Applied Intelligence 19, 27–38, 2003

Word Sense Disambiguation by Learning Decision Trees from Unlabeled Data

SEONG-BAE PARK, BYOUNG-TAK ZHANG* AND YUNG TAEK KIM
Biointelligence Lab, School of Computer Science and Engineering, Seoul National University, Seoul 151-742, Korea
sbpark@scai.snu.ac.kr
btzhang@scai.snu.ac.kr
ytkim@cse.snu.ac.kr

Abstract. In this paper we describe a machine learning approach to word sense disambiguation that uses unlabeled data. Our method is based on selective sampling with committees of decision trees. The committee members are trained on a small set of labeled examples which are then augmented by a large number of unlabeled examples. Using unlabeled examples is important because obtaining labeled data is expensive and time-consuming while it is easy and inexpensive to collect a large number of unlabeled examples. The idea behind this approach is that the labels of unlabeled examples can be estimated by using committees. Using additional unlabeled examples, therefore, improves the performance of word sense disambiguation and minimizes the cost of manual labeling. Effectiveness of this approach was examined on a raw corpus of one million words. Using unlabeled data, we achieved an accuracy improvement up to 20.2%.

Keywords: word sense disambiguation, learning from unlabeled examples, selective sampling, committee learning, decision tree

1. Introduction

The objective of word sense disambiguation (WSD) is to identify the correct sense of a word in context [36]. It is one of the most critical tasks in most natural language processing (NLP) applications, including information retrieval, information extraction, and machine translation. The availability of large-scale corpus and various machine learning algorithms enabled corpus-based approaches to WSD [1–6]. Brown et al. [2] used a Bayesian method and Leacock et al. [4] used neural networks for word sense disambiguation. All these methods were based on a large scale sense-tagged corpus or aligned bilingual corpus.

Wilks and Stevenson [6] achieved high accuracy in WSD of English by integrating multiple knowledge sources from large scale sense-tagged corpus. Hwee and Lee also showed that the combination of different knowledge sources performs WSD effectively [3]. However, most languages except English do not have a reliable sense-tagged corpus. Therefore, any corpus-based approach to WSD for such languages should consider the following problems:

- There is no reliable and available sense-tagged corpus.
- Most words are sense ambiguous.
- Annotating large corpora requires human experts, so that it is too expensive.

Because it is expensive to construct sense-tagged corpus or bilingual corpus, many researchers tried to reduce the number of examples needed to learn WSD. Atsushi et al. [1] adopted a selective sampling method to use a small number of examples in training. They defined a training utility function to select examples with minimum certainty, and at each training iteration the examples with less certainty were saved in the

*Corresponding author.

example database. However, at each iteration of training the similarity among word property vectors must be recalculated due to their k-NN like implementation of training utility. In the text classification domain, Liere and Tadepalli [7] showed that *active learning* with majority voting performs well with only a small number of training examples.

While labeled examples obtained from a sense-tagged corpus are expensive and time-consuming, it is significantly easier to obtain the unlabeled examples. Yarowsky [8] presented, for the first time, the possibility that unlabeled examples can be used for WSD. He used a learning algorithm based on the local context under the assumption that all instances of a word have the same intended meaning within any fixed document and achieved good results with only a few labeled examples and many unlabeled ones. Nigam et al. [9] also showed the unlabeled examples can enhance the accuracy of text categorization.

In this paper, we present a new approach to word sense disambiguation that is based on selective sampling with committees. In this approach, the number of training examples is reduced by determining whether or not a given training example should be learned by weighted majority voting of multiple classifiers. The classifiers of the committee are trained first on a small training set of labeled examples and the training set is augmented by a large number of unlabeled examples. One might think that this has the possibility that the committee is misled by unlabeled examples. But our experimental results confirm that the accuracy of WSD is increased by using unlabeled examples when the members of the committee are well trained with labeled examples. We also theoretically show that performance improvement is guaranteed by a mild requirement, i.e., the base classifiers need to guess slightly better than 50%. This is because the possibility of being misled by unlabeled examples is reduced by combining outputs of multiple classifiers. One advantage of this method is that it effectively performs WSD with only a small number of labeled examples and thus shows a possibility of building word sense disambiguators for languages which have no sense-tagged corpus. In addition, the proposed method is general enough to be applied to other corpus-based approaches to natural language processing.

The rest of this paper is organized as follows. Section 2 introduces the general procedure for word sense disambiguation and the necessity of unlabeled examples. Section 3 explains how the proposed method works using both labeled and unlabeled examples. Section 4 presents the experimental results obtained by using the KAIST (Korea Advanced Institute of Science & Technology) raw corpus. Section 5 draws conclusions.

2. Problem Setting

2.1. Word Sense Disambiguation

Let $S \in \{s_1, \dots, s_k\}$ be the set of possible senses of a word to be disambiguated. To determine the sense of the word we need to consider the contextual properties. Let $\mathbf{x} = \langle x_1, \dots, x_n \rangle$ be the vector for representing selected contextual features. If we have a classifier $f(\mathbf{x}, \theta)$ parameterized with θ, then the sense of a word with property vector $\mathbf{x}$ can be determined by choosing the most probable sense s^*:

$$s^* = \arg\max_{s_j \in S} \{f(\mathbf{x}, \theta) = s_j\}.$$

The parameters θ are determined by training the classifier on a set of labeled examples, $L = \{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_N, y_N)\}$, where $y_i \in S$.

Let us take the following example sentences which show different usages of a noun *plant*, i.e., "flora" and "factory":

She cherished the *plants* like oleanders. (*flora*)
The advice is given on problems of a power *plant*. (*factory*)

What differentiates the sense of *plant* is intuitively the co-occurring neighbor words of it. That is, we can find out that the sense of *plant* in the first example is "flora" due to 'oleanders', whereas the sense of it in the second example is "factory" because of 'power'. Therefore, two labeled examples

$$(\langle \text{plant, oleander} \rangle, \text{flora})$$
$$(\langle \text{plant, power} \rangle, \text{factory})$$

are gathered from these sentences.

For an unknown usage of *plant*, the sense is determined by the classifier trained on the labeled examples. For example, the sense "factory" is assigned to a property vector $\langle$plant, car$\rangle$ from a new sentence

"They need electricity for the car *plant*."

since 'car' is more similar to 'power' than to 'oleander'.

