Proceedings of the 40th Annual Meeting of the Association for
Computational Linguistics (ACL), Philadelphia, July 2002, pp. 343-351.

Word Translation Disambiguation Using Bilingual Bootstrapping

Cong Li
Microsoft Research Asia
5F Sigma Center, No.49 Zhichun Road, Haidian
Beijing, China, 100080
i-congl@microsoft.com

Hang Li
Microsoft Research Asia
5F Sigma Center, No.49 Zhichun Road, Haidian
Beijing, China, 100080
hangli@microsoft.com

Abstract

This paper proposes a new method for word translation disambiguation using a machine learning technique called 'Bilingual Bootstrapping'. Bilingual Bootstrapping makes use of , in learning, a small number of classified data and a large number of unclassified data *in the source and the target languages* in translation. It constructs classifiers in the two languages in parallel and repeatedly boosts the performances of the classifiers by further classifying data in each of the two languages and by exchanging between the two languages information regarding the classified data. Experimental results indicate that word translation disambiguation based on Bilingual Bootstrapping *consistently* and *significantly* outperforms the existing methods based on 'Monolingual Bootstrapping'.

1 Introduction

We address here the problem of word translation disambiguation. For instance, we are concerned with an ambiguous word in English (e.g., 'plant'), which has multiple translations in Chinese (e.g., '工厂 (gongchang)' and '植物 (zhiwu)'). Our goal is to determine the correct Chinese translation of the ambiguous English word, given an English sentence which contains the word. Word translation disambiguation is actually a special case of word sense disambiguation (in the example above, 'gongchang' corresponds to the sense of 'factory' and 'zhiwu' corresponds to the sense of 'vegetation').[1]

Yarowsky (1995) proposes a method for word sense (translation) disambiguation that is based on a bootstrapping technique, which we refer to here as 'Monolingual Bootstrapping (MB)'.

In this paper, we propose a new method for word translation disambiguation using a bootstrapping technique we have developed. We refer to the technique as 'Bilingual Bootstrapping (BB)'.

In order to evaluate the performance of BB, we conducted some experiments on word translation disambiguation using the BB technique and the MB technique. All of the results indicate that BB consistently and significantly outperforms MB.

2 Related Work

The problem of word translation disambiguation (in general, word sense disambiguation) can be viewed as that of classification and can be addressed by employing a *supervised* learning method. In such a learning method, for instance, an English sentence containing an ambiguous English word corresponds to an example, and the Chinese translation of the word under the context corresponds to a classification decision (a label).

Many methods for word sense disambiguation using a supervised learning technique have been proposed. They include those using Naïve Bayes (Gale et al. 1992a), Decision List (Yarowsky 1994), Nearest Neighbor (Ng and Lee 1996), Transformation Based Learning (Mangu and Brill 1997), Neural Network (Towell and

[1] In this paper, we take English-Chinese translation as example; it is a relatively easy process, however, to extend the discussions to translations between other language pairs.

Voorhess 1998), Winnow (Golding and Roth 1999), Boosting (Escudero et al. 2000), and Naïve Bayesian Ensemble (Pedersen 2000). Among these methods, the one using Naïve Bayesian Ensemble (i.e., an ensemble of Naïve Bayesian Classifiers) is reported to perform the best for word sense disambiguation with respect to a benchmark data set (Pedersen 2000).

The assumption behind the proposed methods is that it is nearly always possible to determine the translation of a word by referring to its context, and thus all of the methods actually manage to build a classifier (i.e., a classification program) using features representing context information (e.g., co-occurring words).

Since preparing supervised learning data is expensive (in many cases, manually labeling data is required), it is desirable to develop a *bootstrapping* method that starts learning with a small number of classified data but is still able to achieve high performance under the help of a large number of unclassified data which is not expensive anyway.

Yarowsky (1995) proposes a method for word sense disambiguation, which is based on Monolingual Bootstrapping. When applied to our current task, his method starts learning with a small number of English sentences which contain an ambiguous English word and which are respectively assigned with the correct Chinese translations of the word. It then uses the classified sentences as training data to learn a classifier (e.g., a decision list) and uses the constructed classifier to classify some unclassified sentences containing the ambiguous word as additional training data. It also adopts the heuristics of ‘one sense per discourse’ (Gale et al. 1992b) to further classify unclassified sentences. By repeating the above processes, it can create an accurate classifier for word translation disambiguation.

For other related work, see, for example, (Brown et al. 1991; Dagan and Itai 1994; Pedersen and Bruce 1997; Schutze 1998; Kikui 1999; Mihalcea and Moldovan 1999).

