a recent paper by Daumé III (2007) shows that a simple feature augmentation method for SVM is able to effectively use both labeled target and source data to provide the best domain-adaptation results in a number of NLP tasks. His method improves or equals over previously explored more sophisticated methods (Daumé III and Marcu, 2006; Chelba and Acero, 2004). The feature augmentation consists in making three version of the original features: a general, a source-specific and a target-specific versions. That way the augmented source contains the general and source-specific version and the augmented target data general and specific versions. The idea behind this is that target domain data has twice the influence as the source when making predictions about test target data. We reimplemented this method and show that our results are better.

Regarding WSD, some initial works made a basic analysis of domain adaptation issues. Escudero et al. (2000) tested the supervised adaptation scenario on the DSO corpus, which had examples from the Brown corpus and Wall Street Journal corpus. They found that the source corpus did not help when tagging the target corpus, showing that tagged corpora from each domain would suffice, and concluding that hand tagging a large general corpus would not guarantee robust broad-coverage WSD. Agirre and Martínez (2000) used the DSO corpus in the supervised scenario to show that training on a subset of the source corpora that is topically related to the target corpus does allow for some domain adaptation.

More recently, Chan and Ng (2007) performed supervised domain adaptation on a manually selected subset of 21 nouns from the DSO corpus. They used active learning, count-merging, and predominant sense estimation in order to save target annotation effort. They showed that adding just 30% of the target data to the source examples the same precision as the full combination of target and source data could be achieved. They also showed that using the source corpus allowed to significantly improve results when only 10%-30% of the target corpus was used for training. Unfortunately, no data was given about the target corpus results, thus failing to show that domain-adaptation succeeded. In followup work (Zhong et al., 2008), the feature augmentation approach was combined with active learning and tested on the OntoNotes corpus, on a large domain-adaptation experiment. They reduced significantly the effort of hand-tagging, but only obtained domain-adaptation for smaller fractions of the source and target corpus. Similarly to these works we show that we can save annotation effort on the target corpus, but, in contrast, we do get domain adaptation when using the full dataset. In a way our approach is complementary, and we could also apply active learning to furhter reduce the number of target examples to be tagged.

Though not addressing domain adaptation, other works on WSD also used SVD and are closely related to the present paper. Ando (2006) used Alternative Structured Optimization. She first trained one linear predictor for each target word, and then performed SVD on 7 carefully selected submatrices of the feature-to-predictor matrix of weights. The system attained small but consistent improvements (no significance data was given) on the Senseval-3 lexical sample datasets using SVD and unlabeled data.

Gliozzo et al. (2005) used SVD to reduce the space of the term-to-document matrix, and then computed the similarity between train and test

instances using a mapping to the reduced space (similar to our SMA method in Section 4.2). They combined other knowledge sources into a complex kernel using SVM. They report improved performance on a number of languages in the Senseval-3 lexical sample dataset. Our present paper differs from theirs in that we propose an additional method to use SVD (the OMT method), and that we focus on domain adaptation.

In the semi-supervised setting, Blitzer et al. (2006) used Structural Correspondence Learning and unlabeled data to adapt a Part-of-Speech tagger. They carefully select so-called 'pivot features' to learn linear predictors, perform SVD on the weights learned by the predictor, and thus learn correspondences among features in both source and target domains. Our technique also uses SVD, but we directly apply it to all features, and thus avoid the need to define pivot features. In preliminary work we unsuccessfully tried to carry along the idea of pivot features to WSD. On the contrary, in (Agirre and Lopez de Lacalle, 2008) we show that methods closely related to those presented in this paper produce positive semi-supervised domain adaptation results for WSD.

The methods used in this paper originated in (Agirre et al., 2005; Agirre and Lopez de Lacalle, 2007), where SVD over a feature-to-documents matrix improved WSD performance with and without unlabeled data. The use of several k-NN classifiers trained on a number of reduced and original spaces was shown to get the best results in the Senseval-3 dataset and ranked second in the SemEval 2007 competition. The present paper extends this work and applies it to domain adaptation.

3 Data sets

The dataset we use was designed for domain-related WSD experiments by Koeling et al. (2005), and is publicly available. The examples come from the BNC (Leech, 1992) and the SPORTS and FINANCES sections of the Reuters corpus (Rose et al., 2002), comprising around 300 examples (roughly 100 from each of those corpora) for each of the 41 nouns. The nouns were selected because they were salient in either the SPORTS or FINANCES domains, or because they had senses linked to those domains. The occurrences were hand-tagged with the senses from WordNet (WN) version 1.7.1 (Fellbaum, 1998). In our experiments the BNC examples play the role of general **source** corpora, and the FINANCES and SPORTS examples the role of two specific domain **target** corpora.