To train the classifier f, a number of labeled examples are needed. However, it is expensive and time-consuming to obtain a large number of labeled examples because the sense for each property vector must be given by human experts. Therefore, the cost to obtain them should be reduced for practical applications.

2.2. Unlabeled Data for WSD

Many researchers tried to develop automated methods to reduce training cost in language learning and found out that the cost can be reduced by *active learning* which has control over the training examples [8–16, 38, 40]. The query-by-committee (QBC) is one of the most commonly used methods for the purpose [14, 17]. It selects informative examples out of a stream of unlabeled examples. When an example is selected by the committee the learner asks the label for it to the teacher and add it to the training set. As the examples with large variance are informative QBC selects such examples where the variance of an example is measured by disagreement among committee members.

In text classification, Liere and Tadepalli experimentally showed that QBC achieves accuracy as good as a passive single learner, but uses only 2.9% as many training examples as the single learner. McCallum and Nigam [13] modified QBC to use a naive-Bayes classifier with the unlabeled pool of documents and achieved better performance than the original QBC. However, they used the unlabeled examples just to estimate the document density to select the most informative example for labeling in QBC.

Though the number of labeled examples needed is reduced by active learning, the label of the selected examples must be given by the teacher. Thus, QBC is still expensive and a method for automatic labeling of unlabeled examples is needed to have the learner automatically gather information [5, 8, 18, 19].

As the unlabeled examples can be obtained with ease without human experts it makes WSD robust. Yarowsky presented the possibility of automatic labeling of training examples in WSD [8] and achieved good results with only a few labeled examples and many unlabeled examples. On the other hand, Blum and Mitchell [18] tried to classify Web pages, in which the description of each example can be partitioned into distinct views such as the words occurring on that page and the words occurring in hyperlinks. By using both views together, they augmented a small set of labeled examples with a lot of unlabeled examples.

The unlabeled examples in WSD can provide information about the joint probability distribution over properties but they can also mislead the learner. However, the possibility of being misled by the unlabeled examples is reduced by the committee of classifiers since combining or integrating the outputs of several classifiers in general leads to improved performance. For linear combination of unbiased classifiers, the reduction in added error is the number of classifiers that are combined [20]. Therefore, the generalization accuracy is increased by using committees. This is why we use active learning with committees to select informative unlabeled examples and label them.

3. Active Learning with Committees for WSD

A committee of classifiers is used to learn from the unlabeled examples and to determine whether a given unlabeled example should be learned or not. The label of an unlabeled example is predicted by weighted majority voting [21, 35, 39] among the committee members. Suppose that L be a set of labeled examples, C_j be the jth classifier, and C be the combined committee of classifiers. Figure 1 shows how to train the committee in a simple way and is further explained below.

3.1. Active Learning Using Unlabeled Examples

The algorithm for active learning using unlabeled data is given in Fig. 2. It looks similar to **AdaBoost.M1** proposed by Schapire and Freund [22], except that the distribution is not on the training examples but on the classifiers. It takes two sets of examples as inputs. A set L is the one with labeled examples and $D = \{\mathbf{x}_1, \ldots, \mathbf{x}_T\}$ is the one with unlabeled examples where $\mathbf{x}_i$ is a property

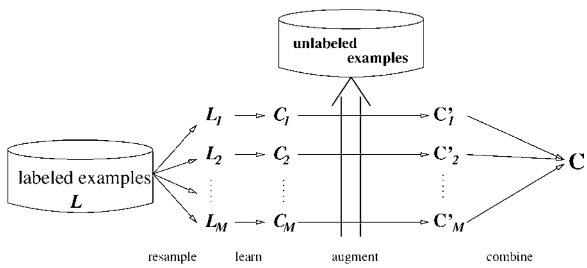

Figure 1. The procedure for training the committee of classifiers. Each classifier is trained on labeled examples and then the training set is augmented by unlabeled examples.

Given an unlabeled example set $D = \{\mathbf{x}_1, \ldots, \mathbf{x}_T\}$
and a labeled example set L
and a word sense set $S \in \{s_1, \ldots, s_k\}$ for $\mathbf{x}_i$,

Initialize $W_1(j) = \frac{1}{M}$,
where M is the number of classifiers in the committee.
Resample $L_j^{(1)}$ from L for each classifier C_j,
where $|L_j^{(1)}| = |L|$ as done in Bagging.
Train base classifier C_j $(1 \leq j \leq M)$ from $L_j^{(1)}$.
For $t = 1, \ldots, T$:

1. Each C_j predicts the sense $y_j \in S$ for $\mathbf{x}_t \in D$.

$$Y =< y_1, \ldots, y_M >$$

2. Find the most likely sense y_t from Y using distribution W:

$$y_t = \arg\max_{y \in S} \sum_{j:C_j(\mathbf{x}_t)=y} W_t(j).$$

3. Set $\alpha_t = \frac{1-\epsilon_t}{\epsilon_t}$, where

$$\epsilon_t = \frac{\text{No. of classifiers whose predictions are not } y_t}{M}.$$

4. **If** α_t is larger than a *certainty* threshold θ, **then** update W_t:

$$W_{t-1}(j) = \frac{W_t(j)}{Z_t} \times \begin{cases} \alpha_t & \text{if } y_j = y_t \\ 1 & \text{otherwise,} \end{cases}$$

where Z_t is a normalization constant.

5. **Otherwise,** every classifier C_j is restructured from new training set $L_j^{(t-1)}$:

$$L_j^{(t+1)} = L_j^{(t)} + \{(\mathbf{x}_t, y_t)\}.$$

Output the final classifier:

$$f(\mathbf{x}) = \arg\max_{y \in S} \sum_{j:C_j(\mathbf{x})=y} W_T(j).$$

Figure 2. The active learning algorithm with committees using unlabeled examples for WSD.

vector. First of all, the training set $L_j^{(1)}$ $(1 \leq j \leq M)$ of labeled examples is constructed for each base classifier C_j. This is done by random resampling using bootstrapping as in Bagging [23]. Then, each base classifier C_j is trained with the set of labeled examples $L_j^{(1)}$.

After the classifiers are trained on labeled examples, the training set is augmented by the unlabeled examples. For each unlabeled example $\mathbf{x}_t \in D$, each classifier j computes the sense $y_j \in S$ which is the label associated with it, where S is the set of possible senses of $\mathbf{x}_t$.