3 Bilingual Bootstrapping

3.1 Overview

Instead of using Monolingual Bootstrapping, we propose a new method for word translation disambiguation using Bilingual Bootstrapping. In translation from English to Chinese, for instance, BB makes use of not only unclassified data in English, but also unclassified data in Chinese. It also uses a small number of classified data in English and, *optionally,* a small number of classified data in Chinese. The data in English and in Chinese are supposed to be not in parallel but from the same domain.

BB constructs classifiers for English to Chinese translation disambiguation by repeating the following two steps: (1) constructing classifiers for each of the languages on the basis of the classified data *in both languages*, (2) using the constructed classifiers in each of the languages to classify some unclassified data and adding them to the classified training data set of the language. The reason that we can use classified data in both languages at step (1) is that words in one language generally have translations in the other and we can find their translation relationship by using a dictionary.

3.2 Algorithm

Let E denote a set of words in English, C a set of words in Chinese, and T a set of links in a translation dictionary as shown in Figure 1. (Any two linked words can be translation of each other.) Mathematically, T is defined as a *relation* between E and C , i.e., $T \subseteq E \times C$.

Let ε stand for a random variable on E, γ a random variable on C. Also let e stand for a random variable on E, c a random variable on C, and t a random variable on T. While ε and γ represent words to be translated, e and c represent context words.

For an English word ε, $T_{\varepsilon} = \{t \mid t = (\varepsilon, \gamma'), t \in T\}$ represents the links

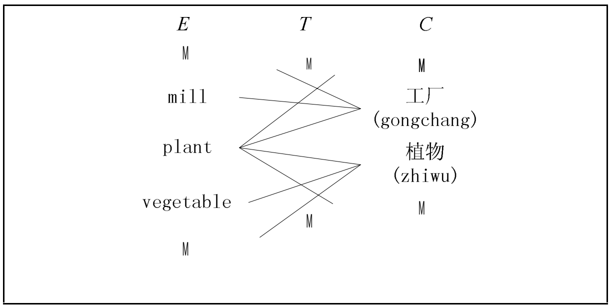

Figure 1: Example of translation dictionary

```
Input: $E, C, T, L_E, U_E, L_C, U_C$ , Parameter: $b, \theta$
Repeat in parallel the following processes for English (left) and Chinese (right), until unable to continue:
1. for each ($\varepsilon \in E$) {
     for each ($t \in T_\varepsilon$) {
       use $L_\varepsilon$ and $L_\gamma$ ($\gamma \in C_\varepsilon$) to create classifier:
       $P_\varepsilon(t|\mathbf{e}),\ t \in T_\varepsilon$ & $P_\varepsilon(\bar{t}|\mathbf{e}),\ \bar{t} \in T_\varepsilon - \{t\};$ }}
2. for each ($\varepsilon \in E$) {
     $NU \leftarrow \{\}; NL \leftarrow \{\};$
     for each ($t \in T_\varepsilon$) { $S_t \leftarrow \{\}; Q_t \leftarrow \{\};$ }
     for each ($\mathbf{e} \in U_\varepsilon$){
       calculate $\lambda^*(\mathbf{e}) = \max_{t \in T_\varepsilon} \frac{P_\varepsilon(t|\mathbf{e})}{P_\varepsilon(\bar{t}|\mathbf{e})}$;
       let $t^*(\mathbf{e}) = \arg\max_{t \in T_\varepsilon} \frac{P_\varepsilon(t|\mathbf{e})}{P_\varepsilon(\bar{t}|\mathbf{e})}$;
       if ($\lambda^*(\mathbf{e}) > \theta$ & $t^*(\mathbf{e}) = t$)
         put e into $S_t$;}
     for each ($t \in T_\varepsilon$){
       sort $\mathbf{e} \in S_t$ in descending order of $\lambda^*(\mathbf{e})$ and
       put the top b elements into $Q_t$;}
     for each ($\mathbf{e} \in \cup_t Q_t$){
       put e into NU and put $(\mathbf{e}, t^*(\mathbf{e}))$ into NL;}
     $L_\varepsilon \leftarrow L_\varepsilon \cup NL; U_\varepsilon \leftarrow U_\varepsilon - NU$;}

for each ($\gamma \in C$) {
  for each ($t \in T_\gamma$) {
    use $L_\gamma$ and $L_\varepsilon$ ($\varepsilon \in E_\gamma$) to create classifier:
    $P_\gamma(t|\mathbf{c}),\ t \in T_\gamma$ & $P_\gamma(\bar{t}|\mathbf{c}),\ \bar{t} \in T_\gamma - \{t\};$ }}
for each ($\gamma \in C$) {
  $NU \leftarrow \{\}; NL \leftarrow \{\};$
  for each ($t \in T_\gamma$) { $S_t \leftarrow \{\}; Q_t \leftarrow \{\};$ }
  for each ($\mathbf{c} \in U_\gamma$){
    calculate $\lambda^*(\mathbf{c}) = \max_{t \in T_\gamma} \frac{P_\gamma(t|\mathbf{c})}{P_\gamma(\bar{t}|\mathbf{c})}$;
    let $t^*(\mathbf{c}) = \arg\max_{t \in T_\gamma} \frac{P_\gamma(t|\mathbf{c})}{P_\gamma(\bar{t}|\mathbf{c})}$;
    if ($\lambda^*(\mathbf{c}) > \theta$ & $t^*(\mathbf{c}) = t$)
      put c into $S_t$;}
  for each ($t \in T_\gamma$){
    sort $\mathbf{c} \in S_t$ in descending order of $\lambda^*(\mathbf{c})$ and
    put the top b elements into $Q_t$;}
  for each ($\mathbf{c} \in \cup_t Q_t$){
    put c into NU and put $(\mathbf{c}, t^*(\mathbf{c}))$ into NL;}
  $L_\gamma \leftarrow L_\gamma \cup NL; U_\gamma \leftarrow U_\gamma - NU$;}
Output: classifiers in English and Chinese
```

