

A Semi-Supervised Feature Clustering Algorithm with Application to Word Sense Disambiguation

Zheng-Yu Niu, Dong-Hong Ji

Institute for Infocomm Research
21 Heng Mui Keng Terrace
119613 Singapore
{zniu, dhji}@i2r.a-star.edu.sg

Chew Lim Tan

Department of Computer Science
National University of Singapore
3 Science Drive 2
117543 Singapore
tancl@comp.nus.edu.sg

Abstract

In this paper we investigate an application of feature clustering for word sense disambiguation, and propose a semi-supervised feature clustering algorithm. Compared with other feature clustering methods (ex. supervised feature clustering), it can infer the distribution of class labels over (unseen) features unavailable in training data (labeled data) by the use of the distribution of class labels over (seen) features available in training data. Thus, it can deal with both seen and unseen features in feature clustering process. Our experimental results show that feature clustering can aggressively reduce the dimensionality of feature space, while still maintaining state of the art sense disambiguation accuracy. Furthermore, when combined with a semi-supervised WSD algorithm, semi-supervised feature clustering outperforms other dimensionality reduction techniques, which indicates that using unlabeled data in learning process helps to improve the performance of feature clustering and sense disambiguation.

1 Introduction

This paper deals with word sense disambiguation (WSD) problem, which is to assign an appropriate sense to an occurrence of a word in a given context. Many corpus based statistical methods have been proposed to solve this problem, including supervised learning algorithms (Leacock et al., 1998; Towel and Voorheest, 1998), weakly supervised learning algorithms (Dagan and Itai, 1994; Li and Li, 2004; Mihalcea, 2004; Niu et al., 2005; Park et al., 2000;

Yarowsky, 1995), unsupervised learning algorithms (or word sense discrimination) (Pedersen and Bruce, 1997; Schütze, 1998), and knowledge based algorithms (Lesk, 1986; McCarthy et al., 2004).

In general, the most common approaches start by evaluating the co-occurrence matrix of features versus contexts of instances of ambiguous word, given sense-tagged training data for this target word. As a result, contexts are usually represented in a high-dimensional sparse feature space, which is far from optimal for many classification algorithms. Furthermore, processing data lying in high-dimensional feature space requires large amount of memory and CPU time, which limits the scalability of WSD model to very large datasets or incorporation of WSD model into natural language processing systems.

Standard dimensionality reduction techniques include (1) supervised feature selection and supervised feature clustering when given labeled data, (2) unsupervised feature selection, latent semantic indexing, and unsupervised feature clustering when only unlabeled data is available. Supervised feature selection improves the performance of an exemplar based learning algorithm over SENSEVAL-2 data (Mihalcea, 2002), Naive Bayes and decision tree over SENSEVAL-1 and SENSEVAL-2 data (Lee and Ng, 2002), but feature selection does not improve SVM and Adaboost over SENSEVAL-1 and SENSEVAL-2 data (Lee and Ng, 2002) for word sense disambiguation. Latent semantic indexing (LSI) studied in (Schütze, 1998) improves the performance of sense discrimination, while unsupervised feature selection also improves the performance of word sense discrimination (Niu et al., 2004). But little work is done on using feature clustering to conduct dimensionality reduction for WSD. This paper will describe an application of feature

clustering technique to WSD task.

Feature clustering has been extensively studied for the benefit of text categorization and document clustering. In the context of text categorization, supervised feature clustering algorithms (Baker and McCallum, 1998; Bekkerman et al., 2003; Slonim and Tishby, 2001) usually cluster words into groups based on the distribution of class labels over features, which can compress the feature space much more aggressively while still maintaining state of the art classification accuracy. In the context of document clustering, unsupervised feature clustering algorithms (Dhillon, 2001; Dhillon et al., 2002; Dhillon et al., 2003; El-Yaniv and Souroujon, 2001; Slonim and Tishby, 2000) perform word clustering by the use of word-document co-occurrence matrix, which can improve the performance of document clustering by clustering documents over word clusters.

