Improving Word Sense Disambiguation Using Topic Features

Jun Fu Cai, Wee Sun Lee
Department of Computer Science
National University of Singapore
3 Science Drive 2, Singapore 117543
{caijunfu, leews}@comp.nus.edu.sg

Yee Whye Teh
Gatsby Computational Neuroscience Unit
University College London
17 Queen Square, London WC1N 3AR, UK
ywteh@gatsby.ucl.ac.uk

Abstract

This paper presents a novel approach for exploiting the global context for the task of word sense disambiguation (WSD). This is done by using topic features constructed using the latent dirichlet allocation (LDA) algorithm on unlabeled data. The features are incorporated into a modified naïve Bayes network alongside other features such as part-of-speech of neighboring words, single words in the surrounding context, local collocations, and syntactic patterns. In both the English all-words task and the English lexical sample task, the method achieved significant improvement over the simple naïve Bayes classifier and higher accuracy than the best official scores on Senseval-3 for both task.

1 Introduction

Natural language tends to be ambiguous. A word often has more than one meanings depending on the context. Word sense disambiguation (WSD) is a natural language processing (NLP) task in which the correct meaning (sense) of a word in a given context is to be determined.

Supervised corpus-based approach has been the most successful in WSD to date. In such an approach, a corpus in which ambiguous words have been annotated with correct senses is first collected. Knowledge sources, or features, from the context of the annotated word are extracted to form the training data. A learning algorithm, like the support vector machine (SVM) or naïve Bayes, is then applied on the training data to learn the model. Finally, in testing, the learnt model is applied on the test data to assign the correct sense to any ambiguous word.

The features used in these systems usually include local features, such as part-of-speech (POS) of neighboring words, local collocations , syntactic patterns and global features such as single words in the surrounding context (bag-of-words) (Lee and Ng, 2002). However, due to the data scarcity problem, these features are usually very sparse in the training data. There are, on average, 11 and 28 training cases per sense in Senseval 2 and 3 lexical sample task respectively, and 6.5 training cases per sense in the SemCor corpus. This problem is especially prominent for the bag-of-words feature; more than hundreds of bag-of-words are usually extracted for each training instance and each feature could be drawn from any English word. A direct consequence is that the global context information, which the bag-of-words feature is supposed to capture, may be poorly represented.

Our approach tries to address this problem by clustering features to relieve the scarcity problem, specifically on the bag-of-words feature. In the process, we construct topic features, trained using the latent dirichlet allocation (LDA) algorithm. We train the topic model (Blei et al., 2003) on unlabeled data, clustering the words occurring in the corpus to a predefined number of topics. We then use the resulting topic model to tag the bag-of-words in the labeled corpus with topic distributions. We incorporate the distributions, called the topic features, using a simple Bayesian network, modified from naïve Bayes

Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, pp. 1015–1023, Prague, June 2007. ©2007 Association for Computational Linguistics

model, alongside other features and train the model on the labeled corpus. The approach gives good performance on both the lexical sample and all-words tasks on Senseval data.

The paper makes mainly two contributions. First, we are able to show that a feature that efficiently captures the global context information using LDA algorithm can significantly improve the WSD accuracy. Second, we are able to obtain this feature from unlabeled data, which spares us from any manual labeling work. We also showcase the potential strength of Bayesian network in the WSD task, obtaining performance that rivals state-of-arts methods.

2 Related Work

Many WSD systems try to tackle the data scarcity problem. Unsupervised learning is introduced primarily to deal with the problem, but with limited success (Snyder and Palmer, 2004). In another approach, the learning algorithm borrows training instances from other senses and effectively increases the training data size. In (Kohomban and Lee, 2005), the classifier is trained using grouped senses for verbs and nouns according to WordNet top-level synsets and thus effectively pooling training cases across senses within the same synset. Similarly, (Ando, 2006) exploits data from related tasks, using all labeled examples irrespective of target words for learning each sense using the Alternating Structure Optimization (ASO) algorithm (Ando and Zhang, 2005a; Ando and Zhang, 2005b). Parallel texts is proposed in (Resnik and Yarowsky, 1997) as potential training data and (Chan and Ng, 2005) has shown that using automatically gathered parallel texts for nouns could significantly increase WSD accuracy, when tested on Senseval-2 English all-words task.