The distribution W over the base classifiers represents the importance weights. As the distribution can be changed each iteration, the distribution in iteration t is denoted by W_t. The importance weight of classifier C_j under distribution W_t is denoted by $W_t(j)$. Initially, the base classifiers have equal weights, so that $W_t(j) = 1/M$.

The sense of the unlabeled example $\mathbf{x}_t$ is determined by majority voting among C_j's with weight distribution W. Formally, the sense y_t of $\mathbf{x}_t$ is predicted by

$$y_t(\mathbf{x}_t) = \arg\max_{y \in S} \sum_{j:C_j(\mathbf{x}_t)=y} W_t(j).$$

If most classifiers believe that y_t is the correct sense of $\mathbf{x}_t$, they need not learn $\mathbf{x}_t$ because this example makes no contribution to reduce the variance over the distribution of examples. In this case, instead of learning the example, the weight of each classifier is updated in such a way that the classifiers whose predictions were correct get a higher importance weight and the classifiers whose predictions were wrong get a lower importance weight under the assumption that the correct sense of $\mathbf{x}_t$ is y_t. This is done by multiplying the weight of the classifier whose prediction is y_t by *certainty* α_t. To ensure the updated W_{t+1} form a distribution, W_{t+1} is normalized by constant Z_t. Formally, the importance weight is updated as follows:

$$W_{t+1}(j) = \frac{W_t(j)}{Z_t} \times \begin{cases} \alpha_t & \text{if } y_j = y_t, \\ 1 & \text{otherwise.} \end{cases}$$

The certainty α_t is computed from error ϵ_t. Because we trust that the correct sense of $\mathbf{x}_t$ is y_t, the error ϵ_t is the ratio of the number of classifiers whose predictions are not y_t. That is, α_t is computed as

$$\alpha_t = \frac{1 - \epsilon_t}{\epsilon_t},$$

where ϵ_t is given as

$$\epsilon_t = \frac{\text{No. of classifiers whose predictions are not } y_t}{M}.$$

Note that the smaller ϵ_t is, the larger the value of α_t. This implies that, if the sense of $\mathbf{x}_t$ is certainly y_t and a classifier predicts it, a higher weight is assigned to the classifier. We assume that most classifiers believe that y_t is the sense of $\mathbf{x}_t$ if the value of y_t is larger than a certainty threshold θ which is set by trial-and-error.

However, if the certainty is below the threshold, the classifiers need to learn the example $\mathbf{x}_t$ yet with belief that the sense of it is y_t. Therefore, the set of training examples, $L_j^{(t)}$, for the classifier C_j is expanded by

$$L_j^{(t+1)} = L_j^{(t)} + \{(\mathbf{x}_t, y_t)\}.$$

Then, each classifier C_j is restructured with $L_j^{(t+1)}$.

This process is repeated until the unlabeled examples are exhausted. The sense of a new example $\mathbf{x}$ is then determined by weighted majority voting among the trained classifiers:

$$f(\mathbf{x}) = \arg\max_{y\in S} \sum_{j:C_j(\mathbf{x})=y} W_T(j),$$

where $W_T(j)$ is the importance weight of classifier C_j after the learning process.

3.2. Theoretical Analysis

Previous studies show that using multiple classifiers rather than a single classifier leads to improved generalization [17, 20, 23] and learning algorithms which use *weak* classifiers can be boosted into *strong* algorithms [22, 24, 25]. In addition, Littlestone and Warmuth [21] showed that the error of the weighted majority algorithm is linearly bounded on that of the best member when the weight of each classifier is determined by held-out examples. They assumed that the classifiers make binary prediction and proved that if the best classifier makes m mistakes, the weighted majority algorithms will make at most c(log(number of classifiers)$+m$) mistakes, where c is a fixed constant. Therefore, the proposed method is plausible if it is not misled by the unlabeled examples.

The performance of the proposed method depends on that of initial base classifiers. This is because it is highly possible for unlabeled examples to mislead the learning algorithm if they are poorly trained in their initial state. However, if the accuracy of the initial majority voting is larger than $\frac{1}{2}$, the proposed method performs well as the following theorem shows.

Lemma 1. *Assume that every unlabeled data* $\mathbf{x}_t$ *is added to the set of training examples for all classifiers and the importance weights are not updated. Suppose that* p_0 *be the probability that the initial classifiers do not make errors and* β_t $(0 \le \beta_t \le 1)$ *be the probability by which the accuracy is increased in adding one more correct example or decreased in adding one more incorrect example at iteration* t*. If* $p_t \ge \frac{1}{2}$ *for all* t*, the accuracy does not decrease as a new unlabeled data is added to the training data set.*

Proof: The probability for the classifiers to predict the correct sense at iteration $t = 1$, p_1, is

$$\begin{aligned} p_1 &= p_0(p_0 + \beta_0) + (1 - p_0)(p_0 - \beta_0) \\ &= p_0(2\beta_0 + 1) - \beta_0 \end{aligned}$$

because the accuracy can be increased or decreased by β_0 with the probability p_0 and $1 - p_0$, respectively. Therefore, without loss of generality, at iteration $t = i + 1$, we have

$$p_{i+1} = p_i(2\beta_i + 1) - \beta_i.$$

To ensure the accuracy does not decrease, the condition $p_{i+1} \ge p_i$ should be satisfied.

$$\begin{aligned} p_{i+1} - p_i &= p_i(2\beta_i + 1) - \beta_i - p_i \\ &= p_i(2\beta_i) - \beta_i \ge 0 \\ \therefore\ p_i &\ge \frac{1}{2} \end{aligned}$$

The theorem follows immediately from this result. □

3.3. Decision Trees as Base Classifiers

Although any kind of learning algorithms which meet the conditions for Theorem 1 can be used as base classifiers, Quinlan's C4.5 release 8 [26] is used in this paper. The merits of decision trees that are distinguished from other learning algorithms are:

- Decision trees are strong classifiers. Although the classifiers only need to be better than random selection, the stronger the classifiers, the better the performance of the committee. This is because the possibility being misled by unlabeled examples is reduced as the classifiers get stronger.
- There is a fast restructuring algorithm for decision trees. Adding an unlabeled example with a predicted label to the existing set of training examples makes the classifiers restructured. Because the restructuring of classifiers is time-consuming, the proposed method is of little practical use without an efficient way to restructure. Utgoff et al. [27] presented two kinds of efficient algorithms for restructuring decision trees and showed experimentally that their methods perform well with only small restructuring cost.
- The values of the properties used in this paper are discrete and there could be missing values for some properties. As decision tree learning provides a practical method for discrete-valued functions and for accommodating training examples with missing attribute values, it is appropriate for the proposed method.