Figure 2: Bilingual Bootstrapping

from it, and $C_\varepsilon = \{\gamma' | (\varepsilon, \gamma') \in T\}$ represents the Chinese words which are linked to it. For a Chinese word γ, let $T_\gamma = \{t | t = (\varepsilon', \gamma), t \in T\}$ and $E_\gamma = \{\varepsilon' | (\varepsilon', \gamma) \in T\}$. We can define C_e and E_c similarly.

Let **e** denote a sequence of words (e.g., a sentence or a text) in English

$$\mathbf{e} = \{e_1, e_2, \mathrm{L}, e_m\},\ e_i \in E\ \ (i = 1,2, \mathrm{L}, m).$$

Let **c** denote a sequence of words in Chinese

$$\mathbf{c} = \{c_1, c_2, \mathrm{L}, c_n\},\ c_i \in C\ \ (i = 1,2, \mathrm{L}, n).$$

We view **e** and **c** as examples representing context information for translation disambiguation.

For an English word ε, we define a *binary* classifier for resolving each of its translation ambiguities in T_ε in a general form as:

$$P_\varepsilon(t|\mathbf{e}),\ t \in T_\varepsilon \quad \& \quad P_\varepsilon(\bar{t}|\mathbf{e}), \bar{t} \in T_\varepsilon - \{t\},$$

where **e** denotes an example in English. Similarly, for a Chinese word γ, we define a classifier as:

$$P_\gamma(t|\mathbf{c}),\ t \in T_\gamma \quad \& \quad P_\gamma(\bar{t}|\mathbf{c}),\ \bar{t} \in T_\gamma - \{t\},$$

where **c** denotes an example in Chinese.

Let L_ε denote a set of classified examples in English, each representing one context of ε

$$L_\varepsilon = \{(\mathbf{e}_1, t_1)_\varepsilon, (\mathbf{e}_2, t_2)_\varepsilon, \mathrm{L}, (\mathbf{e}_k, t_k)_\varepsilon\},$$
$$t_i \in T_\varepsilon\ (i = 1,2, \mathrm{L}, k),$$

and U_ε a set of unclassified examples in English, each representing one context of ε

$$U_\varepsilon = \{(\mathbf{e}_1)_\varepsilon, (\mathbf{e}_2)_\varepsilon, \mathrm{L}, (\mathbf{e}_l)_\varepsilon\}.$$

Similarly, we denote the sets of classified and unclassified examples with respect to γ in Chinese as L_γ and U_γ respectively. Furthermore, we have

$$L_E = \bigcup_{\varepsilon \in E} L_\varepsilon, L_C = \bigcup_{\gamma \in C} L_\gamma, U_E = \bigcup_{\varepsilon \in E} U_\varepsilon, U_C = \bigcup_{\gamma \in C} U_\gamma.$$

We perform Bilingual Bootstrapping as described in Figure 2. Hereafter, we will only explain the process for English (left-hand side); the process for Chinese (right-hand side) can be conducted similarly.

3.3 Naïve Bayesian Classifier

```
estimate $P_\varepsilon^{(E)}(e|t)$ with MLE using $L_\varepsilon$ as data;
estimate $P_\varepsilon^{(C)}(e|t)$ with EM Algorithm using $L_\gamma$ for each $\gamma \in C_\varepsilon$ as data;
calculate $P_\varepsilon(e|t)$ as a linear combination of $P_\varepsilon^{(E)}(e|t)$ and $P_\varepsilon^{(C)}(e|t)$;
estimate $P_\varepsilon(t)$ with MLE using $L_\varepsilon$;
calculate $P_\varepsilon(e|\bar{t})$ and $P_\varepsilon(\bar{t})$ similarly.
```

Figure 3: Creating Naïve Bayesian Classifier

While we can in principle employ any kind of classifier in BB, we use here a Naïve Bayesian Classifier. At step 1 in BB, we construct the classifier as described in Figure 3. At step 2, for each example **e**, we calculate with the Naïve Bayesian Classifier:

$$\lambda^*(\mathbf{e}) = \max_{t \in T_\varepsilon} \frac{P_\varepsilon(t|\mathbf{e})}{P_\varepsilon(\bar{t}|\mathbf{e})} = \max_{t \in T_\varepsilon} \frac{P_\varepsilon(t) P_\varepsilon(\mathbf{e}|t)}{P_\varepsilon(\bar{t}) P_\varepsilon(\mathbf{e}|\bar{t})}.$$

The second equation is based on Bayes' rule.