Our approach is somewhat similar to that of using generic language features such as POS tags; the words are tagged with its semantic topic that may be trained from other corpuses.

3 Feature Construction

We first present the latent dirichlet allocation algorithm and its inference procedures, adapted from the original paper (Blei et al., 2003).

3.1 Latent Dirichlet Allocation

LDA is a probabilistic model for collections of discrete data and has been used in document modeling and text classification. It can be represented as a three level hierarchical Bayesian model, shown graphically in Figure 1. Given a corpus consisting of M documents, LDA models each document using a mixture over K topics, which are in turn characterized as distributions over words.

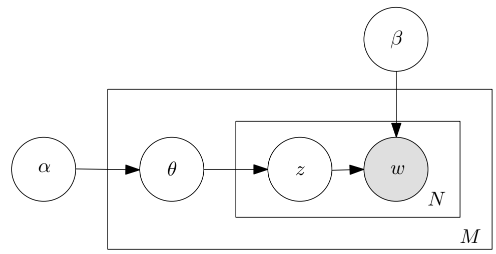

Figure 1: Graphical Model for LDA

In the generative process of LDA, for each document d we first draw the mixing proportion over topics θ_d from a Dirichlet prior with parameters α. Next, for each of the N_d words w_{dn} in document d, a topic z_{dn} is first drawn from a multinomial distribution with parameters θ_d. Finally w_{dn} is drawn from the topic specific distribution over words. The probability of a word token w taking on value i given that topic $z = j$ was chosen is parameterized using a matrix β with $\beta_{ij} = p(w = i|z = j)$. Integrating out θ_d's and z_{dn}'s, the probability $p(D|\alpha, \beta)$ of the corpus is thus:

$$\prod_{d=1}^{M} \int p(\theta_d|\alpha) \left(\prod_{n=1}^{N_d} \sum_{z_{dn}} p(z_{dn}|\theta_d) p(w_{dn}|z_{dn}, \beta) \right) d\theta_d$$

3.1.1 Inference

Unfortunately, it is intractable to directly solve the posterior distribution of the hidden variables given a document, namely $p(\theta, \mathbf{z}|\mathbf{w}, \alpha, \beta)$. However, (Blei et al., 2003) has shown that by introducing a set of variational parameters, γ and ϕ, a tight lower bound on the log likelihood of the probability can be found using the following optimization procedure:

$$(\gamma^*, \phi^*) = \arg\min_{\gamma,\phi} D(q(\theta, \mathbf{z}|\gamma, \phi) \| p(\theta, \mathbf{z}|\mathbf{w}, \alpha, \beta))$$

where

$$q(\theta, \mathbf{z}|\gamma, \phi) = q(\theta|\gamma) \prod_{n=1}^{N} q(z_n|\phi_n),$$

γ is the Dirichlet parameter for ϕ and the multinomial parameters $(\phi_1 \cdots \phi_N)$ are the free variational parameters. Note here γ is document specific instead of corpus specific like α. Graphically, it is represented as Figure 2. The optimizing values of γ and ϕ can be found by minimizing the Kullback-Leibler (KL) divergence between the variational distribution and the true posterior.

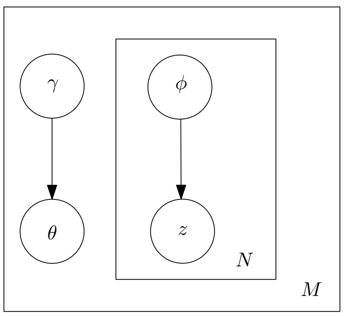

Figure 2: Graphical Model for Variational Inference

3.2 Baseline Features

For both the lexical sample and all-words tasks, we use the following standard *baseline features* for comparison.

POS Tags For each training or testing word, w, we include POS tags for P words prior to as well as after w within the same sentence boundary. We also include the POS tag of w. If there are fewer than P words prior or after w in the same sentence, we denote the corresponding feature as NIL.

Local Collocations Collocation $C_{i,j}$ refers to the ordered sequence of tokens (words or punctuations) surrounding w. The starting and ending position of the sequence are denoted i and j respectively, where a negative value refers to the token position prior to w. We adopt the same 11 collocation features as (Lee and Ng, 2002), namely $C_{-1,-1}$, $C_{1,1}$, $C_{-2,-2}$, $C_{2,2}$, $C_{-2,-1}$, $C_{-1,1}$, $C_{1,2}$, $C_{-3,-1}$, $C_{-2,1}$, $C_{-1,2}$, and $C_{1,3}$.

