SEMANTIC CLASSIFICATION OF UNKNOWN WORDS BASED ON GRAPH-BASED SEMI-SUPERVISED CLUSTERING

Fumiyo Fukumoto and Yoshimi Suzuki
Interdisciplinary Graduate School of Medicine and Engineering, University of Yamanashi, Yamanashi, Japan

Keywords: Unknown words, Polysemous verbs, Levin-style semantic classes, Semi-supervised clustering.

Abstract: This paper presents a method for semantic classification of unknown verbs including polysemies into Levin-style semantic classes. We propose a semi-supervised clustering, which is based on a graph-based unsupervised clustering technique. The algorithm detects the spin configuration that minimizes the energy of the spin glass. Comparing global and local minima of an energy function, called the Hamiltonian, allows for the detection of nodes with more than one cluster. We extended the algorithm so as to employ a small amount of labeled data to aid unsupervised learning, and applied the algorithm to cluster verbs including polysemies. The distributional similarity between verbs used to calculate the Hamiltonian is in the form of probability distributions over verb frames. The result obtained using 110 test polysemous verbs with labeled data of 10% showed 0.577 F-score.

1 INTRODUCTION

Semantic verb classification is not an end task in itself, but supports many NLP tasks, such as subcategorization acquisition (Korhonen, 2002; Kermanidis et al., 2008), word sense disambiguation (Navigli, 2009), and language generation (Reiter and Dale, 2000). Much of the previous work on verb classification has been to classify verbs into classes with semantically similar senses taken from an existing thesaurus or taxonomy. However, such a resource makes it nearly impossible to cover large, and fast-changing linguistic knowledge required for these NLP tasks, depending on text-type and subject domain. Let us take a look at the Levin-style semantic classes (Levin, 1993). It consists of 3,024 verbs. Similarly, Japanese thesaurus dictionary called Bunrui-Goi-Hyo consists of 87,743 content words. Therefore, considering this resource scarcity problem, semantic classification of verbs which do not appear in the resource but appear in corpora has been an interest since the earliest days when a number of large scale corpora have become available.

A number of methodologies have been developed for verb classification. One such attempt is to apply clustering techniques to classification. However, two main difficulties arise in the use of clustering algorithms. The first is that we do not know how many classes there are in a given input. The usual drawback in many algorithms is that they cannot give a valid criterion for measuring class structure. The second is that the algorithm should allow each data point (verb) to belong to more than one cluster because of the existence of polysemous verbs.

The aim of this work is to resolve these problems. We focus on unknown verbs including polysemies, and present a method for classifying them into Levin-style semantic classes. We propose a graph-based semi-supervised clustering method which allows nodes (verbs) to belong to multiple clusters (senses). The essence of the approach is to define an energy function, called the Hamiltonian which achieves minimal energy when there is high within-cluster connectivity and low between-cluster connectivity. The energy minimum is obtained by simulated annealing. In this context, two verbs are "connected" if they share many of the same subcategorization frames. Comparing global and local minima of an energy function Hamiltonian, allows for the detection of overlapping nodes. We extended the algorithm so as to employ a small amount of labeled data to aid unsupervised learning, and clustered polysemous verbs. The distributional similarity between verbs used to calculate the Hamiltonian is in the form of probability distributions over verb frames. The results obtained using 110 test verbs including polysemies with labeled data of 10% showed 0.577 F-score, and it was comparable with previous work.

Fukumoto F. and Suzuki Y..
SEMANTIC CLASSIFICATION OF UNKNOWN WORDS BASED ON GRAPH-BASED SEMI-SUPERVISED CLUSTERING.
DOI: 10.5220/0003633100370046
In *Proceedings of the International Conference on Knowledge Engineering and Ontology Development* (KEOD-2011), pages 37-46
ISBN: 978-989-8425-80-5

The rest of the paper is organized as follows. The next section provides an overview of existing techniques. Section 3 explains verb description, *i.e.*, verb frame patterns. Section 4 describes distributional similarity measures to compute semantic similarities between pairs of verbs. Section 5 explains our clustering algorithm. Finally, we report some experiments using 110 verbs including polysemies, and end with a discussion of evaluation.

2 RELATED WORK

Much of the previous work on verb classification is to classify verbs into classes with semantically similar senses taken from an existing thesaurus or taxonomy. One attractive attempt is to use Levin-style semantic classes (Levin, 1993), as this classification includes the largest number of English verbs with fine-grained classes. Moreover, it is based on the assumption that the sense of a verb influences its syntactic behavior, particularly with respect to the choice of its arguments. Therefore, if we induce a verb classification on the basis of verb features, *i.e.*, syntactic information obtained from corpora, then the resulting classification should agree with a semantic classification to a certain extent.