In the calculation, we assume that the context words in **e** (i.e., $e_1, e_2, \ldots, e_m$) are independently generated from $P_\varepsilon(e|t)$ and thus we have

$$P_\varepsilon(\mathbf{e}|t) = \prod_{i=1}^{m} P_\varepsilon(e_i|t).$$

We can calculate $P_\varepsilon(\mathbf{e}|\bar{t})$ similarly.

For $P_\varepsilon(e|t)$, we calculate it at step 1 by linearly combining $P_\varepsilon^{(E)}(e|t)$ estimated from English and $P_\varepsilon^{(C)}(e|t)$ estimated from Chinese:

$$P_\varepsilon(e|t) = (1-\alpha-\beta) P_\varepsilon^{(E)}(e|t) + \alpha P_\varepsilon^{(C)}(e|t) + \beta P^{(U)}(e), \qquad (1)$$

where $0 \le \alpha \le 1$, $0 \le \beta \le 1$, $\alpha + \beta \le 1$, and $P^{(U)}(e)$ is a uniform distribution over E, which is used for avoiding zero probability. In this way, we estimate $P_\varepsilon(e|t)$ using information from not only English but also Chinese.

For $P_\varepsilon^{(E)}(e|t)$, we estimate it with MLE (Maximum Likelihood Estimation) using L_ε as data. For $P_\varepsilon^{(C)}(e|t)$, we estimate it as is described in Section 3.4.

3.4 EM Algorithm

For the sake of readability, we rewrite $P_\varepsilon^{(C)}(e|t)$ as $P(e|t)$. We define a finite mixture model of the form $P(c|t) = \sum_{e \in E} P(c|e,t) P(e|t)$ and for a specific ε we assume that the data in

$$L_\gamma = \{(\mathbf{c}_1, t_1)_\gamma, (\mathbf{c}_2, t_2)_\gamma, \ldots, (\mathbf{c}_h, t_h)_\gamma\},$$
$$t_i \in T_\gamma (i = 1, \ldots, h), \ \ \forall \gamma \in C_\varepsilon$$

are independently generated on the basis of the model. We can, therefore, employ the Expectation and Maximization Algorithm (EM Algorithm) (Dempster et al. 1977) to estimate the parameters of the model including $P(e|t)$. We also use the relation *T* in the estimation.

Initially, we set

$$P(c|e,t) = \begin{cases} \dfrac{1}{|C_e|}, & \text{if } c \in C_e \\ 0, & \text{if } c \notin C_e \end{cases},$$

$$P(e|t) = \frac{1}{|E|}, \ \ e \in E.$$

We next estimate the parameters by iteratively updating them ass described in Figure 4 until they converge. Here $f(c,t)$ stands for the frequency of c related to *t*. The context information in Chinese is then 'translated' into that in English through the links in *T*.

E-step: $$P(e|c,t) \leftarrow \frac{P(c|e,t)P(e|t)}{\sum_{e \in E} P(c|e,t)P(e|t)}$$

M-step: $$P(c|e,t) \leftarrow \frac{f(c,t)P(e|c,t)}{\sum_{c \in C} f(c,t)P(e|c,t)}$$

$$P(e|t) \leftarrow \frac{\sum_{c \in C} f(c,t)P(e|c,t)}{\sum_{c \in C} f(c,t)}$$

Figure 4: EM Algorithm

4 Comparison between BB and MB

We note that Monolingual Bootstrapping is a special case of Bilingual Bootstrapping (consider the situation in which α equals 0 in formula (1)). Moreover, it seems safe to say that BB can always perform better than MB.

The many-to-many relationship between the words in the two languages stands out as key to the higher performance of BB.