Bag-of-Words For each training or testing word, w, we get G words prior to as well as after w, within the same document. These features are position insensitive. The words we extract are converted back to their morphological root forms.

Syntactic Relations We adopt the same syntactic relations as (Lee and Ng, 2002). For easy reference, we summarize the features into Table 1.

POS of w	Features
Noun	Parent headword h
	POS of h
	Relative position of h to w
Verb	Left nearest child word of w, l
	Right nearest child word of w, r
	POS of l
	POS of r
	POS of w
	Voice of w
Adjective	Parent headword h
	POS of h

Table 1: Syntactic Relations Features

The exact values of P and G for each task are set according to cross validation result.

3.3 Topic Features

We first select an unlabeled corpus, such as 20 Newsgroups, and extract individual words from it (excluding stopwords). We choose the number of topics, K, for the unlabeled corpus and we apply the LDA algorithm to obtain the β parameters, where β represents the probability of a word w_i given a topic z_j, $p(w_i|z_j) = \beta_{ij}$. The model essentially clusters words that occurred in the unlabeled corpus according to K topics. The conditional probability $p(w_i|z_j) = \beta_{ij}$ is later used to tag the words in the unseen test example with the probability of each topic.

For some variants of the classifiers that we construct, we also use the γ parameter, which is document specific. For these classifiers, we may need to run the inference algorithm on the labeled corpus and possibly on the test documents. The γ parameter provides an approximation to the probability of selecting topic i in the document:

$$p(z_i|\gamma) = \frac{\gamma_i}{\sum_K \gamma_k}. \quad (1)$$

4 Classifier Construction

4.1 Bayesian Network

We construct a variant of the naïve Bayes network as shown in Figure 3. Here, w refers to the word. s refers to the sense of the word. In training, s is observed while in testing, it is not. The features f_1 to f_n are baseline features mentioned in Section 3.2 (including bag-of-words) while z refers to the latent topic that we set for clustering unlabeled corpus. The bag-of-words b are extracted from the neighbours of w and there are L of them. Note that L can be different from G, which is the number of bag-of-words in baseline features. Both will be determined by the validation result.

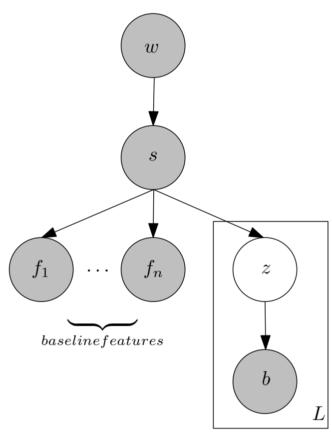

Figure 3: Graphical Model with LDA feature

The log-likelihood of an instance, $\ell(w, s, F, b)$ where F denotes the set of baseline features, can be written as

$$= \log p(w) + \log p(s|w) + \sum_F \log(p(f|s))$$
$$+ \sum_L \log\left(\sum_K p(z_k|s)p(b_l|z_k)\right).$$

The $\log p(w)$ term is constant and thus can be ignored. The first portion is normal naïve Bayes. And second portion represents the additional LDA plate. We decouple the training process into three separate stages. We first extract baseline features from the task training data, and estimate, using normal naïve Bayes, $p(s|w)$ and $p(f|s)$ for all w, s and f. The parameters associated with $p(b|z)$ are estimated using LDA from unlabeled data. Finally we estimate the parameters associated with $p(z|s)$. We experimented with three different ways of both doing the estimation as well as using the resulting model and chose one which performed best empirically.

4.1.1 Expectation Maximization Approach

For $p(z|s)$, a reasonable estimation method is to use maximum likelihood estimation. This can be done using the expectation maximization (EM) algorithm. In classification, we just choose s^* that maximizes the log-likelihood of the test instance, where:

$$s^* = \arg\max_s \ell(w, s, F, b)$$

In this approach, γ is never used which means the LDA inference procedure is not used on any labeled data at all.