Schulte (Schulte im Walde, 2000) attempted to classify verbs using two algorithms: iterative clustering based on a definition by (Hughes, 1994), and unsupervised latent class analysis as described by (Rooth, 1998), based on the expectation maximization algorithm. Stevenson and Joanis compared their supervised method for verb classification with semi-supervised and unsupervised techniques (Stevenson and Joanis, 2003). Brew *et al.* focused on dimensionality reduction on the verb frame patterns, and applied a spectral clustering technique (Ng et al., 2002) to the unsupervised clustering of German verbs to Levin's English classes (Brew and Walde, 2002). They reported that the results by a spectral clustering outperformed the standard *k*-means against all the evaluation measures including "F-measure" and all the distance measures including "skew divergence."

In the context of graph-based clustering of words, Widdows and Dorow used a graph model for unsupervised lexical acquisition (Widdows and Dorow, 2002). The graph structure is built by linking pairs of words that participate in particular syntactic relationships. An incremental cluster-building algorithm using the graph structure achieved 82% accuracy at a lexical acquisition task, evaluated against WordNet 10 classes, and each class consists of 20 words. Matsuo *et al.* proposed a method of word clustering based on a word similarity measure by Web counts (Matsuo et al., 2006). They used *Newman* clustering for the clustering algorithm, and reported that the results obtained with the algorithm were better than those obtained by average-link agglomerative clustering using 90 Japanese noun words. However, all these methods relied on hard-clustering models, and thus have largely ignored the issue of polysemy by assuming that words belong to only one cluster.

In contrast to hard-clustering algorithms, soft clustering allows that words to belong to more than one cluster. Much of the previous work on soft clustering is based on EM algorithm. The earliest work in this direction is that of Pereira *et al* (Pereira et al., 1993), who described a hierarchical soft clustering method that clusters noun words. The clustering result was a hierarchy of noun clusters, where every noun belongs to every cluster with a membership probability. The initial data for the clustering process were frequencies of verb–noun pairs in a direct object relationship, as extracted from conditional verb–noun probabilities, the similarity of the distributions was determined by the KL divergence. The EM algorithm was used to learn the hidden cluster membership probabilities, and deterministic annealing performed the divisive hierarchical clustering. Schulte *et al.* (Schulte im Walde et al., 2008) proposed a method for semantic verb classification that relies on selectional preferences as verb properties. The model was implemented as a soft clustering approach to capture the polysemy of the verbs. The training procedure used the EM algorithm to iteratively improve the probabilistic parameters of the model, and applied the MDL principle to induce WordNet-based selectional preferences for arguments within subcategorization frames. The results showed that after 10 training iterations the verb class model results were above the baseline results. Our work is similar to their method in the use of semi-supervised clustering, while they did not report in detail whether the clusters captured polysemic verbs. Moreover, the algorithm cannot assign unlabeled data to a new class other than known classes.

Korhonen *et al.* (Korhonen et al., 2003) used verb–frame pairs to cluster verbs relying on the information bottleneck. Our work is similar to their method in the use of 110 test verbs provided by (Korhonen et al., 2003), and focused especially on verbal polysemy. However, their method interpreted polysemy as represented by the soft clusters, *i.e.*, they used the Information Bottleneck, an iterative soft method with hardening of the output, while the method presented in this paper allows that verbs belong to more than one cluster.

3 SUBCATEGORIZATION INFORMATION

A typical word clustering task is to cluster words into classes based on their distributional similarity. Similarity measures based on distributional hypothesis compare a pair of weighted feature vectors that characterize two words (Hindle, 1990; Lin, 1998; Dagan et al., 1999).

Like much previous work on verb classification, we used subcategorization frame distributions to calculate similarity between verbs (Schulte im Walde, 2000; Brew and Walde, 2002). More precisely, (Korhonen et al., 2003) provided subcategorization frame data. They used the subcategorization acquisition systems of (Briscoe and Carroll, 1997). The system employs a robust statistical parser (Briscoe and Carroll, 2002), which yields complete but shallow parses, and a comprehensive subcategorization frame classifier. It incorporates 163 subcategorization distinctions, a set of those found in the ANLT and COMLEX dictionaries (Bouraev et al., 1987; Grishman et al., 1994). A total of 6,433 verbs were first selected from COMLEX and British National Corpus (Leech, 1992). Next, to obtain as comprehensive subcategorization frequency information as possible, up to 10,000 sentences containing an occurrence of each of these verbs were included in the input data for subcategorization acquisition. These sentences were extracted from five different corpora, including BNC (Korhonen et al., 2006). We used these data to calculate similarity between verbs.

4 DISTRIBUTIONAL SIMILARITY

There is a large body of work on distributional similarity measures. Here, we concentrate on eight more commonly used measures. In the following formulae, *x* and *y* refer to the verb vectors, their subscripts to the verb subcategorization frame values.