Suppose that the classifier with respect to 'plant' has two decisions (denoted as A and B in Figure 5). Further suppose that the classifiers with

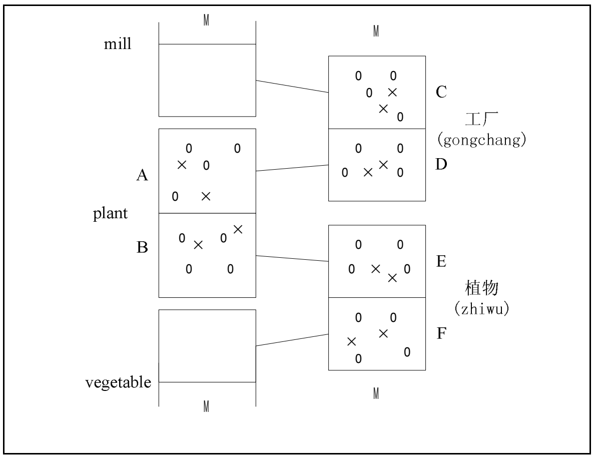

Figure 5: Example of BB

respect to ‘gongchang’ and ‘zhiwu’ in Chinese have two decisions respectively, (C and D) (E and F). A and D are equivalent to each other (i.e., they represent the same sense), and so are B and E.

Assume that examples are classified after several iterations in BB as depicted in Figure 5. Here, circles denote the examples that are correctly classified and crosses denote the examples that are incorrectly classified.

Since A and D are equivalent to each other, we can ‘translate’ the examples with D and use them to boost the performance of classification to A. This is because the misclassified examples (crosses) with D are those mistakenly classified from C and they will not have much negative effect on classification to A, even though the translation from Chinese into English can introduce some noises. Similar explanations can be stated to other classification decisions.

In contrast, MB only uses the examples in A and B to construct a classifier, and when the number of misclassified examples increases (this is inevitable in bootstrapping), its performance will stop improving.

5 Word Translation Disambiguation

5.1 Using Bilingual Bootstrapping

While it is possible to straightforwardly apply the algorithm of BB described in Section 3 to word translation disambiguation, we use here a variant of it for a better adaptation to the task and for a fairer comparison with existing technologies. The variant of BB has four modifications.

(1) It actually employs an ensemble of the Naïve Bayesian Classifiers (NBC), because an ensemble of NBCs generally performs better than a single NBC (Pedersen 2000). In an ensemble, it creates different NBCs using as data the words within different window sizes surrounding the word to be disambiguated (e.g., ‘plant’ or ‘zhiwu’) and further constructs a new classifier by linearly combining the NBCs.

(2) It employs the heuristics of ‘one sense per discourse’ (cf., Yarowsky 1995) after using an ensemble of NBCs.

(3) It uses only classified data in English at the beginning.

(4) It individually resolves ambiguities on selected English words such as ‘plant’, ‘interest’. As a result, in the case of ‘plant’; for example, the classifiers with respect to ‘gongchang’ and ‘zhiwu’ only make classification decisions to D and E but not C and F (in Figure 5). It calculates $\lambda^*(\mathbf{c})$ as $\lambda^*(\mathbf{c}) = P(\mathbf{c} | t)$ and sets $\theta = 0$ at the right-hand side of step 2.

5.2 Using Monolingual Bootstrapping

We consider here two implementations of MB for word translation disambiguation.

In the first implementation, in addition to the basic algorithm of MB, we also use (1) an ensemble of Naïve Bayesian Classifiers, (2) the heuristics of ‘one sense per discourse’, and (3) a small number of classified data in English at the beginning. We will denote this implementation as MB-B hereafter.

The second implementation is different from the first one only in (1). That is, it employs as a classifier a decision list instead of an ensemble of NBCs. This implementation is exactly the one proposed in (Yarowsky 1995), and we will denote it as MB-D hereafter.

MB-B and MB-D can be viewed as the *state-of-the-art* methods for word translation disambiguation using bootstrapping.

6 Experimental Results

Table 1: Data descriptions in Experiment 1

Words	Chinese translations	Corresponding English senses	Seed words
interest	兴趣	readiness to give attention	show
	利息	money paid for the use of money	rate
	股份, 股权	a share in company or business	hold
	利益	advantage, advancement or favor	conflict
line	绳索, 缆绳	a thin flexible object	cut
	行, 句	written or spoken text	write
	线路	telephone connection	telephone
	队伍, 队列	formation of people or things	wait
	界线, 边界	an artificial division	between
	产品, 品种	product	product

Table 2: Data sizes in Experiment 1

Words	Unclassified sentences		Test sentences
	English	Chinese	
interest	1927	8811	2291
line	3666	5398	4148

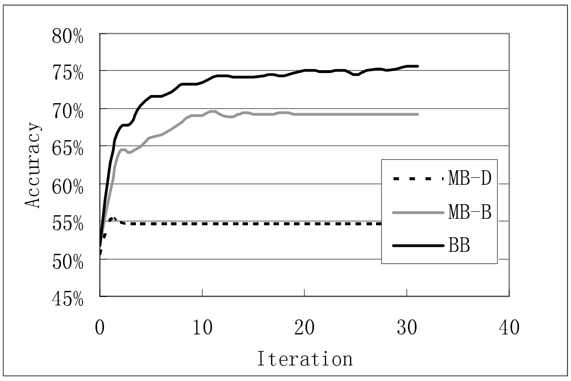

Figure 6: Learning curves with 'interest'