If we apply C4.5 directly to the property vectors, there could be a severe data-sparseness problem when some contextual properties take values of the morphological forms. Therefore, we modified C4.5 so that word matching is accomplished not by comparing morphological forms but by calculating similarity between words.

4. Experiments

We conducted experiments to see whether the unlabeled examples would enhance the committee of the classifiers learned from a small set of labeled examples.

4.1. Data Set

We used the KAIST (Korea Advanced Institute of Science & Technology) raw corpus[1] for the experiments. The entire corpus consists of about 10 million words and we used in this paper the corpus containing one million words excluding the duplicated news articles. Table 1 shows various senses of ambiguous Korean nouns considered and their sense distributions. The *percentage* column in the table denotes the ratio that the word is used with the sense in the corpus. Therefore, we can regard the maximum percentage as a lower bound on the correct sense for each word.

From the raw corpus, the property vectors are automatically extracted using a morphological analyzer and a syntactic parser [28, 29, 37]. For the part-of-speech of the words, only the best two candidates proposed by the morphological analyzer are considered by the parser to reduce the complexity of the parser. After the property vectors are generated, the sense of every vector is manually annotated. The number of examples for each sense of ambiguous nouns is also shown in Table 1.

Table 1. Various senses of Korean nouns used for the experiments and their distributions in the corpus.

Word	No. of senses	No. of examples	Sense	Percentage
bae	4	876	pear	6.2
			ship	**55.2**
			times	13.7
			stomach	24.9
bun	3	796	person	46.2
			minute	**50.8**
			indignation	3.0
jonja	2	350	the former	28.6
			electron	**71.4**
dari	2	498	bridge	30.9
			leg	**69.1**

4.2. Property Sets

Atsushi et al. [30] showed experimentally that the case markers play an important role in WSD for Japanese which has a lot of grammatical commonality with Korean. Because they paid attention to verb senses, they only considered a nominative case and an objective case. Lin [31] also showed the possibility that syntactic information can be used as properties for WSD. On the other hand, various knowledge sources such as POS and morphological forms of neighboring words are used for WSD in English [3, 6].

To select particular properties for Korean, the following characteristics should be considered:

- Korean is a partially free-order language. The ordering information on the neighbors of the ambiguous word, therefore, is probably meaningless in Korean. This is also why n-gram approaches are not generally used in statistical language processing for Korean.
- In Korean, ellipses appear very often even with a nominative case or objective case. Therefore, it is difficult to build a large scale database of labeled examples with case markers.

Considering both characteristics and results of previous work, we select eight properties for WSD of Korean nouns (Table 2). Three of them (*PARENT*,

Table 2. The properties used to distinguish the sense of an ambiguous Korean noun w.

Attribute	Substance
GFUNC	The grammatical function of w
PARENT	The word of the node modified by w
SUBJECT	Whether or not *PARENT* has a subject
OBJECT	Whether or not *PARENT* has an object
NMODWORD	The word of the noun modifier of w
ADNWORD	The head word of the adnominal phrase of w
ADNSUBJ	Whether or not the adnominal phrase of w has a subject
ADNOBJ	Whether or not the adnominal phrase of w has an object

dot-ul	*dalda-un*	*bae-ga*	*natanatda.*
sail-OBJ	hang-ADN	*ship*-NOM	appear
(A ship	with	a sail	appeared.)
sulpum-un	*myut*	*bae*	*gipda.*
sorrow-NOM	several	*times*	deep
(The sorrow	becomes	deeper several	times.)
siksa	*hu-ehnun*	*bae-ga*	*buruda.*
meal	after-TIME	*stomach*-NOM	full
(After	meal	I am	full.)
kamagwi-ga	*nal-ja*	*bae-ga*	*tulojinda.*
crow	fly-TIME	*pear*-NOM	fall down
(Pears	fall down	when	crows fly.)

Figure 3. Example sentences associated with Korean noun *bae*.

NMODWORD, *ADNWORD*) take morphological form as their value, one (*GFUNC*) takes 11 values of grammatical functions,[2] and others take only *true* or *false*. Although the number of properties considering morphological forms is three, it can be reduced into one or two because some nouns do not have noun or adnominal modifiers. This is helpful to tackle the data sparseness problem.

For example, let us consider a Korean noun *bae*. Figure 3 shows example sentences associated with it. The noun *bae* has four senses: ship, pear, times and stomach. The symbols for the case markers used in the examples are NOM (*nominative*), OBJ (*objective*), ADN (*adnominal*), and TIME (*time*). Given input sentences, the property vectors are automatically gathered. Let us take an example the first sentence. In this example, there is an adnominal phrase but no noun phrase which modifies *bae*. Therefore, a property vector

$$\langle \text{SUBJECT}, \mathit{natanatda}, \text{True}, \text{False}, \text{None}, \\ \mathit{dalda}, \text{False}, \text{True}\rangle$$

is obtained from this example because *bae* plays a subject role, modifies a verb *natanatda* (appear) which has a subject and no object, and is modified by the adnominal phrase whose head verb is *dalda* (hang) only with an object. As the sense of the first example is 'ship', a labeled example (<SUBJECT, *natanatda*, True, False, None, *dalda*, False, True>, *ship*) is obtained. From other senses, the labeled examples are also obtained in the same way.

4.3. Experimental Results

In the experiments, if there is a tie in predicting senses, the sense with the lowest order is chosen as in [23]. For each noun, 90% of the examples are used for training and the remaining 10% are used for testing. For the experiments, 15 base classifiers are used and the certainty threshold θ is set empirically to 2.0, which implies that two thirds of the classifiers agree. Though this does not mean that 15 classifiers are necessary or sufficient, 15 seems to be a reasonable number. Table 3 shows the average accuracy of WSD for Korean nouns obtained by using 1, 10, 15, and 20 classifiers. Using 15 classifiers, we get higher accuracy than using 10 classifiers and as good accuracy as using 20 classifiers. Although 15 may not be optimal, it seems reasonable.