1. **The Binary Cosine Measure (bCos).**
   The cosine measures the similarity of the two vectors *x* and *y* by calculating the cosine of the angle between vectors, where each dimension of the vector corresponds to each of 163 subcategorization patterns and each value of the dimension is the frequency of each pattern. The binary cosine measure is a flattened version of the cosine measure in which all non-zero counts are replaced by 1.0.
2. **The Cosine Measure based on Probability of Relative Frequencies (rfCos).**
   The differences between the cosine and the value based on relative frequencies of subcategorization frames are the values of each dimension, *i.e.*, the former are frequencies of each pattern and the latter are the probability of relative frequencies of each pattern.
3. **The Dice Coefficient (Dice).**
   The Dice Coefficient is a combinatorial similarity measure adopted from the field of Information Retrieval for use as a measure of lexical distributional similarity. It is computed as twice the ratio between size of the intersection of the two subcategorization patterns and the sum of the sizes of the individual subcategorization patterns:

$$Dice(x,y) = \frac{2\cdot|F(x)\cap F(y)|}{|F(x)|+|F(y)|}.$$

4. **Jaccard's Coefficient (Jacc).**
   Jaccard's Coefficient can be defined as the ratio between the size of the intersection of the two subcategorization patterns and the size of the union of the subcategorization patterns:

$$Jacc(x,y) = \frac{|F(x)\cap F(y)|}{|F(x)\cup F(y)|}.$$

5. **L_1 Norm (L_1)**
   The L_1 Norm is a member of a family of measures known as the Minkowski Distance, for measuring the distance between two points in space. The L_1 distance between two verbs can be written as:

$$L_1(x,y) = \sum_{i=1}^{n}|x_i-y_i|.$$

6. **Kullback-Leibler (KL).**
   Kullback-Leibler is a measure from information theory that determines the inefficiency of assuming a model probability distribution given the true distribution.

$$D(x||y) = \sum_{i=1}^{n} x_i * \log\frac{x_i}{y_i}.$$

   KL is not defined in case $y_i = 0$. Thus, the probability distributions must be smoothed. We used two smoothing methods, *i.e.*, Add-one smoothing

and Witten and Bell smoothing (Witten and Bell, 1991).[1] Moreover, two variants of KL, α-skew divergence and the Jensen-Shannon, were used to perform smoothing.

7. **α-skew Divergence (α div.).**
   The α-skew divergence measure is a variant of KL, and is defined as:

$$\alpha div(x,y) \quad = \quad D(y \,||\, \alpha \cdot x + (1-\alpha) \cdot y).$$

   Lee reported the best results with α = 0.9 (Lee, 1999). We used the same value.

8. **The Jensen-Shannon (JS).**
   The Jensen-Shannon is a measure that relies on the assumption that if x and y are similar, they are close to their average. It is defined as:

$$JS(x,y) = \frac{1}{2}[D(x \,||\, \frac{x+y}{2}) + D(y \,||\, \frac{x+y}{2})].$$

All measures except bCos, rfCos, Dice, and Jacc showed that smaller values indicate a closer relation between two verbs. Thus, we used inverse of each value.

5 CLUSTERING METHOD

We now proceed to a discussion of our modifications to the algorithm reported by (Reichardt and Bornholdt, 2006); we call this semi-supervised RB algorithm. In this work, we focus on background knowledge that can be expressed as a set of constraints on the clustering process. After a discussion of the kind of constraints we are using, we describe semi-supervised RB algorithm.

5.1 The Constraints

In semi-supervised clustering, a small amount of labeled data is available to aid the clustering process. Like much previous work on semi-supervised clustering (Bar-Hillel et al., 2003; Bilenko et al., 2004), our work uses both must-link and cannot-link constraints between pairs of nodes (Wagstaff et al., 2001). Must-link constraints specify that two nodes (verbs) have to be in the same cluster. Cannot-link constraints, on the other hand, specify that two nodes must not be placed in the same cluster. These constraints are derived from a small amount of labeled data.

[1] We report Add-one smoothing results in the evaluation, as it was better than Witten and Bell smoothing.

5.2 Clustering Algorithm

The clustering algorithm used in this study was based on a graph-based unsupervised clustering technique reported by (Reichardt and Bornholdt, 2006). This algorithm detects the spin configuration that minimizes the energy of the spin glass. The energy function Hamiltonian, for assignment of nodes into communities clusters together those that are linked, and keeps separate those that are not by rewarding internal edges between nodes and penalizing existing edges between different clusters. Here, "community" or "cluster" have in common that they are groups of densely interconnected nodes that are only sparsely connected with the rest of the network. Only local information is used to update the nodes which makes parallelization of the algorithm straightforward and allows the application to very large networks. Moreover, comparing global and local minima of the energy function allows the detection of overlapping nodes. Reichardt *et al.* evaluated their method by applying several data, the college football network and a large protein folding network, and reported that the algorithm successfully detected overlapping nodes (Reichardt and Bornholdt, 2004). We extended the algorithm so as to employ a small amount of labeled data to aid unsupervised learning, and clustered polysemous verbs. Let v_i ($1 \leq i \leq n$) be a verb in the input, and σ_i be a label assigned to the cluster in which v_i is placed. The Hamiltonian is defined as:

$$H(\{\sigma_i\}) \quad = \quad -\sum_{i<j}(A_{ij}(\theta) - \gamma p_{ij})\delta_{\sigma_i\sigma_j}. \qquad (1)$$