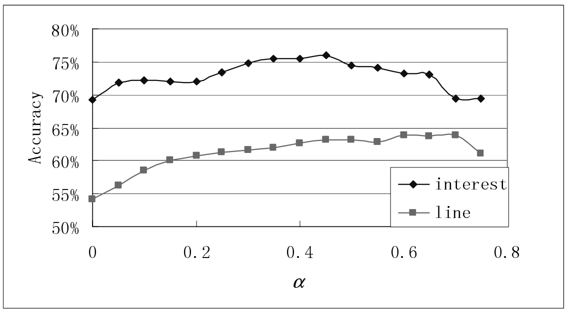

Figure 8: Accuracies of BB with different α

Table 3: Accuracies in Experiment 1

Words	Major (%)	MB-D (%)	MB-B (%)	BB (%)
interest	54.6	54.7	69.3	**75.5**
line	53.5	55.6	54.1	**62.7**

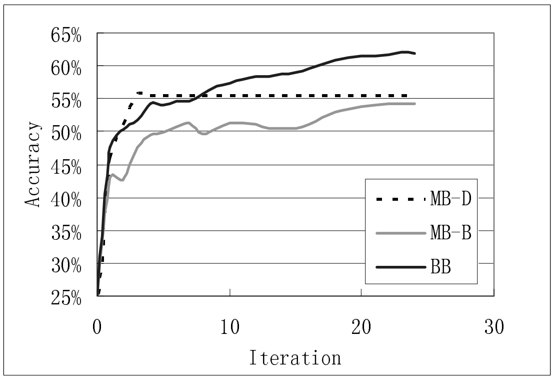

Figure 7: Learning curves with 'line'

Table 4: Accuracies of supervised methods

	interest (%)	line (%)
Ensembles of NBC	89	88
Naïve Bayes	74	72
Decision Tree	78	-
Neural Network	-	76
Nearest Neighbor	87	-

We conducted two experiments on English-Chinese translation disambiguation.

6.1 Experiment 1: WSD Benchmark Data

We first applied BB, MB-B, and MB-D to translation of the English words 'line' and 'interest' using a benchmark data[2]. The data mainly consists of articles in the Wall Street Journal and it is designed for conducting Word Sense Disambiguation (WSD) on the two words (e.g., Pedersen 2000).

We adopted from the HIT dictionary[3] the Chinese translations of the two English words, as listed in Table 1. One sense of the words corresponds to one group of translations.

We then used the benchmark data as our *test data*. (For the word 'interest', we only used its four major senses, because the remaining two minor senses occur in only 3.3% of the data)

[2] http://www.d.umn.edu/~tpederse/data.html.

[3] The dictionary is created by Harbin Institute of Technology.

Table 5: Data descriptions and data sizes in Experiment 2

Words	Chinese translations	Unclassified sentences		Seed words	Test sentences
		English	Chinese		
bass	鱼, 鱼类 / 低音, 低音部	142	8811	fish / music	200
drug	药物, 药品 / 毒品	3053	5398	treatment / smuggler	197
duty	责任, 职责 / 税, 税收	1428	4338	discharge / export	197
palm	棕榈树, 棕榈 / 手掌	366	465	tree / hand	197
plant	工厂, 厂 / 植物	7542	24977	industry / life	197
space	空间, 间隙 / 太空, 宇宙空间	3897	14178	volume / outer	197
tank	坦克 / 水箱, 油箱	417	1400	combat / fuel	199
Total	-	16845	59567	-	1384

As classified data in English, we defined a ‘seed word’ for each group of translations based on our intuition (cf., Table 1). Each of the seed words was then used as a classified ‘sentence’. This way of creating classified data is similar to that in (Yarowsky, 1995). As unclassified data in English, we collected sentences in news articles from a web site (www.news.com), and as unclassified data in Chinese, we collected sentences in news articles from another web site (news.cn.tom.com). We observed that the distribution of translations in the unclassified data was balanced.

Table 2 shows the sizes of the data. Note that there are in general more unclassified sentences in Chinese than in English because an English word usually has several Chinese words as translations (cf., Figure 5).

As a translation dictionary, we used the HIT dictionary, which contains about 76000 Chinese words, 60000 English words, and 118000 links.

We then used the data to conduct translation disambiguation with BB, MB-B, and MB-D, as described in Section 5.