As there are three properties for a morphological form, the problem of data sparseness may occur. Such a problem can be overcome by using a thesaurus or word classes, but no reliable thesaurus is available for Korean yet. Many researchers proposed statistical methods to overcome data sparseness [32]. However, it is also difficult to build word classes for a reasonable number of words in a statistical method because there is no syntax-tagged corpus for Korean for practical applications.

In this paper, we calculate the similarity between two Korean words using Korean-English dictionary and *WordNet*, the thesaurus for English [33]. The relation among words in WordNet is expressed by autonym, hypernym, hyponym, meronym and holonym, but only hypernym and hyponym which represent 'is-a' relation are used in this paper. The similarity between two Korean words is regarded as an average similarity of their English translated words in the Korean-English dictionary. Because the word translation between two languages is not one-to-one mapping in general, a Korean word can be translated into several English words. Therefore, the similarity between English translated words are averaged.

Table 3. The accuracy obtained of word sense disambiguation for Korean nouns by using the various number of classifiers.

No. of classifiers	Accuracy (%)
1	78.2
10	85.9
15	87.0
20	87.0

Table 4. The accuracy of word sense disambiguation for Korean nouns by the proposed method. For the proposed method which uses partially labeled examples, the accuracy is measured when it shows best accuracy. The proposed method achieves 23.6% improvement over the lower bound and it shows higher accuracy than the single C4.5 trained on the whole labeled examples for noun '*jonja*'.

Word	Using partially labeled data	Using all labeled data	Lower bound
bae	81.5 ± 7.7%	**82.3% ± 5.9%**	55.2%
bun	92.3 ± 7.7%	**94.3% ± 5.7%**	50.8%
jonja	**93.5 ± 6.5%**	90.6% ± 9.4%	71.4%
dari	73.3 ± 14.2%	**80.8 ± 10.9%**	69.1%
Average	85.2%	**87.0%**	61.6%

Table 4 shows the 10-fold cross validation result of WSD experiments for nouns listed in Table 1. The accuracy of the proposed method shown in Table 4 is measured when the accuracy is in its best for various ratios of the number of labeled examples for base classifiers to total examples. The results show that WSD by selective sampling with committees using both labeled and unlabeled examples is comparable to a single learner using all the labeled examples. In addition, the method proposed in this paper achieves 26.3% improvement over the lower bound for '*bae*', 41.5% for '*bun*', 22.1% for '*jonja*', and 4.2% for '*dari*', which is 23.6% improvement on the average. Especially, for '*jonja*' the proposed method shows higher accuracy than the single C4.5 trained on the whole labeled examples.

Figure 4 shows how the initial number of labeled examples for the base classifiers influences the performance. The x-axis of Fig. 4 represents the ratio of the number of labeled examples to the entire training examples. The horizontal lines in the figure show the lower bounds on the accuracy of sense determination that corresponds to choosing the one appeared most

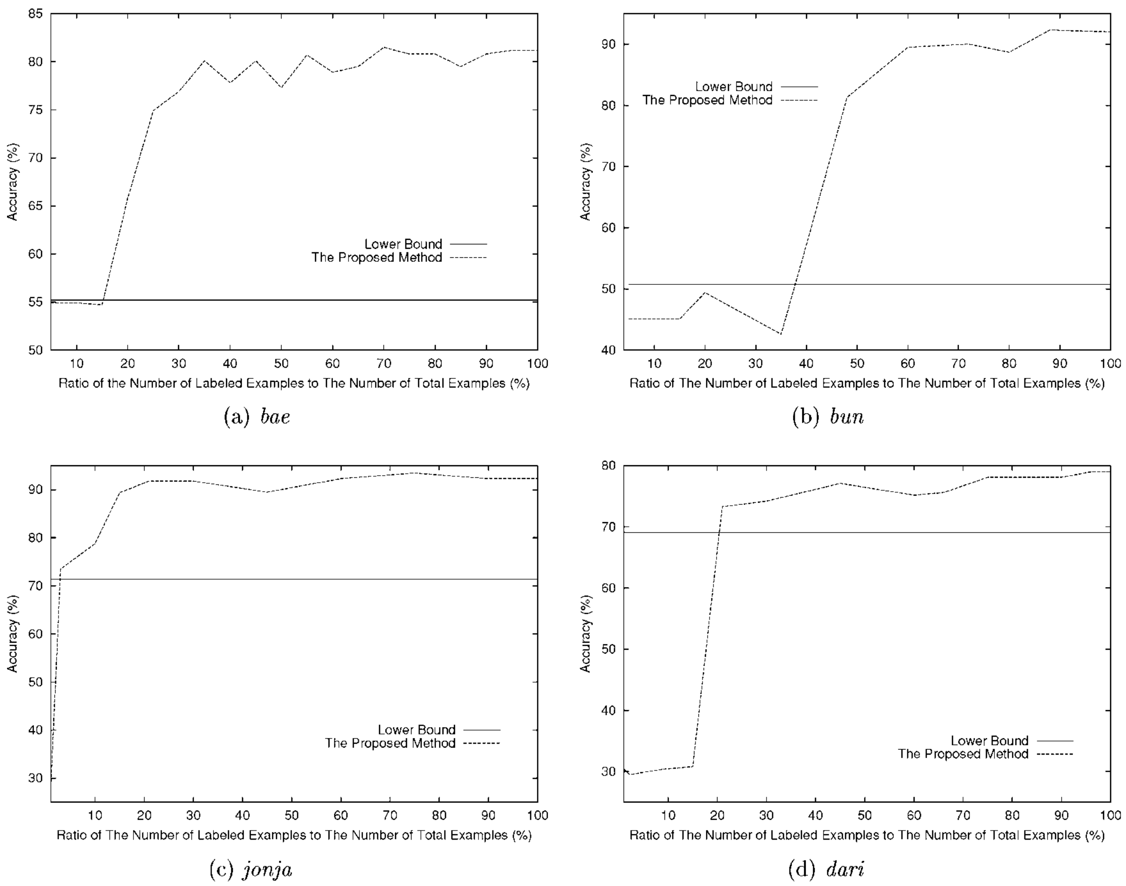

Figure 4. The accuracy of word sense disambiguation as a function of the number of labeled examples. The number of labeled examples is increased by 5% and the accuracy was measured using the 10-fold cross validation.

frequently. Figure 4(a) shows the accuracy for noun '*bae*'. As the number of labeled examples for the classifiers increases, the accuracy goes up and is almost flat around 35%. From this figure and Table 4, we find that the proposed method that uses only 35% of labeled examples in its initial state can achieve the accuracy of the learning algorithms which use all the labeled examples.

