Synonymous Collocation Extraction Using Translation Information

Hua WU, Ming ZHOU
Microsoft Research Asia
5F Sigma Center, No.49 Zhichun Road, Haidian District
Beijing, 100080, China
wu_hua_@msn.com, mingzhou@microsoft.com

Abstract

Automatically acquiring synonymous collocation pairs such as <turn on, OBJ, light> and <switch on, OBJ, light> from corpora is a challenging task. For this task, we can, in general, have a large monolingual corpus and/or a very limited bilingual corpus. Methods that use monolingual corpora alone or use bilingual corpora alone are apparently inadequate because of low precision or low coverage. In this paper, we propose a method that uses both these resources to get an optimal compromise of precision and coverage. This method first gets candidates of synonymous collocation pairs based on a monolingual corpus and a word thesaurus, and then selects the appropriate pairs from the candidates using their translations in a second language. The translations of the candidates are obtained with a statistical translation model which is trained with a small bilingual corpus and a large monolingual corpus. The translation information is proved as effective to select synonymous collocation pairs. Experimental results indicate that the average precision and recall of our approach are 74% and 64% respectively, which outperform those methods that only use monolingual corpora and those that only use bilingual corpora.

1 Introduction

This paper addresses the problem of automatically extracting English synonymous collocation pairs using translation information. A synonymous collocation pair includes two collocations which are similar in meaning, but not identical in wording. Throughout this paper, the term *collocation* refers to a lexically restricted word pair with a certain syntactic relation. For instance, <turn on, OBJ, light> is a collocation with a syntactic relation verb-object, and <turn on, OBJ, light> and <switch on, OBJ, light> are a synonymous collocation pair. In this paper, translation information means translations of collocations and their translation probabilities.

Synonymous collocations can be considered as an extension of the concept of synonymous expressions which conventionally include synonymous words, phrases and sentence patterns. Synonymous expressions are very useful in a number of NLP applications. They are used in information retrieval and question answering (Kiyota et al., 2002; Dragomia et al., 2001) to bridge the expression gap between the query space and the document space. For instance, “buy book” extracted from the users’ query should also in some way match “order book” indexed in the documents. Besides, the synonymous expressions are also important in language generation (Langkilde and Knight, 1998) and computer assisted authoring to produce vivid texts.

Up to now, there have been few researches which directly address the problem of extracting synonymous collocations. However, a number of studies investigate the extraction of synonymous words from monolingual corpora (Carolyn et al., 1992; Grefenstatte, 1994; Lin, 1998; Gasperin et al., 2001). The methods used the contexts around the investigated words to discover synonyms. The problem of the methods is that the precision of the extracted synonymous words is low because it extracts many word pairs such as “cat” and “dog”, which are similar but not synonymous. In addition, some studies investigate the extraction of synonymous words and/or patterns from bilingual corpora (Barzilay and Mckeown, 2001; Shimohata and Sumita, 2002). However, these methods can only extract synonymous expressions which occur in the bilingual corpus. Due to the limited size of the bilingual corpus, the coverage of the extracted expressions is very low.

Given the fact that we usually have large mono-

lingual corpora (unlimited in some sense) and very limited bilingual corpora, this paper proposes a method that tries to make full use of these different resources to get an optimal compromise of precision and coverage for synonymous collocation extraction. We first obtain candidates of synonymous collocation pairs based on a monolingual corpus and a word thesaurus. We then select those appropriate candidates using their translations in a second language. Each translation of the candidates is assigned a probability with a statistical translation model that is trained with a small bilingual corpus and a large monolingual corpus. The similarity of two collocations is estimated by computing the similarity of their vectors constructed with their corresponding translations. Those candidates with larger similarity scores are extracted as synonymous collocations. The basic assumption behind this method is that two collocations are synonymous if their translations are similar. For example, <turn on, OBJ, light> and <switch on, OBJ, light> are synonymous because both of them are translated into <开, OBJ, 灯> (<kai1, OBJ, deng1>) and <打开, OBJ, 灯> (<da3 kai1, OBJ, deng1>) in Chinese.

In order to evaluate the performance of our method, we conducted experiments on extracting three typical types of synonymous collocations. Experimental results indicate that our approach achieves 74% average precision and 64% recall respectively, which considerably outperform those methods that only use monolingual corpora or only use bilingual corpora.

The remainder of this paper is organized as follows. Section 2 describes our synonymous collocation extraction method. Section 3 evaluates the proposed method, and the last section draws our conclusion and presents the future work.

2 Our Approach

Our method for synonymous collocation extraction comprises of three steps: (1) extract collocations from large monolingual corpora; (2) generate candidates of synonymous collocation pairs with a word thesaurus WordNet; (3) select synonymous collocation candidates using their translations.

2.1 Collocation Extraction

This section describes how to extract English collocations. Since Chinese collocations will be used to train the language model in Section 2.3, they are also extracted in the same way.

Collocations in this paper take some syntactical relations (dependency relations), such as <verb, OBJ, noun>, <noun, ATTR, adj>, and <verb, MOD, adv>. These dependency triples, which embody the syntactic relationship between words in a sentence, are generated with a parser—we use NLPWIN in this paper[1]. For example, the sentence "She owned this red coat" is transformed to the following four triples after parsing: <own, SUBJ, she>, <own, OBJ, coat>, <coat, DET, this>, and <coat, ATTR, red>. These triples are generally represented in the form of <Head, Relation Type, Modifier>.

The measure we use to extract collocations from the parsed triples is weighted mutual information (WMI) (Fung and Mckeown, 1997), as described as

$$WMI(w_1,r,w_2) = p(w_1,r,w_2)\log\frac{p(w_1,r,w_2)}{p(w_1 \mid r)p(w_2 \mid r)p(r)}$$

Those triples whose WMI values are larger than a given threshold are taken as collocations. We do not use the point-wise mutual information because it tends to overestimate the association between two words with low frequencies. Weighted mutual information meliorates this effect by adding $p(w_1, r, w_2)$.