Supervised feature clustering algorithm groups features into clusters based on the distribution of class labels over features. But it can not group unseen features (features that do not occur in labeled data) into meaningful clusters since there are no class labels associated with these unseen features. On the other hand, while given labeled data, unsupervised feature clustering method can not utilize class label information to guide feature clustering procedure. While, as a promising classification strategy, semi-supervised learning methods (Zhou et al., 2003; Zhu and Ghahramani, 2002; Zhu et al., 2003) usually utilize all the features occurring in labeled data and unlabeled data. So in this paper we propose a semi-supervised feature clustering algorithm to overcome this problem. Firstly, we try to induce class labels for unseen features based on the similarity among seen features and unseen features. Then all the features (including seen features and unseen features) are clustered based on the distribution of class labels over them.

This paper is organized as follows. First, we will formulate a feature clustering based WSD problem in section 2. Then in section 3 we will describe a semi-supervised feature clustering algorithm. Section 4 will provide experimental results of various dimensionality reduction techniques with combination of state of the art WSD algorithms on SENSEVAL-3 data. Section 5 will provide a review

of related work on feature clustering. Finally we will conclude our work and suggest possible improvement in section 6.

2 Problem Setup

Let $X = \{x_i\}_{i=1}^n$ be a set of contexts of occurrences of an ambiguous word w , where x_i represents the context of the i -th occurrence, and n is the total number of this word's occurrences. Let $S = \{s_j\}_{j=1}^c$ denote the sense tag set of w . The first l examples $x_g (1 \leq g \leq l)$ are labeled as $y_g (y_g \in S)$ and other $u (l+u = n)$ examples $x_h (l+1 \leq h \leq n)$ are unlabeled. The goal is to predict the sense of w in context x_h by the use of label information of x_g and similarity information among examples in X .

We use \tilde{F} to represent feature clustering result into $N_{\tilde{F}}$ clusters when F is a set of features. After feature clustering, any context x_i in X can be represented as a vector over feature clusters \tilde{F} . Then we can use supervised methods (ex. SVM) (Lee and Ng, 2002) or semi-supervised methods (ex. label propagation algorithm) (Niu et al., 2005) to perform sense disambiguation on unlabeled instances of target word.

3 Semi-Supervised Feature Clustering Algorithm

In supervised feature clustering process, F consists of features occurring in the first l labeled examples, which can be denoted as F_L . But in the setting of transductive learning, semi-supervised learning algorithms will utilize not only the features in labeled examples (F_L), but also unseen features in unlabeled examples (denoted as $F_{\bar{L}}$). $F_{\bar{L}}$ consists of the features that occur in unlabeled data, but never appear in labeled data.

Supervised feature clustering algorithm usually performs clustering analysis over feature-class matrix, where each entry (i, j) in this matrix is the number of times of the i -th feature co-occurring with the j -th class. Therefore it can not group features in $F_{\bar{L}}$ into meaningful clusters since there are no class labels associated with these features. We overcome this problem by firstly inducing class labels for unseen features based on the similarity among features in F_L and $F_{\bar{L}}$, then clustering all the features (including F_L and $F_{\bar{L}}$) based on the distribution of class

labels over them.

This semi-supervised feature clustering algorithm is defined as follows:

Input:

Feature set $F = F_L \cup F_{\bar{L}}$ (the first $|F_L|$ features in F belong to F_L , and the remaining $|F_{\bar{L}}|$ features belong to $F_{\bar{L}}$), context set X , the label information of $x_g (1 \leq g \leq l)$, $N_{\tilde{F}}$ (the number of clusters in \tilde{F});

Output:

Clustering solution \tilde{F} ;

Algorithm:

1. Construct $|F| \times |X|$ feature-example matrix $M^{F,X}$, where entry $M_{i,j}^{F,X}$ is the number of times of f_i co-occurring with example $x_j (1 \leq j \leq n)$.

2. Form $|F| \times |F|$ affinity matrix W defined by $W_{ij} = \exp(-\frac{d_{ij}^2}{\sigma^2})$ if $i \neq j$ and $W_{ii} = 0 (1 \leq i, j \leq |F|)$, where d_{ij} is the distance (ex. Euclidean distance) between f_i (the i -th row in $M^{F,X}$) and f_j (the j -th row in $M^{F,X}$), and σ is used to control the weight W_{ij} .