4.1.2 Soft Tagging Approach

Classification in this approach is done using the full Bayesian network just as in the EM approach. However we do the estimation of $p(z|s)$ differently. Essentially, we perform LDA inference on the training corpus in order to obtain γ for each document. We then use the γ and β to obtain $p(z|b)$ for each word using

$$p(z_i|b_l, \gamma) = \frac{p(b_l|z_i)p(z_i|\gamma)}{\sum_K p(b_l|z_k)p(z_k|\gamma)},$$

where equation [1] is used for estimation of $p(z_i|\gamma)$.

This effectively transforms b to a topical distribution which we call a soft tag where each soft tag is probability distribution $t_1, \ldots, t_K$ on topics. We then use this topical distribution for estimating $p(z|s)$. Let s^i be the observed sense of instance i and $t_1^{ij}, \ldots, t_K^{ij}$ be the soft tag of the j-th bag-of-word feature of instance i. We estimate $p(z|s)$ as

$$p(z_{jk}|s) = \frac{\sum_{s^i=s} t_k^{ij}}{\sum_{s^i=s} \sum_{k'} t_{k'}^{ij}} \quad (2)$$

This approach requires us to do LDA inference on the corpus formed by the labeled training data, but not the testing data. This is because we need γ to get transformed topical distribution in order to learn $p(z|s)$ in the training. In the testing, we only apply the learnt parameters to the model.

4.1.3 Hard Tagging Approach

Hard tagging approach no longer assumes that z is latent. After $p(z|b)$ is obtained using the same procedure in Section 4.1.2, the topic z_i with the highest $p(z_i|b)$ among all K topics is picked to represent z. In this way, b is transformed into a single most "prominent" topic. This topic label is used in the same way as baseline features for both training and testing in a simple naïve Bayes model.

This approach requires us to perform the transformation both on the training as well as testing data, since z becomes an observed variable. LDA inference is done on two corpora, one formed by the training data and the other by testing data, in order to get the respective values of γ.

4.2 Support Vector Machine Approach

In the SVM (Vapnik, 1995) approach, we first form a training and a testing file using all standard features for each sense following (Lee and Ng, 2002) (one classifier per sense). To incorporate LDA feature, we use the same approach as Section 4.1.2 to transform b into soft tags, $p(z|b)$. As SVM deals with only observed features, we need to transform b both in the training data and in the testing data. Compared to (Lee and Ng, 2002), the only difference is that for each training and testing case, we have additional $L * K$ LDA features, since there are L bag-of-words and each has a topic distribution represented by K values.

5 Experimental Setup

We describe here the experimental setup on the English lexical sample task and all-words task.

We use MXPOST tagger (Adwait, 1996) for POS tagging, Charniak parser (Charniak, 2000) for extracting syntactic relations, SVMlight[1] for SVM classifier and David Blei's version of LDA[2] for LDA training and inference. All default parameters are used unless mentioned otherwise. For all standard baseline features, we use Laplace smoothing but for the soft tag (equation [2]), we use a smoothing parameter value of 2.

[1]http://svmlight.joachims.org

[2]http://www.cs.princeton.edu/~blei/lda-c/

5.1 Development Process

5.1.1 Lexical Sample Task

We use the Senseval-2 lexical sample task for preliminary investigation of different algorithms, datasets and other parameters. As the dataset is used extensively for this purpose, only the Senseval-3 lexical sample task is used for evaluation.

Selecting Bayesian Network The best achievable result, using the three different Bayesian network approaches, when validating on Senseval-2 test data is shown in Table 2. The parameters that are used are P = 3 and G = 3.

EM	68.0
Hard Tagging	65.6
Soft Tagging	68.9

Table 2: Results on Senseval-2 English lexical sample using different Bayesian network approaches.

From the results, it appears that both the EM and the Hard Tagging approaches did not yield as good results as the Soft Tagging approach did. The EM approach ignores the LDA inference result, γ, which we use to get our topic prior. This information is document specific and can be regarded as global context information. The Hard Tagging approach also uses less information, as the original topic distribution is now represented only by the topic with the highest probability of occurring. Therefore, both methods have information loss and are disadvantaged against the Soft Tagging approach. We use the Soft Tagging approach for the Senseval-3 lexical sample and the all-words tasks.