For both BB and MB-B, we used an ensemble of five Naïve Bayesian Classifiers with the window sizes being ±1, ±3, ±5, ±7, ±9 words. For both BB and MB-B, we set the parameters of β, b, and θ to 0.2, 15, and 1.5 respectively. The parameters were tuned based on our preliminary experimental results on MB-B, they were not tuned, however, for BB. For the BB specific parameter α, we set it to 0.4, which meant that we treated the information from English and that from Chinese equally.

Table 3 shows the translation disambiguation accuracies of the three methods as well as that of a baseline method in which we always choose the *major translation*. Figures 6 and 7 show the learning curves of MB-D, MB-B, and BB. Figure 8 shows the accuracies of BB with different α values.

From the results, we see that BB *consistently* and *significantly* outperforms both MB-D and MB-B. The results from the *sign test* are statistically significant (p-value < 0.001).

Table 4 shows the results achieved by some existing *supervised* learning methods with respect to the benchmark data (cf., Pedersen 2000). Although BB is a method nearly equivalent to one based on unsupervised learning, it still performs favorably well when compared with the supervised methods (note that since the experimental settings are different, the results cannot be *directly* compared).

6.2 Experiment 2: Yarowsky’s Words

We also conducted translation on seven of the twelve English words studied in (Yarowsky, 1995). Table 5 shows the list of the words.

For each of the words, we extracted about 200 sentences containing the word from the Encarta[4] English corpus and labeled those sentences with Chinese translations ourselves. We used the labeled sentences as test data and the remaining sentences as unclassified data in English. We also used the sentences in the Great Encyclopedia[5] Chinese corpus as unclassified data in Chinese. We defined, for each translation,

[4] http://encarta.msn.com/default.asp

[5] http://www.whlib.ac.cn/sjk/bkqs.htm

Table 6: Accuracies in Experiment 2

Words	Major (%)	MB-D (%)	MB-B (%)	BB (%)
bass	61.0	57.0	87.0	**89.0**
drug	77.7	78.7	79.7	**86.8**
duty	86.3	**86.8**	72.0	75.1
palm	82.2	80.7	83.3	**92.4**
plant	71.6	89.3	95.4	**95.9**
space	64.5	71.6	84.3	**87.8**
tank	60.3	62.8	76.9	**84.4**
Total	71.9	75.2	82.6	**87.4**

Table 7: Top words for '利息' of 'interest'

MB-B	BB
payment	saving
cut	payment
earn	benchmark
short	whose
short-term	base
yield	prefer
u.s.	fixed
margin	debt
benchmark	annual
regard	dividend

a seed word in English as a classified example (cf., Table 5).

We did not, however, conduct translation disambiguation on the words 'crane', 'sake', 'poach', 'axes', and 'motion', because the first four words do not frequently occur in the Encarta corpus, and the accuracy of choosing the major translation for the last word has already exceeded 98%.

We next applied BB, MB-B, and MB-D to word translation disambiguation. The experiment settings were the same as those in Experiment 1.

From Table 6, we see again that BB *significantly* outperforms MB-D and MB-B. (We will describe the results in detail in the full version of this paper.) Note that the results of MB-D here cannot be *directly* compared with those in (Yarowsky, 1995), mainly because the data used are different.

6.3 Discussions

We investigated the reason of BB's outperforming MB and found that the explanation on the reason in Section 4 appears to be true according to the following observations.

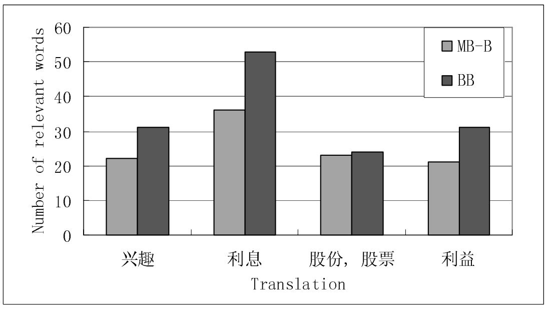

Figure 9: Number of relevant words

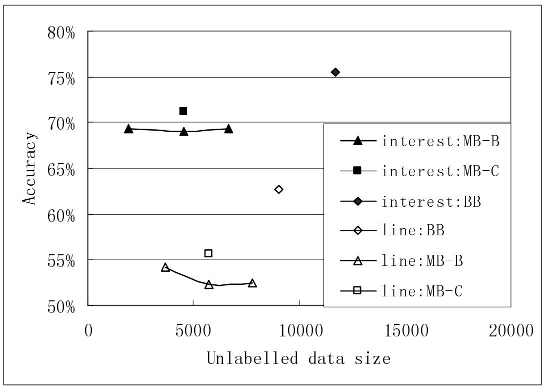

Figure 10: When more unlabeled data available

(1) In a Naïve Bayesian Classifier, words having large values of probability ratio $\frac{P(e|t)}{P(e|\bar{t})}$ have strong influence on the classification of t when they occur, particularly, when they frequently occur. We collected the words having large values of probability ratio for each t in both BB and MB-B and found that BB obviously has more 'relevant words' than MB-B. Here 'relevant words' for t refer to the words which are strongly indicative to t on the basis of human judgments.