Here, δ denotes the Kronecker delta. The function $A_{ij}(\theta)$ refers to the adjacency matrix of the graph. If both of the v_i and v_j are labeled data, it is defined by Eq. (2), otherwise it is defined by Eq. (3).

$$A_{ij}(\theta) = \begin{cases} 1 & \text{if } v_i \text{ and } v_j \text{ satisfy must-link} \\ 0 & \text{if } v_i \text{ and } v_j \text{ satisfy cannot-link.} \end{cases} \qquad (2)$$

$$A_{ij}(\theta) = \begin{cases} 1 & \text{if } sim(v_i, v_j) \geq \theta \\ 0 & \text{otherwise.} \end{cases} \qquad (3)$$

We calculated $sim(v_i, v_j)$, *i.e.*, similarity between v_i and v_j using one of the measures mentioned in Section 4. If θ is 0.9 for example, the value of the topmost 10% of the verb pairs are 1, and the remaining pairs are 0.

The matrix p_{ij} in Eq. (1) denotes the probability that a link exists between verb v_i and v_j, and is defined as:

$$p_{ij} = \sum_{i<j} \frac{A_{ij}(\theta)}{N(N-1)/2}, \quad (4)$$

where N in Eq. (4) denotes the number of verbs and $N(N-1)/2$ is the total number of verb pairs. As the parameter γ in Eq. (1) increases, each verb is distributed into larger number of clusters. Eq. (1) thus shows comparison of the actual values of internal or external edges with its respective expectation value under the assumption of equally probable links and given data sizes. The minima of the Hamiltonian H are obtained by simulated annealing (Kirkpatrick et al., 1983) as illustrated in Figure 1.

We applied the flow of the minima of the Hamiltonian shown in Figure 1 for M runs. We need to find a global minimum of the Hamiltonian. Each value of the $H_{\min}$ for M runs does not generally coincide with each other. Only the minimum among the values can be a global minimum and others are local minima. However, it often happens that one of the local minima is unexpectedly the minimum value, *i.e.*, a global minimum [2]. Thus, we regarded the minimum value which appears more than m times among the M results as the desired global minimum. We picked $H_{\min}$ and its corresponding all $\{\sigma_i\}_{\min}$. If a v_i belongs to more than two $\{\sigma_i\}_{\min}$, the v_i is regarded as a polysemous verb. The procedure of verb classification using the RB consists of four steps.

1. **Input.**
   The input is a set of verbs $\{v_1, \cdots, v_n\}$, where n is the number of input verbs.
2. **Calculation of Similarity.**
   Similarities for each pair of v_i are calculated by using measures mentioned in Section 4.
3. **Construction of Adjacency Matrix.**
   According to Eq. (2) and (3), adjacency matrix, $A_{ij}(\theta)$ is created.
4. **Running RB Algorithm.**
   The RB algorithm shown in Figure 1 is applied to the adjacency matrix, and clusters of verbs are obtained.

We note that the algorithm applies m times to find the minima of the Hamiltonian. Therefore, we parallelized the algorithm using the Message Passing Interface (MPI), as we applied simulated annealing for M runs. For implementation, we used a supercomputer, SPARC Enterprise M9000, 64 CPU, 1 TB memory.

[2]The method to obtain the $H_{\min}$ does not warrant the value to be an actual global minimum, as it is based on the Monte-Carlo way.

Input {
$A_{ij}(\theta)$ // The adjacency matrix.
γ
}

Output {
$(H_{\min}, \{\sigma_i\}_{\min})$
// The minimum of the H and its corresponding
// clusters.
}

Initialization {
1. Assign an initial random value to each $\{\sigma_i\}$.
2. Calculate $H_{\text{in}} := H(\{\sigma_i\})$.
3. $H := H_{\text{in}}$, $H_{\min} := H_{\text{in}}$.
4. $T_{\max} := 10 \times |H_{\text{in}}|$, $T_{\min} := 0.1 \times |H_{\text{in}}|$.
// T is a parameter called the temperature.
// $T_{\max}$ and $T_{\min}$ are the maximum and minimum
// value of T, respectively
}