Unlabeled Corpus Selection The unlabeled corpus we choose to train LDA include 20 Newsgroups, Reuters, SemCor, Senseval-2 lexical sample data and Senseval-3 lexical sample data. Although the last three are labeled corpora, we only need the words from these corpora and thus they can be regarded as unlabeled too. For Senseval-2 and Senseval-3 data, we define the whole passage for each training and testing instance as one document.

The relative effect using different corpus and combinations of them is shown in Table 3, when validating on Senseval-2 test data using the Soft Tagging approach.

Corpus	$\|w\|$	K	L	Senseval-2
20 Newsgroups	1.7M	40	60	67.9
Reuters	1.3M	30	60	65.5
SemCor	0.3M	30	60	66.9
Senseval-2	0.6M	30	40	66.9
Senseval-3	0.6M	50	60	67.6
All	4.5M	60	40	68.9

Table 3: Effect of using different corpus for LDA training, $|w|$ represents the corpus size in terms of the number of words in the corpus

The 20 Newsgroups corpus yields the best result if used individually. It has a relatively larger corpus size at 1.7 million words in total and also a well balanced topic distribution among its documents, ranging across politics, finance, science, computing, etc. The Reuters corpus, on the other hand, focuses heavily on finance related articles and has a rather skewed topic distribution. This probably contributed to its inferior result. However, we found that the best result comes from combining all the corpora together with $K = 60$ and $L = 40$.

Results for Optimized Configuration As baseline for the Bayesian network approaches, we use naïve Bayes with all baseline features. For the baseline SVM approach, we choose P = 3 and include all the words occurring in the training and testing passage as bag-of-words feature.

The F-measure result we achieve on Senseval-2 test data is shown in Table 4. Our four systems are listed as the top four entries in the table. Soft Tag refers to the soft tagging Bayesian network approach. Note that we used the Senseval-2 test data for optimizing the configuration (as is done in the ASO result). Hence, the result should not be taken as reliable. Nevertheless, it is worth noting that the improvement of Bayesian network approach over its baseline is very significant ($+5.5\%$). On the other hand, SVM with topic features shows limited improvement over its baseline ($+0.8\%$).

Bayes (Soft Tag)	68.9
SVM-Topic	66.0
SVM baseline	65.2
NB baseline	63.4
ASO(best configuration)(Ando, 2006)	68.1
Classifier Combination(Florian, 2002)	66.5
Polynomial KPCA(Wu et al., 2004)	65.8
SVM(Lee and Ng, 2002)	65.4
Senseval-2 Best System	64.2

Table 4: Results (best configuration) compared to previous best systems on Senseval-2 English lexical sample task.

5.1.2 All-words Task

In the all-words task, no official training data is provided with Senseval. We follow the common practice of using the SemCor corpus as our training data. However, we did not use SVM approach in this task as there are too few training instances per sense for SVM to achieve a reasonably good accuracy.

As there are more training instances in SemCor, $230,000$ in total, we obtain the optimal configuration using 10 fold cross validation on the SemCor training data. With the optimal configuration, we test our system on both Senseval-2 and Senseval-3 official test data.

For baseline features, we set P = 3 and B = 1. We choose a LDA training corpus comprising 20 Newsgroups and SemCor data, with number of topics K = 40 and number of LDA bag-of-words L = 14.

6 Results

We now present the results on both English lexical sample task and all-words task.

6.1 Lexical Sample Task

With the optimal configurations from Senseval-2, we tested the systems on Senseval-3 data. Table 5 shows our F-measure result compared to some of the best reported systems. Although SVM with topic features shows limited success with only a 0.6% improvement, the Bayesian network approach has again demonstrated a good improvement of 3.8% over its baseline and is better than previous reported best systems except ASO(Ando, 2006).

Bayes (Soft Tag)	73.6
SVM-topic	73.0
SVM baseline	72.4
NB baseline	69.8
ASO(Ando, 2006)	74.1
SVM-LSA (Strapparava et al., 2004)	73.3
Senseval-3 Best System(Grozea, 2004)	72.9

Table 5: Results compared to previous best systems on Senseval-3 English lexical sample task.

6.2 All-words Task

The F-measure micro-averaged result for our systems as well as previous best systems for Senseval-2 and Senseval-3 all-words task are shown in Table 6 and Table 7 respectively. Bayesian network with soft tagging achieved 2.6% improvement over its baseline in Senseval-2 and 1.7% in Senseval-3. The results also rival some previous best systems, except for SMUaw (Mihalcea, 2002) which used additional labeled data.