Compared to the DSO corpus used in prior work (cf. Section 2) this corpus has been explicitly created for domain adaptation studies. DSO contains texts coming from the Brown corpus and the Wall Street Journal, but the texts are not classified according to specific domains (e.g. Sports, Finances), which make DSO less suitable to study domain adaptation. The fact that the selected nouns are related to the target domain makes the (Koeling et al., 2005) corpus more demanding than the DSO corpus, because one would expect the performance of a generic WSD system to drop when moving to the domain corpus for domain-related words (cf. Table 1), while the performance would be similar for generic words.

In addition to the labeled data, we also use unlabeled data coming from the three sources used in the labeled corpus: the 'written' part of the BNC (89.7M words), the FINANCES part of Reuters (32.5M words), and the SPORTS part (9.1M words).

4 Original and SVD features

In this section, we review the features and two methods to apply SVD over the features.

4.1 Features

We relied on the usual features used in previous WSD work, grouped in three main sets. **Local collocations** comprise the bigrams and trigrams formed around the target word (using either lemmas, word-forms, or PoS tags) , those formed with the previous/posterior lemma/word-form in the sentence, and the content words in a $\pm$4-word window around the target. **Syntactic dependencies** use the object, subject, noun-modifier, preposition, and sibling lemmas, when available. Finally, **Bag-of-words features** are the lemmas of the content words in the whole context, plus the salient bigrams in the context (Pedersen, 2001). We refer to these features as **original features**.

4.2 SVD features

Apart from the original space of features, we have used the so called **SVD features**, obtained from the projection of the feature vectors into the reduced space (Deerwester et al., 1990). Basically,

we set a term-by-document or feature-by-example matrix M from the corpus (see section below for more details). SVD decomposes M into three matrices, $M = U\Sigma V^T$. If the desired number of dimensions in the reduced space is p, we select p rows from Σ and V, yielding Σ_p and V_p respectively. We can map any feature vector $\vec{t}$ (which represents either a train or test example) into the p-dimensional space as follows: $\vec{t_p} = \vec{t}^T V_p \Sigma_p^{-1}$. Those mapped vectors have p dimensions, and each of the dimensions is what we call a SVD feature. We have explored two different variants in order to build the reduced matrix and obtain the SVD features, as follows.

Single Matrix for All target words (SVD-SMA). The method comprises the following steps: (i) extract bag-of-word features (terms in this case) from unlabeled corpora, (ii) build the term-by-document matrix, (iii) decompose it with SVD, and (iv) map the labeled data (train/test). This technique is very similar to previous work on SVD (Gliozzo et al., 2005; Zelikovitz and Hirsh, 2001). The dimensionality reduction is performed once, over the whole unlabeled corpus, and it is then applied to the labeled data of each word. The reduced space is constructed only with terms, which correspond to bag-of-words features, and thus discards the rest of the features. Given that the WSD literature shows that all features are necessary for optimal performance (Pradhan et al., 2007), we propose the following alternative to construct the matrix.

One Matrix per Target word (SVD-OMT). For each word: (i) construct a corpus with its occurrences in the labeled and, if desired, unlabeled corpora, (ii) extract all features, (iii) build the feature-by-example matrix, (iv) decompose it with SVD, and (v) map all the labeled training and test data for the word. Note that this variant performs one SVD process for each target word separately, hence its name.

When building the SVD-OMT matrices we can use only the training data (TRAIN) or both the train and unlabeled data (+UNLAB). When building the SVD-SMA matrices, given the small size of the individual word matrices, we always use both the train and unlabeled data (+UNLAB). Regarding the amount of data, based also on previous work, we used 50% of the available data for OMT, and the whole corpora for SMA. An important parameter when doing SVD is the number of dimensions in the reduced space (p). We tried two different values for p (25 and 200) in the BNC domain, and set a dimension for each classifier/matrix combination.