3. Construct $|F_L| \times |S|$ feature-class matrix $Y^{F_L,S}$, where the entry $Y_{i,j}^{F_L,S}$ is the number of times of feature $f_i (f_i \in F_L)$ co-occurring with sense s_j .

4. Obtain hard label matrix for features in F_L (denoted as $Y_{hard}^{F_L,S}$) based on $Y^{F_L,S}$, where entry $Y_{hard,i,j}^{F,S} = 1$ if the hard label of f_i is s_j , otherwise zero. Obtain hard labels for features in $F_{\bar{L}}$ using a classifier based on W and $Y_{hard}^{F_L,S}$. In this paper we use label propagation (LP) algorithm (Zhu and Ghahramani, 2002) to get hard labels for $F_{\bar{L}}$.

5. Construct $|F| \times |S|$ feature-class matrix $Y_{hard}^{F,S}$, where entry $Y_{hard,i,j}^{F,S} = 1$ if the hard label of f_i is s_j , otherwise zero.

6. Construct the matrix $L = D^{-1/2}WD^{-1/2}$ in which D is a diagonal matrix with its (i, i) -element equal to the sum of the i -th row of W .

7. Label each feature in F as soft label $\hat{Y}_i^{F,S}$, the i -th row of $\hat{Y}^{F,S}$, where $\hat{Y}^{F,S} = (I - \alpha L)^{-1}Y_{hard}^{F,S}$.

8. Obtain the feature clustering solution \tilde{F} by clustering the rows of $\hat{Y}_i^{F,S}$ into $N_{\tilde{F}}$ groups. In this paper we use sequential information bottleneck (sIB) algorithm (Slonim and Tishby, 2000) to perform clustering analysis.

End

Step 3 \sim 5 are the process to obtain hard la-

els for features in F , while the operation in step 6 and 7 is a local and global consistency based semi-supervised learning (LGC) algorithm (Zhou et al., 2003) that smooth the classification result of LP algorithm to acquire a soft label for each feature.

At first sight, this semi-supervised feature clustering algorithm seems to make little sense. Since we run feature clustering in step 8, why not use LP algorithm to obtain soft label matrix $Y^{F_{\bar{L}},S}$ for features in $F_{\bar{L}}$ by the use of $Y^{F_L,S}$ and W , then just apply sIB directly to soft label matrix $\hat{Y}^{F,S}$ (constructed by concatenating $Y^{F_L,S}$ and $Y^{F_{\bar{L}},S}$)?

The reason for using LGC algorithm to acquire soft labels for features in F is that in the context of transductive learning, the size of labeled data is rather small, which is much less than that of unlabeled data. This makes it difficult to obtain reliable estimation of class label's distribution over features from only labeled data. This motivates us to use raw information (hard labels of features in F_L) from labeled data to estimate hard labels of features in $F_{\bar{L}}$. Then LGC algorithm is used to smooth the classification result of LP algorithm based on the assumption that a good classification should change slowly on the coherent structure aggregated by a large amount of unlabeled data. This operation makes our algorithm more robust to the noise in feature-class matrix $Y^{F_L,S}$ that is estimated from labeled data.

In this paper, σ is set as the average distance between labeled examples from different classes, and $N_{\tilde{F}} = |F|/10$. Latent semantic indexing technique (LSI) is used to perform factor analysis in $M^{F,X}$ before calculating the distance between features in step 2.

4 Experiments and Results

4.1 Experiment Design

For empirical study of dimensionality reduction techniques on WSD task, we evaluated five dimensionality reduction algorithms on the data in English lexical sample (ELS) task of SENSEVAL-3 (Mihalcea et al., 2004)(including all the 57 English words)¹: supervised feature clustering (SuFC) (Baker and McCallum, 1998; Bekkerman et al., 2003; Slonim

¹Available at <http://www.senseval.org/senseval3>

and Tishby, 2001), iterative double clustering (IDC) (El-Yaniv and Souroujon, 2001), semi-supervised feature clustering (SemiFC) (our algorithm), supervised feature selection (SuFS) (Forman, 2003), and latent semantic indexing (LSI) (Deerwester et. al., 1990)².