Bayes (Soft Tag)	66.3
NB baseline	63.7
SMUaw (Mihalcea, 2002)	69.0
Simil-Prime (Kohomban and Lee, 2005)	66.4
Senseval-2 Best System (CNTS-Antwerp (Hoste et al., 2001))	63.6

Table 6: Results compared to previous best systems on Senseval-2 English all-words task.

Bayes (Soft Tag)	66.1
NB baseline	64.6
Simil-Prime (Kohomban and Lee, 2005)	66.1
Senseval-3 Best System (GAMBL-AW-S(Decadt et al., 2004))	65.2
Senseval-3 2nd Best System (SenseLearner (Mihalcea and Faruque, 2004))	64.6

Table 7: Results compared to previous best systems on Senseval-3 English all-words task.

6.3 Significance of Results

We perform the χ^2-test, using the Bayesian network and its naïve Bayes baseline (NB baseline) as pairs, to verify the significance of these results. The result is reported in Table 8. The results are significant at 90% confidence level, except for the Senseval-3 all-words task.

	Senseval-2	Senseval-3
All-word	0.0527	0.2925
Lexical Sample	$<$0.0001	0.0002

Table 8: P value for χ^2-test significance levels of results.

6.4 SVM with Topic Features

The results on lexical sample task show that SVM benefits less from the topic feature than the Bayesian approach. One possible reason is that SVM baseline is able to use all bag-of-words from surrounding context while naïve Bayes baseline can only use very few without decreasing its accuracy, due to the sparse representation. In this sense, SVM baseline already captures some of the topical information, leaving a smaller room for improvement. In fact, if we exclude the bag-of-words feature from the SVM baseline and add in the topic features, we are able to achieve almost the same accuracy as we did with both features included, as shown in Table 9. This further shows that the topic feature is a better representation of global context than the bag-of-words feature.

SVM baseline	72.4
SVM baseline - BAG + topic	73.5
SVM-topic	73.6

Table 9: Results on Senseval-3 English lexical sample task

6.5 Results on Different Parts-of-Speech

We analyse the result obtained on Senseval-3 English lexical sample task (using Senseval-2 optimal configuration) according to the test instance's part-of-speech, which includes noun, verb and adjective, compared to the naïve Bayes baseline. Table 10 shows the relative improvement on each part-of-speech. The second column shows the number of testing instances belonging to the particular part-of-speech. The third and fourth column shows the

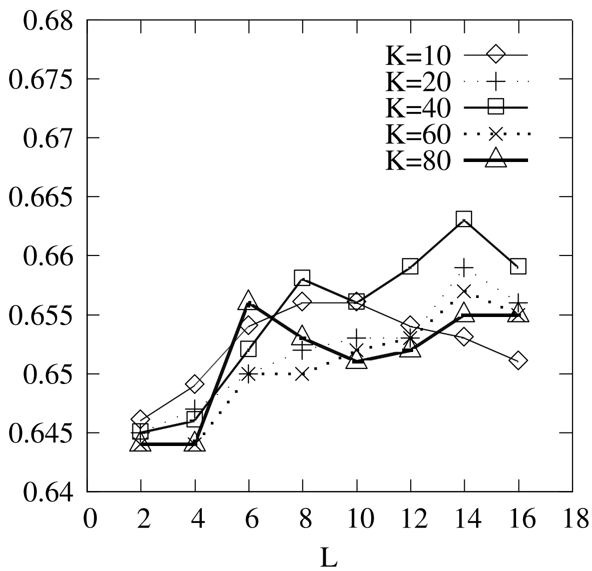

Figure 4: Accuracy with varing L and K on Senseval-2 all-words task

accuracy achieved by naïve Bayes baseline and the Bayesian network. Adjectives show no improvement while verbs show a moderate $+2.2\%$ improvement. Nouns clearly benefit from topical information much more than the other two parts-of-speech, obtaining a $+5.7\%$ increase over its baseline.