4.3 Motivation

The motivation behind our method is that although the train and test feature vectors overlap sufficiently in the usual WSD task, the domain difference makes such overlap more scarce. SVD implicitly finds correlations among features, as it maps related features into nearby regions in the reduced space. In the case of SMA, SVD is applied over the joint term-by-document matrix of labeled (and possibly unlabeled corpora), and it thus can find correlations among closely related words (e.g. *cat* and *dog*). These correlations can help reduce the gap among bag-of-words features from the source and target examples. In the case of OMT, SVD over the joint feature-by-example matrix of labeled and unlabeled examples of a word allows to find correlations among features that show similar occurrence patterns in the source and target corpora for the target word.

5 Learning methods

k-NN is a memory based learning method, where the neighbors are the k most similar labeled examples to the test example. The similarity among instances is measured by the cosine of their vectors. The test instance is labeled with the sense obtaining the maximum sum of the weighted vote of the k most similar contexts. We set k to 5 based on previous results published in (Agirre and Lopez de Lacalle, 2007).

Regarding SVM, we used linear kernels, but also purpose-built kernels for the reduced spaces and the combinations (cf. Section 5.2). We used the default soft margin (C=0). In previous experiments we learnt that C is very dependent on the feature set and training data used. As we will experiment with different features and training datasets, it did not make sense to optimize it across all settings.

We will now detail how we combined the original and SVD features in each of the machine learning methods.

5.1 k-NN combinations

Our k-NN combination method (Agirre et al., 2005; Agirre and Lopez de Lacalle, 2007) takes

advantage of the properties of k-NN classifiers and exploit the fact that a classifier can be seen as k points (number of nearest neighbor) each casting one vote. This makes easy to combine several classifiers, one for each feature space. For instance, taking two k-NN classifiers of $k = 5$, C_1 and C_2, we can combine them into a single $k = 10$ classifier, where five votes come from C_1 and five from C_2. This allows to smoothly combine classifiers from different feature spaces.

In this work we built three single k-NN classifiers trained on OMT, SMA and the original features, respectively. In order to combine them we weight each vote by the inverse ratio of its position in the rank of the single classifier, $(k - r_i + 1)/k$, where r_i is the rank.

5.2 Kernel combination

The basic idea of kernel methods is to find a suitable mapping function (ϕ) in order to get a better generalization. Instead of doing this mapping explicitly, kernels give the chance to do it inside the algorithm. We will formalize it as follows. First, we define the mapping function $\phi : \mathcal{X} \rightarrow \mathcal{F}$. Once the function is defined, we can use it in the kernel function in order to become an implicit function $K(\mathbf{x}, \mathbf{z}) = \langle \phi(\mathbf{x}) \cdot \phi(\mathbf{z}) \rangle$, where $\langle \cdot \rangle$ denotes a inner product between vectors in the feature space. This way, we can very easily define mappings representing different information sources and use this mappings in several machine learning algorithm. In our work we use SVM.

We defined three individual kernels (OMT, SMA and original features) and the combined kernel.

The **original feature kernel** (K_{Orig}) is given by the identity function over the features $\phi : \mathcal{X} \rightarrow \mathcal{X}$, defining the following kernel:

$$K_{Orig}(\mathbf{x_i}, \mathbf{x_j}) = \frac{\langle \mathbf{x_i} \cdot \mathbf{x_j} \rangle}{\sqrt{\langle \mathbf{x_i} \cdot \mathbf{x_i} \rangle \langle \mathbf{x_j} \cdot \mathbf{x_j} \rangle}}$$

where the denominator is used to normalize and avoid any kind of bias in the combination.

The **OMT kernel** (K_{Omt}) and **SMA kernel** (K_{Sma}) are defined using OMT and SMA projection matrices, respectively (cf. Section 4.2). Given the OMT function mapping $\phi_{omt} : \mathbb{R}^m \rightarrow \mathbb{R}^p$, where m is the number of the original features and p the reduced dimensionality, then we define $K_{Omt}(\mathbf{x_i}, \mathbf{x_j})$ as follows (K_{Sma} is defined similarly):

$$\frac{\langle \phi_{omt}(\mathbf{x_i}) \cdot \phi_{omt}(\mathbf{x_j}) \rangle}{\sqrt{\langle \phi_{omt}(\mathbf{x_i}) \cdot \phi_{omt}(\mathbf{x_i}) \rangle \langle \phi_{omt}(\mathbf{x_j}) \cdot \phi_{omt}(\mathbf{x_j}) \rangle}}$$

BNC $\rightarrow \mathcal{X}$	SPORTS	FINANCES
MFS	39.0±1.3	51.2±1.6
k-NN	51.7±1.3	60.4±1.6
SVM	53.9±1.3	62.9±1.6

Table 1: Source to target results: Train on BNC, test on SPORTS and FINANCES.