We used *sIB* algorithm³ to cluster features in F_L into groups based on the distribution of class labels associated with each feature. This procedure can be considered as our re-implementation of supervised feature clustering. After feature clustering, examples can be represented as vectors over feature clusters.

IDC is an extension of double clustering method (DC) (Slonim and Tishby, 2000), which performs iterations of DC. In the transductive version of IDC, they cluster features in F as distributions over class labels (given by the labeled data) during the first stage of the IDC first iteration. This phase results in feature clusters \tilde{F} . Then they continue as usual; that is, in the second phase of the first IDC iteration they group X into $N_{\tilde{X}}$ clusters, where X is represented as distribution over \tilde{F} . Subsequent IDC iterations use all the unlabeled data. This IDC algorithm can result in two clustering solutions: \tilde{F} and \tilde{X} . Following (El-Yaniv and Souroujon, 2001), the number of iterations is set as 15, and $N_{\tilde{X}} = |S|$ (the number of senses of target word) in our re-implementation of IDC. After performing IDC, examples can be represented as vectors over feature clusters \tilde{F} .

Supervised feature selection has been extensively studied for text categorization task (Forman, 2003). Information gain (IG) is one of state of the art criteria for feature selection, which measures the decrease in entropy when the feature is given vs. absent. In this paper, we calculate IG score for each feature in F_L , then select top $|F|/10$ features with highest scores to form reduced feature set. Then examples can be represented as vectors over the reduced feature set.

LSI is an unsupervised factor analysis technique based on Singular Value Decomposition of a $|X| \times |F|$ example-feature matrix. The underlying technique for LSI is to find an orthogonal basis for the

²Following (Baker and McCallum, 1998), we use LSI as a representative method for unsupervised dimensionality reduction.

³Available at <http://www.cs.huji.ac.il/~noamm/>

feature-example space for which the axes lie along the dimensions of maximum variance. After using LSI on the example-feature matrix, we can get vector representation for each example in X in reduced feature space.

For each ambiguous word in ELS task of SENSEVAL-3, we used three types of features to capture contextual information: part-of-speech of neighboring words with position information, unordered single words in topical context, and local collocations (as same as the feature set used in (Lee and Ng, 2002) except that we did not use syntactic relations). We removed the features with occurrence frequency (counted in both training set and test set) less than 3 times.

We ran these five algorithms for each ambiguous word to reduce the dimensionality of feature space from $|F|$ to $|F|/10$ no matter which training data is used (ex. full SENSEVAL-3 training data or sampled SENSEVAL-3 training data). Then we can obtain new vector representation of X in new feature space acquired by SuFC, IDC, SemiFC, and LSI or reduced feature set by SuFS.

Then we used SVM⁴ and LP algorithm to perform sense disambiguation on vectors in dimensionality reduced feature space. SVM and LP were evaluated using accuracy⁵ (fine-grained score) on test set of SENSEVAL-3. For LP algorithm, the test set in SENSEVAL-3 data was also used as unlabeled data in transductive learning process.

We investigated two distance measures for LP: cosine similarity and Jensen-Shannon (JS) divergence (Lin, 1991). Cosine similarity measures the angle between two feature vectors, while JS divergence measures the distance between two probability distributions if each feature vector is considered as probability distribution over features.

For sense disambiguation on SENSEVAL-3 data, we constructed connected graphs for LP algorithm following (Niu et al., 2005): two instances u, v will be connected by an edge if u is among v 's k nearest neighbors, or if v is among u 's k nearest neighbors

⁴We used *SVM^{light}* with linear kernel function, available at <http://svmlight.joachims.org/>.

⁵If there are multiple sense tags for an instance in training set or test set, then only the first tag is considered as correct answer. Furthermore, if the answer of the instance in test set is "U", then this instance will be removed from test set.

as measured by cosine or JS distance measure. k is 5 in later experiments.