In Fig. 4, it is interesting to observe jumps in the accuracy curve. The jump appears because the unlabeled examples mislead the classifiers only when the classifiers are poorly trained, but they play an important role as information to select senses when the classifiers are well trained on labeled examples. Other nouns show similar phenomena though the percentage of labeled examples is different when the accuracy gets flat. The accuracy gets flat with 60% of labeled examples for '*bun*', 15% for '*jonja*', and 20% for '*dari*'. That is, the committee is trained enough to predict the unlabeled examples if only about 32% of the examples on the average are labeled in advance.

Figure 5 shows the performance improved by using unlabeled examples. This figure demonstrates that the proposed method outperforms the one without using unlabeled examples. The *initial learning* in the figure means that the committee is trained with labeled examples but is not augmented by unlabeled examples. The difference between two lines is the improved accuracy obtained by using unlabeled examples. When the accuracy of the proposed method gets stabilized for the first time, the improved accuracy by using unlabeled examples is 20.2% for '*bae*', 9.9% for '*bun*', 13.5% '*jonja*', and 13.4% for '*dari*', which is 14.3% on the average. It should be mentioned that the results also show that the accuracy of the proposed method may be dropped when the classifier is trained on too small a set of labeled data, as is the case in the early stages

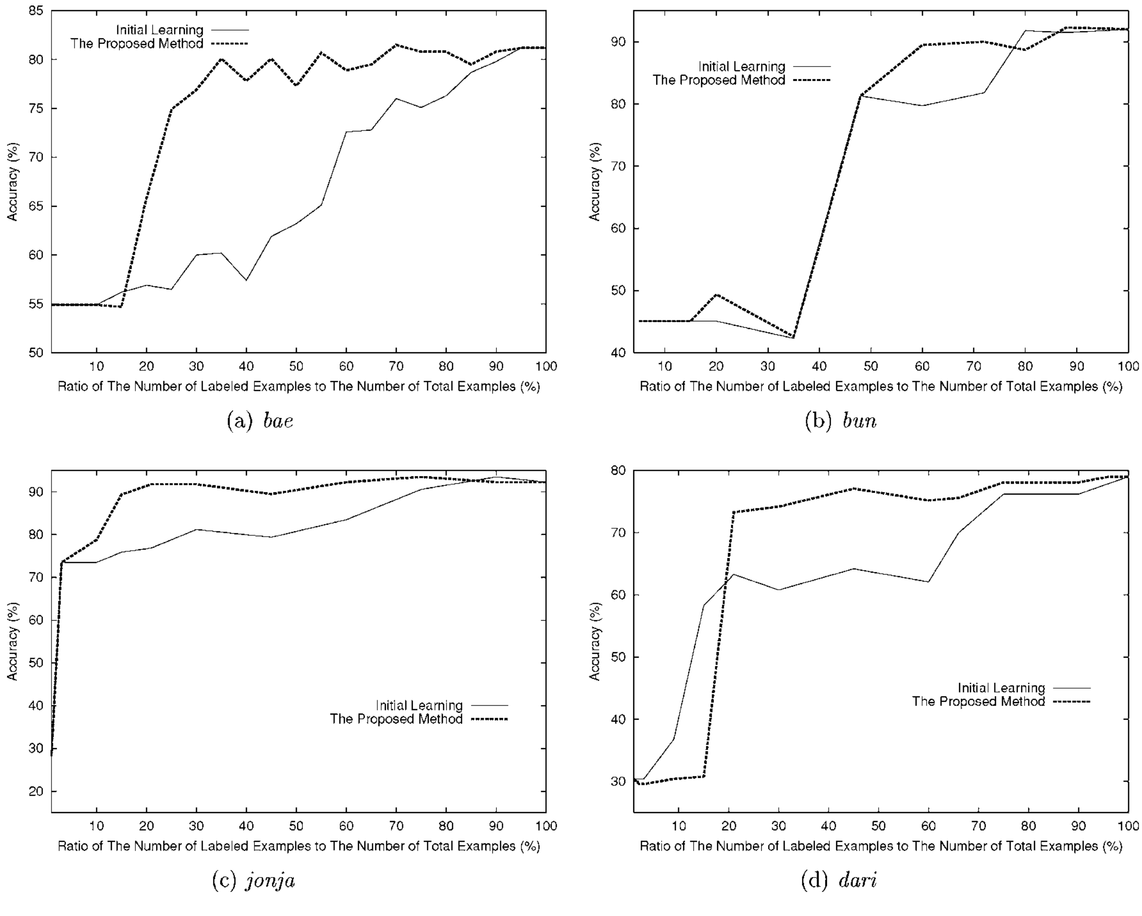

Figure 5. Improvement in accuracy by using unlabeled examples. The *initial learning* means that the committee is trained on labeled examples, but is not augmented by unlabeled examples.

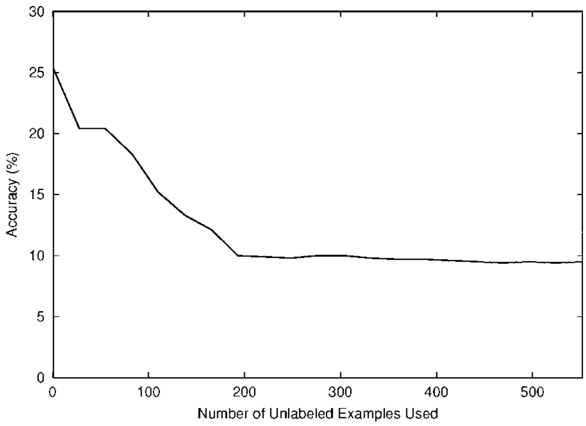

Figure 6. The impact of Lemma 1. The experiment is performed on a noun '*bae*'. The label of unlabeled examples is always predicted to be '*stomach*'.

of Fig. 5(d). However, in typical situations where the classifiers are trained on minimum training set size, this does not happen as the results in Fig. 4 show. In addition, we can find in this particular experiment that the accuracy is always improved by using unlabeled examples if only about 22% of training examples, on the average, are labeled in advance.