Finally, we define the kernel combination:

$$K_{Comb}(\mathbf{x_i}, \mathbf{x_j}) = \sum_{l=1}^{n} \frac{K_l(\mathbf{x_i}, \mathbf{x_j})}{\sqrt{K_l(\mathbf{x_i}, \mathbf{x_i}) K_l(\mathbf{x_j}, \mathbf{x_j})}}$$

where n is the number of single kernels explained above, and l the index for the kernel type.

6 Domain adaptation experiments

In this section we present the results in our two reference scenarios (source to target, target) and our reference scenario (domain adaptation). Note that all methods presented here have full coverage, i.e. they return a sense for all test examples, and therefore precision equals recall, and suffices to compare among systems. We have computed significance ranges for all results in this paper using bootstrap resampling (Noreen, 1989). Precision figures outside of these intervals are assumed to be significantly different from the related precision figures ($p < 0.05$).

6.1 Source to target scenario: BNC $\rightarrow \mathcal{X}$

In this scenario our supervised WSD systems are trained on the general source corpus (BNC) and tested on the specific target domains separately (SPORTS and FINANCES). We do not perform any kind of adaptation, and therefore the results are those expected for a generic WSD system when applied to domain-specific texts.

Table 1 shows the results for k-NN and SVM trained with the original features on the BNC. In addition, we also show the results for the Most Frequent Sense baseline (MFS) taken from the BNC. The second column denotes the accuracies obtained when testing on SPORTS, and the third column the accuracies for FINANCES. The low accuracy obtained with MFS, e.g. 39.0 of precision in SPORTS, shows the difficulty of this task. Both classifiers improve over MFS, and SVM outperforms significantly k-NN. These are weak baselines for the domain adaptation system.

$\mathcal{X} \rightarrow \mathcal{X}$	SPORTS		FINANCES	
	TRAIN	+ UNLAB	TRAIN	+ UNLAB
MFS	77.8±1.2	-	82.3±1.3	-
k-NN	84.5±1.0	-	87.1±1.0	-
SVM	85.1±1.0	-	87.0±1.0	-
k-NN-OMT	85.0±1.1	**86.1±0.9**	87.3±1.1	87.6±0.8
SVM-OMT	82.9±1.0	85.1±1.1	85.3±1.0	86.4±0.9
k-NN-SMA	-	81.1±1.1	-	83.2±1.4
SVM-SMA	-	81.3±1.1	-	84.1±1.0
k-NN-COMB	**86.0±0.9**	**86.7±1.0**	**87.9±0.9**	**88.6±0.8**
SVM-COMB	-	**86.5±0.9**	-	**88.5±0.8**

Table 2: Target results: train and test on SPORTS, train and test on FINANCES, using 3-fold cross-validation. Bold signals statistical significance over respective baseline classifier (first rows).

6.2 Target scenario $\mathcal{X} \rightarrow \mathcal{X}$

In this scenario we lay the harder baseline which the domain adaptation experiments should improve (cf. next section). The WSD systems are trained and tested on each of the target corpora (SPORTS and FINANCES) using 3-fold cross-validation.

Table 2 summarizes the results for this scenario. TRAIN denotes that only tagged data was used to train, +UNLAB denotes that we added unlabeled data related to the source corpus when computing SVD. The rows denote the classifier and the feature spaces used, which are organized in four sections. On the top rows we show the three baseline classifiers on the original features. The two sections below show the results of those classifiers on the reduced dimensions, OMT and SMA (cf. Section 4.2). Finally, the last rows show the results of the combination strategies (cf. Sections 5.1 and 5.2). Note that some of the cells have no result, because that combination is not applicable (e.g. using the train and unlabeled data in the original space).

First of all note that the results for the baselines (MFS, SVM, k-NN) are much larger than those in Table 1, showing that this dataset is specially demanding for supervised WSD, and particularly difficult for domain adaptation experiments. These results seem to indicate that the examples from the source general corpus could be of little use when tagging the target corpora. Note specially the difference in MFS performance. The priors of the senses are very different in the source and target corpora, which is a well-known shortcoming for supervised systems. Note the high results of the baseline classifiers, which leave small room for improvement.