Simulated annealing {
For $t = 1, 2, \cdots, N$ {
// N is the number of iteration
$T := T_{\max} - (T_{\max} - T_{\min}) \frac{(t-1)}{(N-1)}$
// Let T be linearly decreased.
For $i = 1, 2, \cdots, n$ {
1. Calculate the change $\triangle H(\sigma_i \to s)$ from the current value of H when σ_i is changed to be s as:
$\triangle H(\sigma_i \to s) = -\sum_{m(m \neq i)} (A_{im}(e) - \gamma p_{im}) \delta_{s\sigma_m}$.
2. Choose a random cluster label q as a candidate of σ_i, such that $q \neq \sigma_i$.
3. Calculate $p(\sigma_i = q)$ as:
$$p(\sigma_i = q) = \frac{\exp(-\frac{1}{T} \triangle H(\sigma_i \to q))}{\sum_s \exp(-\frac{1}{T} \triangle H(\sigma_i \to s))}.$$
// $p(\sigma_i = q)$ is a probability that a cluster
// label of σ_i is q.
4. If $r < p(\sigma_i = q)$, then σ_i and H are updated as follows:
$H := H + \triangle H(\sigma_i = q)$, $\sigma_i := q$
// r is a random value and $0 \leq r \leq 1$
5. If $H < H_{\min}$, then
$H_{\min} := H$, $\{\sigma_i\}_{\min} := \{\sigma_i\}$
}
}
}

Figure 1: Minima of the Hamiltonian by Simulated Annealing.

Table 1: Clustering results.

	θ	γ	*Sim*	C	Prec	Rec	F
RB	0.2	1.0	Dice	48	0.536	0.626	0.577
EM	–	–	–	59	0.301	0.512	0.387

Table 2: Results against each measure.

Sim	θ	γ	*C*	Prec	Rec	F
Cos	0.1	1.1	39	0.402	0.583	0.476
rfCos	0.1	1.0	39	0.396	0.565	0.466
Dice	0.2	1.0	48	0.536	0.626	0.577
Jacc	0.1	0.7	36	0.314	0.785	0.449
L_1	0.2	0.7	46	0.378	0.724	0.497
KL	0.1	0.9	38	0.411	0.630	0.497
αdiv.	0.1	1.2	39	0.421	0.634	0.506
JS	0.4	1.0	37	0.380	0.539	0.446
EM	–	–	59	0.301	0.512	0.387

6 EXPERIMENTS

6.1 Experimental Setup

We used the data consisting of 110 test verbs constructed by Korhonen *et al*. (Korhonen et al., 2003). There are two types: one is the monosemous gold standard, which lists only a single sense for each test verb corresponding to the most frequent sense in WordNet, and the other is the polysemous standard, which provides all senses for each verb. We used polysemous data in the experiments. We randomly selected 10% of verbs from 110 test verbs, and used them as a labeled data. The remaining verbs are used as unknown verbs. The selection of labeled data was repeated 10 times. All results are averaged performance against 10 trials. The similarity between verbs are calculated by using subcategorization frame data provided by (Korhonen et al., 2003). In the experiments, we experimentally set *m* and *M* in semi-supervised RB to 3 and 1,000, respectively.

For evaluation of verb classification, we used the precision, recall, and F-score, which were defined by (Schulte im Walde, 2000), especially to capture how many verbs does the algorithm actually detect more than just the predominant sense. Precision was defined by the percentage of verb senses appearing in the correct clusters compared to the number of verb senses appearing in any cluster, and recall was defined by the percentage of verb senses within the correct clusters compared to the total number of verb senses to be clustered.

For comparison, we utilized the EM algorithm which is widely used as a soft clustering and semi-supervised clustering technique (Schulte im Walde et al., 2008). We followed the method presented in (Rooth et al., 1999). We used a probability distribution over verb frames with selectional preferences, and used up to 30 iterations to learn the model probabilities.

6.2 Basic Results

The results are shown in Table 1. "γ" and "θ" refer to the parameters used by semi-supervised RB algorithm. "*Sim*" indicates similarity measure reported in Section 4. We performed experiments by varying these values. Table 1 denotes the value that maximized F-score. "*C*" refers to the number of clusters obtained by the method, and "EM" shows the results obtained by EM algorithm. Table 1 shows that the results obtained by semi-supervised RB algorithm were comparable to those obtained using the EM algorithm. We note that the number of clusters obtained by EM algorithm is the number of different labeled verbs in the test data, as the algorithm cannot assign unlabeled data to a new class other than known classes (labeled verbs). The result obtained by semi-supervised RB shows that three new classes on an average are correctly obtained in 10 trials, while the number of clusters is smaller than that obtained by EM and the correct clusters, 62 clusters.

Table 2 shows the results by using each similarity measure. We can see from Table 2 that our core finding, that unknown words including polysemy actually aids verb classification, was robust across a wide variety of distributional similarity measures, although the Dice coefficient was decidedly the best such measure for this particular problem. The observation indicates that the RB algorithm, especially the minima of the Hamiltonian, demonstrates our basic assumption: a verb which belongs to multiple clusters will in general reduce the energy.

We recall that semi-supervised RB algorithm uses two parameters: γ and θ. We examined how these parameters affect overall clustering results. Figure 2 shows F-score of 110 verbs plotted against γ value for the top approach, *i.e.*, by running RB with Dice coefficient as similarity measure. Similarly, Figure 3 shows F-score of 110 verbs plotted against θ value by running RB with Dice coefficient.