Figure 6 shows the impact of Lemma 1. If the base classifier predicts that the label of unlabeled examples for a noun '*bae*' would be always '*stomach*', the accuracy ratio constantly decreases to near 10%. Since '*stomach*' has 24.9% of distribution (Table 1), most of unlabeled examples are mislabeled so that the accuracy decreases rather than increases by unlabeled data. Therefore, it is required to keep the probability of correct prediction being greater than $\frac{1}{2}$.

In order to display how the proposed method performs for English, the proposed method is compared with that of Wilks and Stevenson [6]. In the approach of Wilks and Stevenson, the sense of words is disambiguated by an optimized combination of lexical knowledge sources and a POS filter. In the experiments, we choose the *Longman Dictionary of Contemporary English* (LDOCE) for lexical knowledge sources, Brill's POS tagger [34] for the POS filter. For the sense-tagged corpus, we take SEMCOR, a 200,000 word corpus with the words manually tagged as part of the WordNet project. The experiment is performed on noun '*plant*'. Since LDOCE has five senses and WordNet has four senses for '*plant*', all usages of '*plant*' whose sense is not one of two senses in Section 2.1 are ignored. The classification ratio of the approach of Wilks and Stenvenson is 88.2% for '*plant*'. The

Table 5. The experimental result on an English noun *plant*.

Property	Value
No. of Examples with Sense '*flora*'	38
No. of Examples with Sense '*factory*'	63
No. of Labeled Examples Need to Achieve 86% Accuracy	66

proposed method achieves 86% of accuracy with only 66 labeled examples (Table 5), where the total number of labeled examples is 101. In effect, the proposed method achieves the disambiguation as accurate as the approach of Wilks and Stevenson with only two thirds of labeled examples.

5. Conclusions

In this paper we have proposed a new method for word sense disambiguation that uses unlabeled data. Our method is based on selective sampling with committees of decision trees. The committee members are first trained on a small training set of labeled examples and the training set is augmented by a large number of unlabeled examples.

Using unlabeled data is especially important in word sense disambiguation because unlabeled data are ubiquitous whereas labeled data are expensive to obtain. In a series of experiments on Korean nouns we showed that the accuracy is improved up to 20.2% using only 32% of labeled data. This implies, the learning model trained on a small number of labeled data can be enhanced by using additional unlabeled data. The accuracy may deteriorate when the classifiers are trained on a fewer number of labeled data than are actually needed because the model becomes unstable. However, as Lemma 1 proves, the accuracy is always improved if the individual classifiers do better than random selection after being trained on labeled data. In our experiments, 22% of labeled data satisfies this condition and thus guarantee an improved accuracy.

Note that hand-labeling is one of the major drawbacks for corpus-based approaches to natural language processing. Our approach minimizes the burden of manual labeling by using additional unlabeled data because the labels of unlabeled data are estimated by committees of decision trees. Our experimental results show that the committee model using only 32% of labeled data is comparable to a single learner using all the labeled data. The proposed learning model seems especially effective and useful when only a sense-tagged corpus of small size is available. An additional advantage of the proposed method is that it can reduce the cost and efforts on observing non-informative examples through selective sampling.

The computational complexity of the proposed method is large due to overhead of constructing and evaluating all the intermediate classifiers. However, the constructing cost of intermediate classifiers can be reduced, since we use decision trees for which a fast restructuring algorithm exists. Moreover, the proposed method is also suitable for parallel computing. The construction of each classifier proceeds without communication from the other classifiers. Thus, the computational complexity is not too serious.

A final remark is that the proposed method can also be applied to other kinds of language learning problems such as POS-tagging, PP attachment, and text classification. These problems are similar to word sense disambiguation in the sense that they have limited and expensive labeled data, but abundant and inexpensive unlabeled data. Thus, the method of selective sampling from unlabeled samples can also lead to improvement in predictive accuracy in these domains.

Acknowledgments

This research was supported in part by the Korean Ministry of Education under the BK21 IT Program and by the Korean Ministry of Science and Technology under the Brain Tech program.

Notes

1. This corpus is distributed by the Korea Terminology Research Center for Language and Knowledge Engineering. It can be accessed in http://kibs.kaist.ac.kr.
2. These 11 grammatical functions are from the parser, KEMTS (Korean-to-English Machine Translation System) developed in Seoul National University, Korea.