The results for the more sophisticated methods show that SVD and unlabeled data helps slightly, but the differences are not statistically significant, except for k-NN-OMT on Sports. SMA decreases the performance compared to the classifiers trained on original features. The significant improvements come when the three strategies are combined in one. Both the kernel and k-NN combinations obtain statistically significant improvements over the respective single classifiers. Note that both k-NN and SVM combinations perform similarly.

In the combination strategy we show that unlabeled data helps slightly, because instead of only combining OMT and original features we have the opportunity to introduce SMA. The difference between both it is not enough to be significant. Note that it was not our aim to improve the results of the basic classifiers on this scenario, but given the fact that we are going to apply all these techniques in the domain adaptation scenario, we need to show these results as baselines. That is, in the next section we will try to obtain results which improve significantly over the best results in this section.

6.3 Domain adaptation scenario BNC + $\mathcal{X} \rightarrow \mathcal{X}$

In this last scenario we try to show that our WSD system trained on both source (BNC) and target (SPORTS and FINANCES) data performs better than the one trained on the target data alone. We also use 3-fold cross-validation for the target data, but the entire source data is used in each turn. The unlabeled data here refers to the combination of unlabeled source and target data.

The results are presented in table 3. Again, the columns denote if unlabeled data has been used in the learning process. The rows correspond to classifiers and the feature spaces involved. The first rows report the best results in the previous scenarios: BNC $\rightarrow \mathcal{X}$ for the source to target scenario, and $\mathcal{X} \rightarrow \mathcal{X}$ for the target scenario. The rest of the table corresponds to the domain adaptation scenario. The rows below correspond to the MFS and the baseline classifiers, the OMT and SMA results and the combination results. The last row shows the results for the feature augmentation algorithm (Daumé III, 2007).

Focusing on the results, the table shows that MFS decreases with respect to the target scenario (cf. Section 6.2) when the source data is added, probably caused by the different sense distribution in BNC and the target corpora. The baseline classi-

	SPORTS		FINANCES	
	TRAIN	+ UNLAB	TRAIN	+ UNLAB
BNC $\rightarrow \mathcal{X}$	53.9±1.3	-	62.9±1.6	-
$\mathcal{X} \rightarrow \mathcal{X}$	86.0±0.9	86.7±1.0	87.9±0.9	88.5±0.8
MFS	68.2±1.3	-	73.1±1.5	-
k-NN	81.3±1.1	-	86.0±0.9	-
SVM	84.7±1.0	-	87.5±0.7	-
k-NN-OMT	84.0±1.0	84.7±1.0	87.5±0.9	86.0±0.9
SVM-OMT	85.1±0.9	84.7±0.9	84.2±0.8	85.5±1.0
k-NN-SMA	-	77.1±1.2	-	81.6±1.2
SVM-SMA	-	78.1±1.2	-	80.7±1.1
k-NN-COMB	84.5±0.9	87.2±0.8	88.1±0.9	88.7±0.8
SVM-COMB	-	**88.4±0.9**	-	**89.7±0.8**
SVM-AUG	85.9±1.0	-	88.1±0.9	-

Table 3: BNC + $\mathcal{X} \rightarrow \mathcal{X}$: train on BNC and SPORTS, test on SPORTS (same for FINANCES. Bold signals statistical significance over best results on target scenario ($\mathcal{X} \rightarrow \mathcal{X}$).

fiers (k-NN and SVM) are not able to improve over the baseline classifiers on the target data alone, which is coherent with part research, and shows that straightforward domain adaptation does not work.

The following rows show that our reduction methods on themselves (OMT, SMA used by k-NN and SVM) also fail to perform better than in the target scenario, but the combinations using SVM do manage to improve significantly, showing that we were able to attain domain adaptation. In contrast, the feature augmentation approach (SVM-AUG) does not yield any improvement, showing the difficulty of domain adaptation for WSD, at least on this dataset.

7 Discussion and analysis

Table XXX summarizes the most improtant results. Compared to the target scenario, the kernel combination method with unlabeled data reduces the error on 22.1% and 17.6% over the baseline SVM (SPORTS and FINANCES respectively), and 12.7% and 9.0% over the kernel combination method. This gains are remarkable given the already high baseline, specially taking into consideration that the 41 nouns are closely related to the domain. This differences are statistical significant according to the Wilcoxon test with $p < 0.01$.