4.2 Experiments on Full SENSEVAL-3 Data

In this experiment, we took the training set in SENSEVAL-3 as labeled data, and the test set as unlabeled data. In other words, all of dimensionality reduction methods and classifiers can use the label information in training set, but can not access the label information in test set. We evaluated different sense disambiguation processes using test set in SENSEVAL-3.

We use features with occurrence frequency no less than 3 in training set and test set as feature set F for each ambiguous word. F consists of two disjoint subsets: F_L and $F_{\bar{L}}$. F_L consists of features occurring in training set of target word in SENSEVAL-3, while $F_{\bar{L}}$ consists of features that occur in test set, but never appear in training set.

Table 1 lists accuracies of SVM and LP without or with dimensionality reduction on full SENSEVAL-3 data. From this table, we have some findings as follows:

(1) If without dimensionality reduction, the best performance of sense disambiguation is 70.3% (LP_{JS}), while if using dimensionality reduction, the best two systems can achieve 69.8% ($SuFS + LP_{JS}$) and 69.0% ($SemiFC + LP_{JS}$) accuracies. It seems that feature selection and feature clustering can significantly reduce the dimensionality of feature space while losing only about 1.0% accuracy.

(2) Furthermore, LP_{JS} algorithm performs better than SVM when combined with the same dimensionality reduction technique (except IDC). Notice that LP algorithm uses unlabelled data during its disambiguation phase while SVM doesn't. This indicates that using unlabeled data helps to improve the performance of sense disambiguation.

(3) When using LP algorithm for sense disambiguation, SemiFC performs better than other feature clustering algorithms, such as SuFC, IDC. This indicates that clustering seen and unseen features can satisfy the requirement of semi-supervised learning algorithm, which does help the classification process.

(4) When using SuFC, IDC, SuFS, or SemiFC for dimensionality reduction, the performance of sense disambiguation is always better than that using LSI

as dimensionality reduction method. SuFC, IDC, SuFS, and SemiFC use label information to guide feature clustering or feature selection, while LSI is an unsupervised factor analysis method that can conduct dimensionality reduction without the use of label information from labeled data. This indicates that using label information in dimensionality reduction procedure can cluster features into better groups or select better feature subsets, which results in better representation of contexts in reduced feature space.

4.3 Additional Experiments on Sampled SENSEVAL-3 Data

For investigating the performance of various dimensionality reduction techniques with very small training data, we ran them with only l_w examples from training set of each word in SENSEVAL-3 as labeled data. The remaining training examples and all the test examples were used as unlabeled data for SemiFC or LP algorithm. Finally we evaluated different sense disambiguation processes using test set in SENSEVAL-3. For each labeled set size l_w , we performed 20 trials. In each trial, we randomly sampled l_w labeled examples for each word from training set. If any sense was absent from the sampled labeled set, we redid the sampling. l_w is set as $N_{w,train} \times 10\%$, where $N_{w,train}$ is the number of examples in training set of word w . Other settings of this experiment is as same as that of previous one in section 4.2.

In this experiment, feature set F is as same as that in section 4.2. F_L consists of features occurring in sampled training set of target word in SENSEVAL-3, while $F_{\bar{L}}$ consists of features that occur in unlabeled data (including unselected training data and all the test set), but never appear in labeled data (sampled training set).

Table 2 lists accuracies of SVM and LP without or with dimensionality reduction on sampled SENSEVAL-3 training data ⁶. From this table, we have some findings as follows:

(1) If without dimensionality reduction, the best performance of sense disambiguation is 54.9% (LP_{JS}), while if using dimensionality reduction, the

⁶We can not obtain the results of IDC over 20 trials since it costs about 50 hours for each trial (Pentium 1.4 GHz CPU/1.0 GB memory).

Table 1: This table lists the accuracies of SVM and LP without or with dimensionality reduction on full SENSEVAL-3 data. There is no result for $LSI + LP_{JS}$, since the vectors obtained by LSI may contain negative values, which prohibits the application of JS divergence for measuring the distance between these vectors.