As can be seen clearly from Figure 2, the overall performance is extremely worse when the γ value is smaller than 0.9. This is because most of the clustering results show that verbs are classified into one of the clusters , while they are polysemous verbs. Similarly, the performance is worse when the γ value is

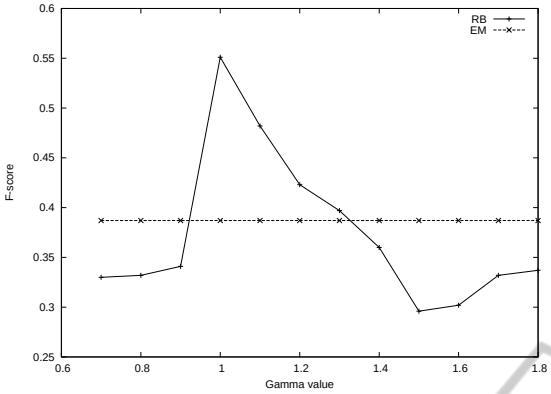

Figure 2: F-score against γ values.

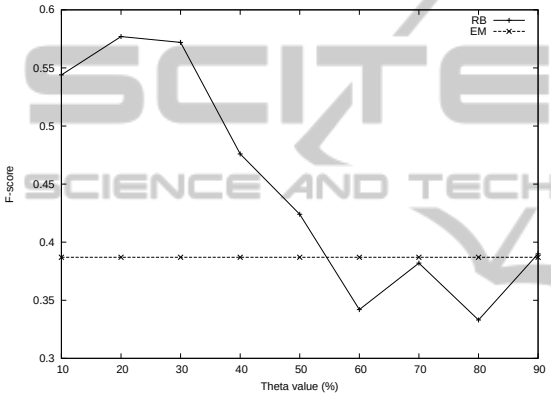

Figure 3: F-score against θ values.

larger than 1.3 because many of the clustering results show that verbs are classified into more than one clusters, while they are not polysemous verbs. This indicates that the RB with an extremely large or small value of γ is not effective for verb classification.

Figure 3 also shows the impact against the θ values on the effectiveness for clustering verbs. As shown in Figure 3, the larger value of θ affects overall performance. This is because the number of selected pairs of verbs is few. As a result, pairs are not chose, even if they are semantically related. The result supports the usefulness of Dice coefficient, as the best performance was obtained when the θ value was around 20%, then the performance is decreased when the value of θ is large.

6.3 Efficacy against Unknown Words

It is interesting to note that how semi-supervised RB algorithm affects the ratio of labeled data. Table 3 shows the results. In table 3, "Labeled data" shows a rate of labeled data against the number of input data. Each value shown in Table 3 denotes the average accuracy, namely we randomly selected labeled data. The process is repeated 10 times for each ratio. The average accuracy is a ratio that the average number of unknown words which is assigned correctly divided by the total number of unknown words over 10 trials.

As can be seen clearly from Table 3, more labeled verbs improves overall performance in both methods, while the ratio equals to 5% and 10%, there is no statistical significance between F-scores, as the result was P-value $\geq$ .005. using micro sign test. Table 3 shows that the results obtained by semi-supervised RB are always better than those of EM at all of the ratios, especially precision. As we have shown in (Korhonen et al., 2003), each sense of the class is very delicate. Therefore, these results show the effectiveness of the method.

6.4 Error Analysis against Polysemy

We examined whether polysemous verbs were correctly classified into classes. Then, we manually analyzed clustering results obtained by running semi-supervised RB algorithm, with Dice coefficient as a similarity measure, which was the best quality F-score against the polysemic gold standard. 13 out of 71 polysemies (unlabeled data) were perfect classification: each polysemous verb was correctly classified into multiple clusters. For example, the polysemous verb, "drop" was correctly classified into four clusters. The words in italics denotes the majority sense, which corresponds to the sense according to (Levin, 1993).

Putting	{**drop** fill}
Change of State	{**drop** dry build}
Existence	{**drop** hang hit}
Motion	{**drop** walk travel}

Others were errors and classified into two patterns shown in Table 4. In Table 4, "Pattern" and "#times" refer to a type of an error and the numbers of errors, respectively. "Example" indicates example verbs, and "Target" sense(s)" denotes sense(s) that polysemies should be assigned. "Clustered sense(s)" refers to the sense(s) assigned by the system. "Partial" refers to partially correct: some senses of a polysemous verb were correctly identified, but others were not. The first example of this pattern is that "sit" has two senses, "verbs of existence" and "verbs of putting". However, only one sense: "verbs of existence" was identified correctly. The second example is that three senses of the verb "remove" were correctly into the classes, while it was classified incorrectly into the class "verbs of change of possession".

Table 3: Results against the ratio of labeled data.