References

1. F. Atsushi, I. Kentaro, T. Takenobu, and T. Hozumi, "Selective sampling of effective example sentence sets for word sense disambiguation," *Computational Linguistics*, vol. 24, no. 4, pp. 573–597, 1998.
2. P. Brown, S. Della-Pietras, V. Della-Pietras, and R. Mercer, "Word sense disambiguation using statistical methods," in *Proceedings of the 29th Annual Meeting of the Association for Computational Linguistics*, 1991, pp. 264–270.
3. T. Hwee and H. Lee, "Integrating multiple knowledge sources to disambiguate word sense: An examplar-based approach," in *Proceedings of the 34th Annual Meeting of the ACL*, 1996, pp. 40–47.
4. C. Leacock, G. Towell, and E. Voorhees, "Towords building contextural representations of word senses using statistical models," in *Proceedings of the SIGLEX Workshop: Acquisition of Lexical Knowledge from Text*, 1993, pp. 10–20.
5. T. Pedersen and R. Bruce, "Distinguishing word senses in untagged text," in *Proceedings of the Second Conference on Empirical Methods in Natural Language Processing*, 1997, pp. 399–401.
6. Y. Wilks and M. Stevenson, "Word sense disambiguation using optimised combinations of knowledge sources," in *Proceedings of COLING-ACL'98*, 1998, pp. 1398–1402.
7. R. Liere and P. Tadepalli, "Active learning with committees for text categorization," in *Proceedings of AAAI-97*, 1997, pp. 591–596.
8. D. Yarowsky, "Unsupervised word sense disambiguation rivaling supervised methods," in *Proceedings of the 33rd Annual Meeting of the ACL*, 1995, pp. 189–196.
9. K. Nigam, A. McCallum, S. Thrun, and T. Mitchell, "Learning to classify text from labeled and unlabeled documents," *Machine Learning*, vol. 39, pp. 1–32, 2000.
10. I. Dagan and S. Engelson, "Committee-based sampling for training probabilistic classifiers," in *Proceedings of the Fourteenth International Conference on Machine Learning*, 1997, pp. 150–157.
11. K. Lang, "Newsweeder: Learning to filter netnews," in *Proceedings of the Twelfth Internation Conference on Machine Learning*, 1997, pp. 331–339.
12. D. Lewis and W. Gale, "A sequential algorithm for training text classifiers," in *Proceedings of SIGIR-94*, 1994, pp. 5–11.
13. A. McCallum and K. Nigam, "Employing EM and pool-based active learning for text classification," in *Proceedings of the Fifteenth International Conference on Machine Learning*, 1998, pp. 359–367.
14. G. Paaß and J. Kindermann, "Bayesian query construction for neural network models," in *Proceedings of Advances in Neural Information Processing Systems 7*, 1995, pp. 443–450.
15. B.-T. Zhang, "Accelerated learning by active example selection," *International Journal of Neural Systems*, vol. 5, no. 1, pp. 67–75, 1994.
16. B.-T. Zhang and D.-Y. Cho, "Genetic programming with active data selection," *Simulated Evolution and Learning*, vol. LNAI 1585, pp. 146–153, 1999.
17. Y. Freund, H. Seung, E. Shamir, and N. Tishiby, "Selective sampling using the query by committee algorithm," *Machine Learning*, vol. 28, pp. 133–168, 1997.
18. A. Blum and T. Mitchell, "Combining labeled and unlabeled data with co-training," in *Proceedings of COLT-98*, 1998, pp. 92–100.
19. D. Miller and H. Uyar, "A mixture of experts classifier with learning based on both labelled and unlabelled data," in *Proceedings of Advances in Neural Information Processing System 9*, 1997, pp. 571–577.
20. K. Tumer and J. Ghosh, "Error correlation and error reduction in ensemble classifiers," *Connection Science*, vol. 8, no. 34, pp. 385–404, 1996.
21. N. Littlestone and M. Warmuth, "The weighted majority algorithm," *Information and Computation*, vol. 108, no. 2, pp. 212–261, 1994.

22. Y. Freund and R. Schapire, "Experiments with a new boosting algorithm," in *Proceedings of the Thirteenth International Conference on Machine Learning*, 1996, pp. 148–156.
23. L. Breiman, "Bagging predictors," *Machine Learning*, vol. 24, pp. 123–140, 1996.
24. T. Dietterich, M. Kearns, and Y. Mansour, "Applying the weak learning framework to understand and improve C4.5," in *Proceedings of the Thirteenth International Conference on Machine Learning*, 1996, pp. 96–104.
25. R. Schapire, "Theoretical views of boosting," in *Proceedings of EuroCOLT*, 1999, pp. 1–10.
26. R. Quinlan, *C4.5: Programs For Machine Learning*, Morgran Kaufmann Publishers, 1993.
27. P. Utgoff, N. Berkman, and J. Clouse, "Decision tree induction based on efficient tree restructuring," *Machine Learning*, vol. 29, pp. 5–44, 1997.
28. S. Kang and Y. Kim, "Syllable-based model for the Korean morphology," in *Proceedings of COLING-94*, 1994, pp. 221–226.
29. J. Yang and Y. Kim, "Korean analysis using multiple knowledge sources," *Journal of The Korea Information Science Society*, vol. 21, no. 7, pp. 1324–1332, 1994. (in Korean)
30. F. Atsushi, I. Kentaro, T. Takenobu, and T. Hozumi, "To what extent does case contribute to verb sense disambiguation?" in *Proceedings of COLING-96*, 1996, pp. 59–64.
31. D. Lin, "Using syntactic dependency as local context to resolve word sense ambiguity," in *Proceedings of the 35th Annual Meeting of the ACL*, 1997, pp. 64–71.
32. S. Chen and J. Goodman, "An empirical study of smoothing techniques for language modeling," in *Proceedings of the 34th Annual Meeting of the ACL*, 1996, pp. 310–318.
33. C. Fellbaum, *WordNet: An Electronic Lexical Databse*, The MIT Press, 1998.
34. E. Brill, "A simple rule-based part of speech tagger," in *Proceedings of the Third Conference on Applied Natural Language Processing*, 1992, pp. 152–155.
35. P. Chan and S. Stolfo, "A comparative evaluation of voting and meta-learning on partitioned data," in *Proceedings of the Twelfth International Conference on Machine Learning*, 1995, pp. 90–98.
36. E. Charniak, *Statistical Language Learning*, The MIT Press, 1993.
37. J.-M. Cho and G.-C. Kim, "Korean verb sense disambiguation using distributional information from corpora," in *Proceedings of Natural Language Processing Pacific Rim Symposium*, 1995, pp. 691–696.
38. J. Diederich, "Connectionist recruitment learning," in *Proceedings of European Conference on Artificial Intelligence*, 1988, pp. 351–356.
39. P. Domingos, "Knowledge acquisition from examples via multiple models," in *Proceedings of the Fourteenth International Conference on Machine Learning*, 1997, pp. 98–106.
40. B.-T. Zhang, "Learning by incremental selection of critical examples," Arbeitspapiere der GMD, No. 735, German National Research Center for Computer Science (GMD), St. Augustin/Bonn, Germany, March 1993.

Seong-Bae Park is a postdoc in the School of Computer Science and Engineering, Seoul National University. He received his BS degree in computer science from Korea Advanced Institute of Science and Technology in 1994, and MS and Ph.D. degrees in computer engineering from Seoul National University in 1996 and 2002, respectively. His research interests include natural language processing, information retrieval, and machine learning.

Byoung-Tak Zhang is an associate professor of Computer Science and Engineering at Seoul National University (SNU). He received his BS and MS degrees in computer engineering from SNU in 1986 and 1988, respectively, and a Ph.D. in Computer Science from University of Bonn, Germany in 1992. Prior to joining SNU, Dr. Zhang has been a research associate at German National Research Center for Information Technology (GMD). He serves as an associate editor of IEEE Transactions on Evolutionary Computation. His research interests are in learning and adaptive systems, evolutionary computation, probabilistic neural networks, and their application to real-world AI problems.

Yung Taek Kim is an emeritus professor of Computer Science and Engineering at Seoul National University. He served as a professor in Seoul National University from 1971 to 2000. He received his MS and Ph.D. in computer science from University of Colorado and University of Utah respectively.