Table 7 shows the top ten words in terms of probability ratio for the '利息' translation ('money paid for the use of money') with respect to BB and MB-B, in which relevant words are underlined. Figure 9 shows the numbers of relevant words for the four translations of 'interest' with respect to BB and MB-B.

(2) From Figure 8, we see that the performance of BB remains high or gets higher when α becomes larger than 0.4 (recall that β was fixed to 0.2). This result strongly indicates that the information from Chinese has positive effects on disambiguation.

(3) One may argue that the higher performance of BB might be attributed to the larger unclassified data size it uses, and thus if we increase the

unclassified data size for MB, it is likely that MB can perform as well as BB.

We conducted an additional experiment and found that this is not the case. Figure 10 shows the accuracies achieved by MB-B when data sizes increase. Actually, the accuracies of MB-B cannot further improve when unlabeled data sizes increase. Figure 10 plots again the results of BB as well as those of a method referred to as MB-C. In MB-C, we linearly combine two MB-B classifiers constructed with two different unlabeled data sets and we found that although the accuracies get some improvements in MB-C, they are still much lower than those of BB.

7 Conclusion

This paper has presented a new word translation disambiguation method using a bootstrapping technique called Bilingual Bootstrapping. Experimental results indicate that BB significantly outperforms the existing Monolingual Bootstrapping technique in word translation disambiguation. This is because BB can effectively make use of information from two sources rather than from one source as in MB.

Acknowledgements

We thank Ming Zhou, Ashley Chang and Yao Meng for their valuable comments on an early draft of this paper.

References

P. Brown, S. D. Pietra, V. D. Pietra, and R. Mercer, 1991. Word Sense Disambiguation Using Statistical Methods. In *Proceedings of the 29th Annual Meeting of the Association for Computational Linguistics*, pp. 264-270.

I. Dagan and A. Itai, 1994. Word Sense Disambiguation Using a Second Language Monolingual Corpus. *Computational Linguistics*, vol. 20, pp. 563-596.

A. P. Dempster, N. M. Laird, and D. B. Rubin, 1977. Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society B*, vol. 39, pp. 1-38.

G. Escudero, L. Marquez, and G. Rigau, 2000. Boosting Applied to Word Sense Disambiguation. In *Proceedings of the 12th European Conference on Machine Learning*.

W. Gale, K. Church, and D. Yarowsky, 1992a. A Method for Disambiguating Word Senses in a Large Corpus. *Computers and Humanities*, vol. 26, pp. 415-439.

W. Gale, K. Church, and D. Yarowsky, 1992b. One sense per discourse. In *Proceedings of DARPA speech and Natural Language Workshop*.

A. R. Golding and D. Roth, 1999. A Winnow-Based Approach to Context-Sensitive Spelling Correction. *Machine Learning*, vol. 34, pp. 107-130.

G. Kikui, 1999. Resolving Translation Ambiguity Using Non-parallel Bilingual Corpora. In *Proceedings of ACL '99 Workshop on Unsupervised Learning in Natural Language Processing*.

L. Mangu and E. Brill, 1997. Automatic rule acquisition for spelling correction. In *Proceedings of the 14th International Conference on Machine Learning*.

R. Mihalcea and D. Moldovan, 1999. A method for Word Sense Disambiguation of unrestricted text. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*.

H. T. Ng and H. B. Lee, 1996. Integrating Multiple Knowledge Sources to Disambiguate Word Sense: An Exemplar-based Approach. In *Proceedings of the 34th Annual Meeting of the Association for Computational Linguistics*, pp. 40-47.

T. Pedersen and R. Bruce, 1997. Distinguishing Word Senses in Untagged Text. In *Proceedings of the 2nd Conference on Empirical Methods in Natural Language Processing*, pp. 197-207.

T. Pedersen, 2000. A Simple Approach to Building Ensembles of Naïve Bayesian Classifiers for Word Sense Disambiguation. In *Proceedings of the 1st Meeting of the North American Chapter of the Association for Computational Linguistics*.

H. Schutze, 1998. Automatic Word Sense Discrimination. In *Computational Linguistics*, vol. 24, no. 1, pp. 97-124.

G. Towell and E. Voothees, 1998. Disambiguating Highly Ambiguous Words. *Computational Linguistics*, vol. 24, no. 1, pp. 125-146.

D. Yarowsky, 1994. Decision Lists for Lexical Ambiguity Resolution: Application to Accent Restoration in Spanish and French. In *Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics*, pp. 88-95.

D. Yarowsky, 1995. Unsupervised Word Sense Disambiguation Rivaling Supervised Methods. In *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics*, pp. 189-196.