Classifier	Without dimensionality reduction	With various dimensionality reduction techniques				
		SuFC	IDC	SuFS	LSI	SemiFC
SVM	69.7%	66.4%	65.1%	65.2%	59.1%	64.0%
LP_{cosine}	68.4%	66.7%	64.9%	66.0%	60.7%	67.6%
LP_{JS}	70.3%	67.2%	64.0%	69.8%	-	69.0%

Table 2: This table lists the accuracies of SVM and LP without or with dimensionality reduction on sampled SENSEVAL-3 training data. For each classifier, we performed paired t-test between the system using SemiFC for dimensionality reduction and any other system with or without dimensionality reduction. \gg (or \ll) means p-value ≤ 0.01 , while $>$ (or $<$) means p-value falling into $(0.01, 0.05]$. Both \gg (or \ll) and $>$ (or $<$) indicate that the performance of current WSD system is significantly better (or worse) than that using SemiFC for dimensionality reduction, when given same classifier.

Classifier	Without dimensionality reduction	With various dimensionality reduction techniques			
		SuFC	SuFS	LSI	SemiFC
SVM	53.4 \pm 1.1% (\gg)	50.4 \pm 1.1% (\ll)	52.2 \pm 1.2% ($>$)	49.8 \pm 0.8% (\ll)	51.5 \pm 1.0%
LP_{cosine}	54.4 \pm 1.2% (\gg)	49.5 \pm 1.1% (\ll)	51.1 \pm 1.0% (\ll)	49.8 \pm 1.0% (\ll)	52.9 \pm 1.0%
LP_{JS}	54.9 \pm 1.1% (\gg)	52.0 \pm 0.9% (\ll)	52.5 \pm 1.0% (\ll)	-	54.1 \pm 1.2%

best performance of sense disambiguation is 54.1% ($SemiFC + LP_{JS}$). Feature clustering can significantly reduce the dimensionality of feature space while losing only 0.8% accuracy.

(2) LP_{JS} algorithm performs better than SVM when combined with most of dimensionality reduction techniques. This result confirmed our previous conclusion that using unlabeled data can improve the sense disambiguation process. Furthermore, SemiFC performs significantly better than SuFC and SuFS when using LP as the classifier for sense disambiguation. The reason is that when given very few labeled examples, the distribution of class labels over features can not be reliably estimated, which deteriorates the performance of SuFC or SuFS. But SemiFC uses only raw label information (hard label of each feature) estimated from labeled data, which makes it robust to the noise in very small labeled data.

(3) SuFC, SuFS and SemiFC perform better than LSI no matter which classifier is used for sense dis-

ambiguation. This observation confirmed our previous conclusion that using label information to guide dimensionality reduction process can result in better representation of contexts in feature subspace, which further improves the results of sense disambiguation.

5 Related Work

Feature clustering has been extensively studied for the benefit of text categorization and document clustering, which can be categorized as supervised feature clustering, semi-supervised feature clustering, and unsupervised feature clustering.

Supervised feature clustering algorithms (Baker and McCallum, 1998; Bekkerman et al., 2003; Slonim and Tishby, 2001) usually cluster words into groups based on the distribution of class labels over features. Baker and McCallum (1998) apply supervised feature clustering based on distributional clustering for text categorization, which can compress the feature space much more aggressively while still

maintaining state of the art classification accuracy. Slonim and Tishby (2001) and Bekkerman et. al. (2003) apply information bottleneck method to find word clusters. They present similar results with the work by Baker and McCallum (1998). Slonim and Tishby (2001) goes further to show that when the training sample is small, word clusters can yield significant improvement in classification accuracy.

Unsupervised feature clustering algorithms (Dhillon, 2001; Dhillon et al., 2002; Dhillon et al., 2003; El-Yaniv and Souroujon, 2001; Slonim and Tishby, 2000) perform word clustering by the use of word-document co-occurrence matrix, which do not utilize class labels to guide clustering process. Slonim and Tishby (2000), El-Yaniv and Souroujon (2001) and Dhillon et. al. (2003) show that word clusters can improve the performance of document clustering.