In addition, we carried extra experiments to check the learning curve. We fixed the source data and used increasing amounts of target data. We chose as baselines the SVM and k-NN-COMB approaches on the target scenario, and SVM-COMB and SVM-AUG as the domain adaptation approaches. The results are shown in figure 1 for SPORTS and figure 2 for FINANCES. The horizontal line corresponds to the performance of SVM on the target doman. The point where the learning curves cross the horizontal line show that domain adaptation kernel methods need between %60 and %65 less target data in order to have the same performance than the baseline SVM. The learning curves also show that, the kernel combination approach, no matter the amount of target data, always is above the rest of the classifier. This shows the robustness of our approach.

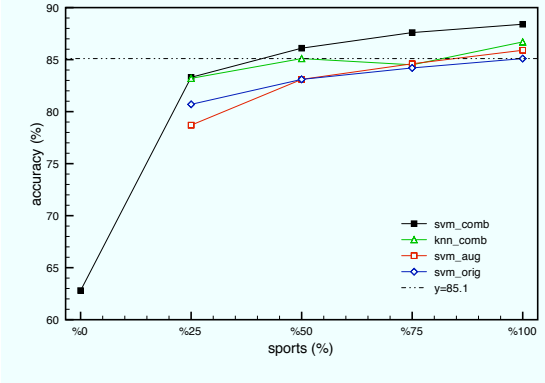

Figure 1: Learning curves for SPORTS target domain. The $\mathcal{X}$ axis denotes the amount of the SPORTS domain data and Y axis corresponds to the accuracy. Note that SVM-ORIG and k-NN-COMB are in-domain classifiers (baselines) and SVM-COMB and SVM-AUG are trained on source and target domain data.

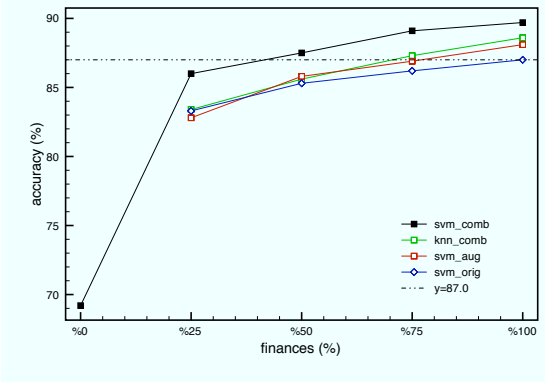

Figure 2: Learning curves for FINANCES target domain. The X axis denotes the amount of the FINANCES domain data and Y axis corresponds to the accuracy. Note that SVM-ORIG and k-NN-COMB are in-domain classifiers (baselines) and SVM-COMB and SVM-AUG are trained on source and target domain data.

8 Conclusion and future work

In this paper we explore the supervised domain adaptation issue for WSD using different strategies of SVD and two variants to combine them. Several experiments have been performed in different cross-domain scenarios. In the first scenario, the classifiers were trained on source domain data (BNC) and test on the target (SPORTS and FINANCES sections of Reuters). In the second scenario we set the baseline performing in-domain experiments. Systems were trained and tested in the target domain data. And in the last, we obtained the adaptation putting together the source and target domains data for training and tested the target domain data.

Our kernel method yields up to 22% error reduction compared to SVM on target domain data alone and using original features. In fact, we show that the combination of kernel method relying on SVD features can take advantage of both source and target domain when target test data in is predicting. The kernels are simple and elegant way to merge different information sources. The learning curves have demonstrated that the around 40% of the target data is enough to get adapted to the target domain, comparing the result of SVM on original features. The fact that we obtain coherent results in two target scenarios gives more robustness to our findings. In addition, given the fact that the nouns in the dataset are related to the target domain makes the adaptation scenario more demanding, as one would expect the performance of a generic WSD system to drop when moving to the domain corpus for domain-related words, while the performance would be similar for generic words.

In the future we plan to carry on a deeper analysis of domain adaptation and make word-by-word examination to have better understanding of this. Taking to the account the difficulty of this words (most of the words are salient in SPORTS or FINANCES domains) a different strategy depending on the type of word could be a better solution on domain adaptation task for WSD.

Acknowledgments

This work has been partially funded by the EU Commission (project KYOTO ICT-2007-211423) and Spanish Research Department (project KNOW TIN2006-15049-C03-01).

References

Eneko Agirre and Oier Lopez de Lacalle. 2007. Ubc-alm: Combining k-nn with svd for wsd. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 342–345, Prague, Czech Republic, June. Association for Computational Linguistics.

