NUS-ML: 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

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). The same method with different labeled data is used for the tasks; SemCor is the labeled corpus used to train our system for the all-words tasks while the labeled corpus that is provided is used for the lexical sample task. The knowledge sources include part-of-speech of neighboring words, single words in the surrounding context, local collocations, and syntactic patterns. In addition, we constructed a topic feature, targeted to capture the global context information, using the latent dirichlet allocation (LDA) algorithm with unlabeled corpus. A modified naïve Bayes classifier is constructed to incorporate all the features. We achieved 81.6%, 57.6%, 88.7% for coarse-grained all-words task, fine-grained all-words task and coarse-grained lexical sample task respectively.

1 Introduction

Supervised corpus-based approach has been the most successful in WSD to date. However, this approach faces severe data scarcity problem, resulting features being sparsely represented in the training data. This problem is especially prominent for the bag-of-words feature. A direct consequence is that the global context information, which the bag-of-words feature is supposed to capture, may be poorly represented.

Our system 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 model, alongside other features and train the model on the labeled corpus.

2 Feature Construction

2.1 Baseline Features

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

POS Tags For each word instance 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 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 $C_{1,3}$.

Proceedings of the 4th International Workshop on Semantic Evaluations (SemEval-2007), pages 249–252,
Prague, June 2007. ©2007 Association for Computational Linguistics

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 validation result.

2.2 Latent Dirichlet Allocation

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

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.

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

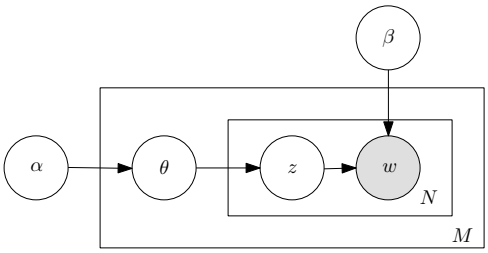

Figure 1: Graphical Model for LDA

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:

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

In variational inference, the latent variables θ_d and z_{dn} are assumed independent and updates to the variational posteriors for θ_d and z_{dn} are derived (Blei et al., 2003). It can be shown that the variational posterior for θ_d is a Dirichlet distribution, say with variational parameters γ_d, which we shall use in the following to construct topic features.

2.3 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.

We also use the document-specific γ_d parameters. Specifically, we need to run the inference algorithm on the labeled corpus to get γ_d for each document d in the corpus. The γ_d parameter provides an approximation to the probability of selecting topic i in the document:

$$p(z_i|\gamma_d) = \frac{\gamma_{di}}{\sum_K \gamma_{dk}}. \quad (1)$$

3 Classifier Construction

We construct a variant of the naïve Bayes network as shown in Figure 2. 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 2.1 (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.

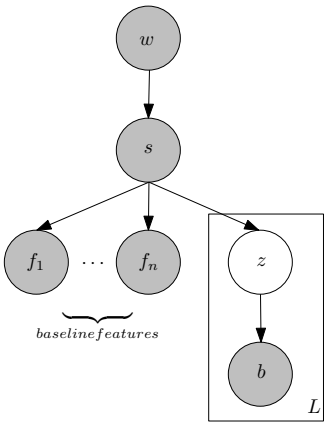

Figure 2: Graphical Model with LDA feature

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 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.

Next, the parameters associated with $p(b|z)$ are estimated using LDA from unlabeled data, which is β. To estimate $p(z|s)$, we perform LDA inference on the training corpus in order to obtain γ_d for each document d. We then use the γ_d and β to obtain $p(z|b)$ for each word using

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

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

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 Experimental Setup

We describe here the experimental setup on the English lexical sample task and all-words task. Note that we do not distinguish the two all-words tasks as the same parameters will be applied.

For lexical sample task, we use 5-fold cross validation on the training data provided to determine our parameters. For all-words task, we use SemCor as our training data and validate on Senseval-2 and Senseval-3 all-words test data.

We use MXPOST tagger (Adwait, 1996) for POS tagging, Charniak parser (Charniak, 2000) for extracting syntactic relations, and David Blei's version of LDA[1] for LDA training and inference. All default parameters are used unless mentioned otherwise.

For the all-word tasks, we use sense 1 as back-off for words that have not appeared in SemCor. We use the same fine-grained system for both the coarse and fine-grained all-words tasks. We make predictions

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

for all words for all the systems - precision, recall and accuracy scores are all the same.

Baseline features For lexical sample task, we choose $P = 3$ and $G = 3$. For all-words task, we choose $P = 3$ and $G = 1$. ($G = 1$ means only the nearest word prior and after the test word.)

Smoothing For all standard baseline features, we use Laplace smoothing but for the soft tag (equation (2)), we use a smoothing parameter value of 2 for all-words task and 0.1 for lexical sample task.

Unlabeled Corpus Selection The unlabeled corpus we select from for LDA training include 20 Newsgroups, Reuters, SemCor, Senseval-2 lexical sample data, Senseval-3 lexical sample data and SemEval-1 lexical sample data. Although the last four are labeled corpora, we only need the words from these corpora and thus they can be regarded as unlabeled too. For lexical sample data, we define the whole passage for each training and testing instance as one document.

For lexical sample task, we use all the unlabeled corpus mentioned with $K = 60$ and $L = 18$. For all-words task, we use a corpora consisting only 20 Newsgroups and SemCor with $K = 40$ and $L = 14$.

Validation Result Table 2 shows the results we get on the validation sets. We give both the system accuracy (named as Soft Tag) and the naïve Bayes result with only standard features as baseline.

Validation Set	Soft Tag	NB baseline
SE-2 All-words	66.3	63.7
SE-3 All-words	66.1	64.6
Lexical Sample	89.3	87.9

Table 2: Validation set results (best configuration).

5 Official Results

We now present the official results on all three tasks we participated in, summarized in Table 3.

The system ranked first, fourth and second in the lexical sample task, fine-grained all-words task and coarse-grained all-words task respectively. For coarse-grained all-words task, we obtained 86.1, 88.3, 81.4, 76.7 and 79.1 for each document, from d001 to d005.

Task	Precision/Recall
Lexical sample(Task 17)	88.7
Fine-grained all-words(Task 17)	57.6
Course-grained all-words(Task 7)	81.6

Table 3: Official Results

5.1 Analysis of Results

For the lexical sample task, we compare the results to that of our naïve Bayes baseline and Support Vector Machine (SVM) (Vapnik, 1995) baseline. Our SVM classifier (using SVMlight) follows that of (Lee and Ng, 2002), which ranked the third in Senseval-3 English lexical sample task. We also analyse the result according to the test instance's part-of-speech and find that the improvements are consistent for both noun and verb.

System	Noun	Verb	Total
Soft Tag	92.7	84.2	88.7
NB baseline	91.7	83.5	87.8
SVM baseline	91.6	83.1	87.6

Table 4: Analysis on different POS on English lexical sample task

Our coarse-grained all-words task result outperformed the first sense baseline score of 0.7889 by about 2.7%.

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*.

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.