El-Yaniv and Souroujon (2001) present an iterative double clustering (IDC) algorithm, which performs iterations of double clustering (Slonim and Tishby, 2000). Furthermore, they extend IDC algorithm for semi-supervised learning when given both labeled and unlabeled data.

Our algorithm belongs to the family of semi-supervised feature clustering techniques, which can utilize both labeled and unlabeled data to perform feature clustering.

Supervised feature clustering can not group unseen features (features that do not occur in labeled data) into meaningful clusters since there are no class labels associated with these unseen features. Our algorithm can overcome this problem by inducing class labels for unseen features based on the similarity among seen features and unseen features, then clustering all the features (including both seen features and unseen features) based on the distribution of class labels over them.

Compared with the semi-supervised version of IDC algorithm, our algorithm is more efficient, since we perform feature clustering without iterations.

The difference between our algorithm and unsupervised feature clustering is that our algorithm depends on both labeled and unlabeled data, but unsupervised feature clustering requires only unlabeled data.

O'Hara et. al. (2004) use semantic class-based collocations to augment traditional word-

based collocations for supervised WSD. Three separate sources of word relatedness are used for these collocations: 1) WordNet hypernym relations; 2) cluster-based word similarity classes; and 3) dictionary definition analysis. Their system achieved 56.6% fine-grained score on ELS task of SENSEVAL-3. In contrast with their work, our data-driven method for feature clustering based WSD does not require external knowledge resource. Furthermore, our *SemiFC+LPJS* method can achieve 69.0% fine-grained score on the same dataset, which shows the effectiveness of our method.

6 Conclusion

In this paper we have investigated feature clustering techniques for WSD, which usually group features into clusters based on the distribution of class labels over features. We propose a semi-supervised feature clustering algorithm to satisfy the requirement of semi-supervised classification algorithms for dimensionality reduction in feature space. Our experimental results on SENSEVAL-3 data show that feature clustering can aggressively reduce the dimensionality of feature space while still maintaining state of the art sense disambiguation accuracy. Furthermore, when combined with a semi-supervised WSD algorithm, semi-supervised feature clustering outperforms supervised feature clustering and other dimensionality reduction techniques. Our additional experiments on sampled SENSEVAL-3 data indicate that our semi-supervised feature clustering method is robust to the noise in small labeled data, which achieves better performance than supervised feature clustering.

In the future, we may extend our work by using more datasets to empirically evaluate this feature clustering algorithm. This semi-supervised feature clustering framework is quite general, which can be applied to other NLP tasks, ex. text categorization.

Acknowledgements We would like to thank anonymous reviewers for their helpful comments. Z.Y. Niu is supported by A*STAR Graduate Scholarship.

References

Baker L. & McCallum A.. 1998. Distributional Clustering of Words for Text Classification. *ACM SIGIR*