Labeled data	RB							EM			
	Sim	θ	γ	C	Prec	Rec	F	C	Prec	Rec	F
5%	Dice	0.2	1.0	44	0.513	0.650	0.574	43	0.241	0.632	0.350
10%	Dice	0.2	1.0	48	0.536	0.626	0.577	59	0.301	0.512	0.387
15%	Dice	0.3	1.0	47	0.541	0.629	0.582	58	0.320	0.634	0.425
20%	Dice	0.4	1.0	58	0.563	0.638	0.593	60	0.381	0.659	0.483
25%	Dice	0.2	1.0	43	0.583	0.644	0.612	58	0.488	0.697	0.574

Table 4: Types of errors.

Pattern	#times	Example	Target sense(s)	Clustered sense(s)
Partial	23	sit	**verbs of existence**, verbs of putting	**verbs of existence**
	31	remove	**verbs of removing**, **verbs of killing** **verbs of sending and carrying**	**verbs of removing** **verbs of killing** verbs of change of possession **verbs of sending and carrying**
Poly → 1	4	hang	verbs of existence, **verbs of putting** verbs of killing **verbs involving the body**	**verbs of putting**

"poly → 1" of "Pattern" refers to polysemous verb classified into only one cluster consisting of multiple senses. "hang" was classified into one cluster. However, the cluster consisted of four senses. There was no error type that the target senses and clustered senses did not completely match.

Error analysis against polysemies provided some interesting insights for further improvement. First, we should be able to obtain further advantages in efficiency and efficacy of the method by using hierarchical splits in the clusters, as the number of clusters obtained by semi-supervised RB algorithm was smaller than the number of correct clusters. One solution is to hierarchically apply RB algorithm, *i.e.*, in the hierarchical approach, the classification problem can be decomposed into a set of smaller problems corresponding to hierarchical splits in the tree (Navigli, 2008). Roughly speaking, one first classifies to distinguish among classes at the top level, then lower level classification is performed only within the appropriate top level of the tree (Pereira et al., 1993). Each of these sub-problems can be solved much more efficiently, and hopefully more accurately as well. This is definitely worth trying with our method. Second, it is important to use other types of features, such as selectional preferences using semantic concepts from thesaurus like WordNet (Schulte im Walde et al., 2008). Third, we plan to apply the method to other thesaurus such as WordNet semantic classes, and other languages to evaluate the robustness of the method.

7 CONCLUSIONS

We have developed an approach for classifying unknown verbs including polysemies into Levin-style semantic classes. We proposed a graph-based semi-supervised clustering method which employs a small amount of labeled data to aid unsupervised learning. Moreover, the method allows verbs to belong to multiple senses. The results using the data consisting 110 test verbs was better than the EM algorithm, as the F-score obtained by the RB was 0.577 and that of the EM was 0.387. Moreover, we found that unknown words including polysemy actually aids verb classification, was robust across a wide variety of distributional similarity measures. To examine the effects of unknown words classification, we applied semi-supervised RB to the different ratio of labeled data against the number of input data. The results showed that RB is always better than the EM, even for a small number of labeled verbs, while more labeled verbs improves the overall performance in both methods. Future work includes (i) extending the method to deal with hierarchical splits in the clusters, (ii) incorporating other semantic concepts into the method, and (iii) applying the method to other dictionaries and languages.

ACKNOWLEDGEMENTS

The authors would like to thank the referees for their comments on the earlier version of this paper. This work was partially supported by the Telecommunications Advancement Foundation.

REFERENCES

Bar-Hillel, A., Hertz, T., Shental, N., and Weinshall, D. (2003). Learning Distance Functions using Equivalence Relations. In *Proc. of the 20th International Conference on Machine Learning*, pages 11–18.

Bilenko, M., Basu, S., and Mooney, R. J. (2004). Integrating Constraints and Metric Learning in Semi-Supervised Clustering. In *Proc. of the 21th International Conference on Machine Learning*, pages 81–88.

Bouraev, B., Briscoe, E. J., Carroll, J., Carter, D., and Grover, C. (1987). The Derivation of a Grammatically-Indexed Lexicon from the Longman Dictionary of Contemporary English. In *Proc. of the 25th Annual Meeting of the Association for Computational Linguistics*, pages 193–200.

Brew, C. and Walde, S. S. (2002). Spectral Clustering for German Verbs. In *Proc. of 2002 Conference on Empirical Methods in Natural Language Processing*, pages 117–123.

Briscoe, E. J. and Carroll, J. (1997). Automatic Extraction of Subcategorization from Corpora. In *Proc. of 5th ACL Conference on Applied Natural Language Processing*, pages 356–363.

Briscoe, E. J. and Carroll, J. (2002). Robust Accurate Statistical Annotaion of General Text. In *Proc. of 3rd International Conference on Language Resources and Evaluation*, pages 1499–1504.

Dagan, I., Lee, L., and Pereira, F. C. N. (1999). Similarity-based Models of Word Cooccurrence Probabilities. *Machine Learning*, 34(1-3):43–69.

Grishman, R., Macleod, C., and Meyers, A. (1994). Complex Syntax: Building a Computational Lexicon. In *Proc. of International Conference on Computational Linguistics*, pages 268–272.