Eneko Agirre and Oier Lopez de Lacalle. 2008. On robustness and domain adaptation using SVD for word sense disambiguation. In *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, pages 17–24, Manchester, UK, August. Coling 2008 Organizing Committee.

Eneko Agirre and David Martínez. 2004. The effect of bias on an automatically-built word sense corpus. *Proceedings of the 4rd International Conference on Languages Resources and Evaluations (LREC)*.

E. Agirre, O.Lopez de Lacalle, and David Martínez. 2005. Exploring feature spaces with svd and unlabeled data for Word Sense Disambiguation. In *Proceedings of the Conference on Recent Advances on Natural Language Processing (RANLP'05)*, Borovets, Bulgaria.

Rie Kubota Ando. 2006. Applying alternating structure optimization to word sense disambiguation. In *Proceedings of the 10th Conference on Computational Natural Language Learning (CoNLL)*, pages 77–84, New York City.

John Blitzer, Ryan McDonald, and Fernando Pereira. 2006. Domain adaptation with structural correspondence learning. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, pages 120–128, Sydney, Australia, July. Association for Computational Linguistics.

Yee Seng Chan and Hwee Tou Ng. 2007. Domain adaptation with active learning for word sense disambiguation. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 49–56, Prague, Czech Republic, June. Association for Computational Linguistics.

Ciprian Chelba and Alex Acero. 2004. Adaptation of maximum entropy classifier: Little data can help a lot. In *Proceedings of of th Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Barcelona, Spain.

Hal Daumé III and Daniel Marcu. 2006. Domain adaptation for statistical classifiers. *Journal of Artificial Intelligence Research*, 26:101–126.

Hal Daumé III. 2007. Frustratingly easy domain adaptation. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 256–263, Prague, Czech Republic, June. Association for Computational Linguistics.

Scott Deerwester, Susan Dumais, Goerge Furnas, Thomas Landauer, and Richard Harshman. 1990. Indexing by Latent Semantic Analysis. *Journal of the American Society for Information Science*, 41(6):391–407.

Gerard Escudero, Lluiz Márquez, and German Rigau. 2000. An Empirical Study of the Domain Dependence of Supervised Word Sense Didanbiguation Systems. *Proceedings of the joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora, EMNLP/VLC*.

C. Fellbaum. 1998. *WordNet: An Electronic Lexical Database*. MIT Press.

Alfio Massimiliano Gliozzo, Claudio Giuliano, and Carlo Strapparava. 2005. Domain Kernels for Word Sense Disambiguation. *43nd Annual Meeting of the Association for Computational Linguistics. (ACL-05)*.

R. Koeling, D. McCarthy, and J. Carroll. 2005. Domain-specific sense distributions and predominant sense acquisition. In *Proceedings of the Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing. HLT/EMNLP*, pages 419–426, Ann Arbor, Michigan.

G. Leech. 1992. 100 million words of English: the British National Corpus. *Language Research*, 28(1):1–13.

David Martínez and Eneko Agirre. 2000. One Sense per Collocation and Genre/Topic Variations. *Conference on Empirical Method in Natural Language*.

Eric W. Noreen. 1989. *Computer-Intensive Methods for Testing Hypotheses*. John Wiley & Sons.

T. Pedersen. 2001. A Decision Tree of Bigrams is an Accurate Predictor of Word Sense. In *Proceedings of the Second Meeting of the North American Chapter of the Association for Computational Linguistics (NAACL-01)*, Pittsburgh, PA.

Sameer Pradhan, Edward Loper, Dmitriy Dligach, and Martha Palmer. 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.

Tony G. Rose, Mark Stevenson, and Miles Whitehead. 2002. The reuters corpus volumen 1 from yesterday's news to tomorrow's language resources. In *Proceedings of the Third International Conference on Language Resources and Evaluation (LREC-2002)*, pages 827–832, Las Palmas, Canary Islands.

Sarah Zelikovitz and Haym Hirsh. 2001. Using LSI for text classification in the presence of background text. In Henrique Paques, Ling Liu, and David Grossman, editors, *Proceedings of CIKM-01, 10th ACM International Conference on Information and Knowledge Management*, pages 113–118, Atlanta, US. ACM Press, New York, US.

Zhi Zhong, Hwee Tou Ng, and Yee Seng Chan. 2008. Word sense disambiguation using OntoNotes: An empirical study. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 1002–1010, Honolulu, Hawaii, October. Association for Computational Linguistics.