- 1998.
- Bekkerman, R., El-Yaniv, R., Tishby, N., & Winter, Y.. 2003. Distributional Word Clusters vs. Words for Text Categorization. *Journal of Machine Learning Research*, Vol. 3: 1183-1208.
- Dagan, I. & Itai A.. 1994. Word Sense Disambiguation Using A Second Language Monolingual Corpus. *Computational Linguistics*, Vol. 20(4), pp. 563-596.
- Deerwester, S.C., Dumais, S.T., Landauer, T.K., Furnas, G.W., & Harshman, R.A.. 1990. Indexing by Latent Semantic Analysis. *Journal of the American Society of Information Science*, Vol. 41(6), pp. 391-407.
- Dhillon I.. 2001. Co-Clustering Documents and Words Using Bipartite Spectral Graph Partitioning. *ACM SIGKDD 2001*.
- Dhillon I., Mallela S., & Kumar R.. 2002. Enhanced Word Clustering for Hierarchical Text Classification. *ACM SIGKDD 2002*.
- Dhillon I., Mallela S., & Modha, D.. 2003. Information-Theoretic Co-Clustering. *ACM SIGKDD 2003*.
- El-Yaniv, R., & Souroujon, O.. 2001. Iterative Double Clustering for Unsupervised and Semi-Supervised Learning. *NIPS 2001*.
- Forman, G.. 2003. An Extensive Empirical Study of Feature Selection Metrics for Text Classification. *Journal of Machine Learning Research* 3(Mar):1289-1305.
- Leacock, C., Miller, G.A. & Chodorow, M.. 1998. Using Corpus Statistics and WordNet Relations for Sense Identification. *Computational Linguistics*, 24:1, 147-165.
- Lee, Y.K. & Ng, H.T.. 2002. An Empirical Evaluation of Knowledge Sources and Learning Algorithms for Word Sense Disambiguation. *EMNLP 2002*, (pp. 41-48).
- Lesk M.. 1986. Automated Word Sense Disambiguation Using Machine Readable Dictionaries: How to Tell a Pine Cone from an Ice Cream Cone. *ACM SIGDOC 1986*.
- Li, H. & Li, C.. 2004. Word Translation Disambiguation Using Bilingual Bootstrapping. *Computational Linguistics*, 30(1), 1-22.
- Lin, J. 1991. Divergence Measures Based on the Shannon Entropy. *IEEE Transactions on Information Theory*, 37:1, 145-150.
- McCarthy, D., Koeling, R., Weeds, J., & Carroll, J.. 2004. Finding Predominant Word Senses in Untagged Text. *ACL 2004*.
- Mihalcea R.. 2002. Instance Based Learning with Automatic Feature Selection Applied to Word Sense Disambiguation. *COLING 2002*.
- Mihalcea R.. 2004. Co-Training and Self-Training for Word Sense Disambiguation. *CoNLL 2004*.
- Mihalcea R., Chklovski, T., & Kilgariff, A.. 2004. The SENSEVAL-3 English Lexical Sample Task. *SENSEVAL 2004*.
- Niu, Z.Y., Ji, D.H., & Tan, C.L.. 2004. Learning Word Senses With Feature Selection and Order Identification Capabilities. *ACL 2004*.
- Niu, Z.Y., Ji, D.H., & Tan, C.L.. 2005. Word Sense Disambiguation Using Label Propagation Based Semi-Supervised Learning. *ACL 2005*.
- O'Hara, T., Bruce, R., Donner, J., & Wiebe, J.. 2004. Class-Based Collocations for Word-Sense Disambiguation. *SENSEVAL 2004*.
- Park, S.B., Zhang, B.T., & Kim, Y.T.. 2000. Word Sense Disambiguation by Learning from Unlabeled Data. *ACL 2000*.
- Pedersen, T., & Bruce, R.. 1997. Distinguishing Word Senses in Untagged Text. *EMNLP 1997*.
- Schütze, H.. 1998. Automatic Word Sense Discrimination. *Computational Linguistics*, 24:1, 97-123.
- Slonim, N. & Tishby, N.. 2000. Document Clustering Using Word Clusters via the Information Bottleneck Method. *ACM SIGIR 2000*.
- Slonim, N. & Tishby, N.. 2001. The Power of Word Clusters for Text Classification. *The 23rd European Colloquium on Information Retrieval Research*.
- Towel, G. & Voorheest, E.M.. 1998. Disambiguating Highly Ambiguous Words. *Computational Linguistics*, 24:1, 125-145.
- Yarowsky, D.. 1995. Unsupervised Word Sense Disambiguation Rivaling Supervised Methods. *ACL 1995*, pp. 189-196.
- Zhou D., Bousquet, O., Lal, T.N., Weston, J., & Schölkopf, B.. 2003. Learning with Local and Global Consistency. *NIPS 16*, pp. 321-328.
- Zhu, X. & Ghahramani, Z.. 2002. Learning from Labeled and Unlabeled Data with Label Propagation. *CMU CALD tech report CMU-CALD-02-107*.
- Zhu, X., Ghahramani, Z., & Lafferty, J.. 2003. Semi-Supervised Learning Using Gaussian Fields and Harmonic Functions. *ICML 2003*.