Hindle, D. (1990). Noun Classification from Predicate-Argument Structures. In *Proc. of 28th Annual Meeting of the Association for Computational Linguistics*, pages 268–275.

Hughes, J. (1994). Automatically Acquiring Classification of Words. *Ph.D. thesis University of Leeds*.

Kermanidis, K., Maragoudakis, M., Fakotakis, N., and Kokkinakis, G. K. (2008). Learning Verb Complements for Modern Greek: Balancing the Noisy Dataset. *Natural Language Engineering*, 14(1):71–100.

Kirkpatrick, S., Jr., C. D. G., and Vecchi, M. P. (1983). Optimization by Simulated Annealing. *Science*, 220(4598):671–680.

Korhonen, A. (2002). Subcategorization Acquisition. *Ph.D. thesis University of Cambridge*.

Korhonen, A., Krymolowski, Y., and Briscoe, T. (2006). A Large Subcategorization Lexicon for Natural Language Processing Applications. In *Proc. of the 5th International Conference on Language Resources and Evaluation*.

Korhonen, A., Krymolowski, Y., and Marx, Z. (2003). Clustering Polysemic Subcategorization Frame Distributions Semantically. In *Proc. of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 64–71.

Lee, L. (1999). Measures of Distributional Similarity. In *Proc. of the 37th Annual Meeting of the Association for Computational Linguistics*, pages 25–32.

Leech, G. (1992). 100 Million Words of English: The British National Corpus. *Language Research*, 28(1):1–13.

Levin, B. (1993.). *English Verb Classes and Alternations*. Chicago University Press.

Lin, D. (1998). Automatic Retrieval and Clustering of Similar Words. In *Proc. of 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics*, pages 768–773.

Matsuo, Y., Sakaki, T., Uchiyama, K., and Ishizuka, M. (2006). Graph-based Word Clustering using a Web Search Engine. In *Proc. of 2006 Conference on Empirical Methods in Natural Language Processing (EMNLP2006)*, pages 542–550.

Navigli, R. (2008). A Structural Approach to the Automatic Adjudication of Word Sense Disagreements. *Natural Language Engineering*, 14(4):547–573.

Navigli, R. (2009). Word Sense Disambiguation: A Survey. *ACM Computing Surveys*, 41(2):1–69.

Ng, A. Y., Jordan, M. I., and Weiss, Y. (2002.). *On Spectral Clustering: Analysis and an Algorithm*. MIT Press.

Pereira, F., Tishby, N., and Lee, L. (1993). Distributional Clustering of English Words. In *Proc. of the 31st Annual Meeting of the Association for Computational Linguistics*, pages 183–190.

Reichardt, J. and Bornholdt, S. (2004). Detecting Fuzzy Community Structure in Complex Networks with a Potts Model. *PHYSICAL REVIEW LETTERS*, 93(21).

Reichardt, J. and Bornholdt, S. (2006). Statistical Mechanics of Community Detection. *PHYSICAL REVIEW E*, 74.

Reiter, E. and Dale, R. (2000.). *Building Natural Language Generation Systems*. Cambridge University Press.

Rooth, M. (1998). Two-Dimensional Clusters in Grammatical Relations. *In Inducing Lexicons with the EM Algorithm, AIMS Report*, 4(3).

Rooth, M., Riezler, S., Prescher, D., Carroll, G., and Beil, F. (1999). Inducing a Semantically Annotated Lexicon via EM-Based Clustering. In *Proc. of the 37th Annual Meeting of the Association for Computational Linguistics*.

Schulte im Walde, S. (2000). Clustering Verbs Semantically according to their Alternation Behaviour. In *Proc. of the 18th International Conference on Computational Linguistics*, pages 747–753.

Schulte im Walde, S., Hying, C., Scheible, C., and Schmid, H. (2008). Combining EM Training and the MDL Principle for an Automatic Verb Classification Incorporating Selectional Preferences. In *Proc. of the 46th Annual Meeting of the Association for Computational Linguistics*, pages 496–504.

Stevenson, S. and Joanis, E. (2003). Semi-Supervised Verb-Class Discovery using Noisy Features. In *Proc. of the 7th Conference on Natural Language Learning at HLT-NAACL 2003*, pages 71–78.

Wagstaff, K., Cardie, C., Rogers, S., and Schroedl, S. (2001). Constrained K-Means Clustering with Background Knowledge. In *Proc. of 18th International Conference on Machine Learning*, pages 577–584.

Widdows, D. and Dorow, B. (2002). A Graph Model for Unsupervised Lexical Acquisition. In *Proc. of 19th International conference on Computational Linguistics (COLING2002)*, pages 1093–1099.

Witten, I. H. and Bell, T. C. (1991). The Zero-Frequency Problem: Estimating the Probabilities of Novel Events in Adaptive Text Compression. *IEEE Transactions on Information Theory*, 37(4):1085–1094.