POS	Total	NB baseline	Bayes (Soft Tag)
Noun	1807	69.5	75.2
Verb	1978	71.1	73.5
Adj	159	57.2	57.2
Total	3944	69.8	73.6

Table 10: Improvement with different POS on Senseval-3 lexical sample task

6.6 Sensitivity to L and K

We tested on Senseval-2 all-words task using different L and K. Figure 4 is the result.

6.7 Results on SemEval-1

We participated in SemEval-1 English coarse-grained all-words task (task 7), English fine-grained all-words task (task 17, subtask 3) and English coarse-grained lexical sample task (task 17, subtask 1), using the method described in this paper. For all-words task, we use Senseval-2 and Senseval-3 all-words task data as our validation set to fine tune the parameters. For lexical sample task, we use the training data provided as the validation set.

We achieved 88.7%, 81.6% and 57.6% for coarse-grained lexical sample task, coarse-grained all-words task and fine-grained all-words task respectively. The results ranked first, second and fourth in the three tasks respectively.

7 Conclusion and Future Work

In this paper, we showed that by using LDA algorithm on bag-of-words feature, one can utilise more topical information and boost the classifiers accuracy on both English lexical sample and all-words task. Only unlabeled data is needed for this improvement. It would be interesting to see how the feature can help on WSD of other languages and other natural language processing tasks such as named-entity recognition.

References

Y. K. Lee and H. T. Ng. 2002. An Empirical Evaluation of Knowledge Sources and Learning Algorithms for Word Sense Disambiguation. In *Proc. of EMNLP*.

B. Snyder and M. Palmer. 2004. The English All-Words Task. In *Proc. of Senseval-3*.

U. S. Kohomban and W. S. Lee 2005. Learning Semantic Classes for Word Sense Disambiguation. In *Proc. of ACL*.

R. K. Ando. 2006. Applying Alternating Structure Optimization to Word Sense Disambiguation. In *Proc. of CoNLL*.

Y. S. Chan and H. T. Ng 2005. Scaling Up Word Sense Disambiguation via Parallel Texts. In *Proc. of AAAI*.

R. K. Ando and T. Zhang. 2005a. A Framework for Learning Predictive Structures from Multiple Tasks and Unlabeled Data. *Journal of Machine Learning Research*.

R. K. Ando and T. Zhang. 2005b. A High-Performance Semi-Supervised Learning Method for Text Chunking. In *Proc. of ACL*.

P. Resnik and D. Yarowsky. 1997. A Perspective on Word Sense Disambiguation Methods and Their Evaluation. In *Proc. of ACL*.

D. M. Blei and A. Y. Ng and M. I. Jordan. 2003. Latent Dirichlet Allocation. *Journal of Machine Learning Research*.

A. Ratnaparkhi 1996. A Maximum Entropy Model for Part-of-Speech Tagging. In *Proc. of EMNLP*.

E. Charniak 2000. A Maximum-Entropy-Inspired Parser. In *Proc. of the 1st Meeting of the North American Chapter of the Association for Computational Linguistics*.

V. N. Vapnik 1995. The Nature of Statistical Learning Theory. Springer-Verlag, New York.

R. Florian and D. Yarowsky 2002. Modeling consensus: Classifier Combination for Word Sense Disambiguation. In *Proc. of EMNLP*.

D. Wu and W. Su and M. Carpuat. 2004. A Kernel PCA Method for Superior Word Sense Disambiguation. In *Proc. of ACL*.

C. Strapparava and A. Gliozzo and C. Giuliano 2004. Pattern Abstraction and Term Similarity for Word Sense Disambiguation: IRST at Senseval-3. In *Proc. of Senseval-3*.

C. Grozea 2004. Finding Optimal Parameter Settings for High Performance Word Sense Disambiguation. In *Proc. of Senseval-3*.

R. Mihalcea 2002. Bootstrapping Large Sense Tagged Corpora. In *Proc. of the 3rd International Conference on Languages Resources and Evaluations*.

V. Hoste and A. Kool and W. Daelmans 2001. Classifier Optimization and Combination in English All Words Task. In *Proc. of Senseval-2*.

B. Decadt and V. Hoste and W. Daelmans 2004. GAMBL, Genetic Algorithm Optimization of Memory-Based WSD. In *Proc. of Senseval-3*.

R. Mihalcea and E. Faruque 2004. Sense-learner: Minimally Supervised Word Sense Disambiguation for All Words in Open Text. In *Proc. of Senseval-3*.