For expository purposes, we will only look into three kinds of collocations for synonymous collocation extraction: <verb, OBJ, noun>, <noun, ATTR, adj> and <verb, MOD, adv>.

Table 1. English Collocations

Class	#Type	#Token
verb, OBJ, noun	506,628	7,005,455
noun, ATTR, adj	333,234	4,747,970
verb, Mod, adv	40,748	483,911

Table 2. Chinese Collocations

Class	#Type	#Token
verb, OBJ, noun	1,579,783	19,168,229
noun, ATTR, adj	311,560	5,383,200
verb, Mod, adv	546,054	9,467,103

The English collocations are extracted from Wall Street Journal (1987-1992) and Association Press (1988-1990), and the Chinese collocations are

[1] The NLPWIN parser is developed at Microsoft Research, which parses several languages including Chinese and English. Its output can be a phrase structure parse tree or a logical form which is represented with dependency triples.

extracted from People Daily (1980-1998). The statistics of the extracted collocations are shown in Table 1 and 2. The thresholds are set as 5 for both English and Chinese. *Token* refers to the total number of collocation occurrences and *Type* refers to the number of unique collocations in the corpus.

2.2 Candidate Generation

Candidate generation is based on the following assumption: For a collocation <Head, Relation Type, Modifier>, its synonymous expressions also take the form of <Head, Relation Type, Modifier> although sometimes they may also be a single word or a sentence pattern.

The synonymous candidates of a collocation are obtained by expanding a collocation <Head, Relation Type, Modifier> using the synonyms of *Head* and *Modifier*. The synonyms of a word are obtained from WordNet 1.6. In WordNet, one synset consists of several synonyms which represent a single sense. Therefore, polysemous words occur in more than one synsets. The synonyms of a given word are obtained from all the synsets including it. For example, the word "turn on" is a polysemous word and is included in several synsets. For the sense "cause to operate by flipping a switch", "switch on" is one of its synonyms. For the sense "be contingent on", "depend on" is one of its synonyms. We take both of them as the synonyms of "turn on" regardless of its meanings since we do not have sense tags for words in collocations.

If we use C_w to indicate the synonym set of a word *w* and U to denote the English collocation set generated in Section 2.1. The detail algorithm on generating candidates of synonymous collocation pairs is described in Figure 1. For example, given a collocation <turn on, OBJ, light>, we expand "turn on" to "switch on", "depend on", and then expand "light" to "lump", "illumination". With these synonyms and the relation type OBJ, we generate synonymous collocation candidates of <turn on, OBJ, light>. The candidates are <switch on, OBJ, light>, <turn on, OBJ, lump>, <depend on, OBJ, illumination>, <depend on, OBJ, light> etc. Both these candidates and the original collocation <turn on, OBJ, light> are used to generate the synonymous collocation pairs.

With the above method, we obtained candidates of synonymous collocation pairs. For example, <switch on, OBJ, light> and <turn on, OBJ, light> are a synonymous collocation pair. However, this method also produces wrong synonymous collocation candidates. For example, <depend on, OBJ, illumination> and <turn on, OBJ, light> is not a synonymous pair. Thus, it is important to filter out these inappropriate candidates.

(1) For each collocation (Col_i=<Head, R, Modifier>)$\in$U, do the following:
   a. Use the synonyms in WordNet 1.6 to expand *Head* and *Modifier* and get their synonym sets C_{Head} and $C_{Modifier}$
   b. Generate the candidate set of its synonymous collocations S_i={<w_1, *R*, w_2> | $w_1 \in$ {Head} $\cup$ C_{Head} & $w_2 \in$ {Modifier} $\cup$ $C_{Modifier}$ & <w_1, *R*, w_2> $\in$U & <w_1, *R*, w_2> $\neq$ Col_i }
(2) Generate the candidate set of synonymous collocation pairs SC= {(Col_i, Col_j)| $Col_i \in$ U & $Col_j \in S_i$ }

Figure 1. Candidate Set Generation Algorithm

2.3 Candidate Selection

In synonymous word extraction, the similarity of two words can be estimated based on the similarity of their contexts. However, this method cannot be effectively extended to collocation similarity estimation. For example, in sentences "They turned on the lights" and "They depend on the illumination", the meaning of two collocations <turn on, OBJ, light> and <depend on, OBJ, illumination> are different although their contexts are the same. Therefore, monolingual information is not enough to estimate the similarity of two collocations. However, the meanings of the above two collocations can be distinguished if they are translated into a second language (e.g., Chinese). For example, <turn on, OBJ, light> is translated into <开, OBJ, 灯> (<kai1, OBJ, deng1) and <打开, OBJ, 灯> (<da3 kai1, OBJ, deng1>) in Chinese while <depend on, OBJ, illumination> is translated into <取决于, OBJ, 光照度> (qu3 jue2 yu2, OBJ, guang1 zhao4 du4>). Thus, they are not synonymous pairs because their translations are completely different.

In this paper, we select the synonymous collocation pairs from the candidates in the following way. First, given a candidate of synonymous collocation pair generated in section 2.2, we translate the two collocations into Chinese with a simple statistical translation model. Second, we calculate the similarity of two collocations with the feature vectors constructed with their translations. A candidate is selected as a synonymous collocation pair

if its similarity exceeds a certain threshold.

2.3.1 *Collocation Translation*

For an English collocation e_{col}=<e_1, r_e, e_2>, we translate it into Chinese collocations[2] using an English-Chinese dictionary. If the translation sets of e_1 and e_2 are represented as CS_1 and CS_2 respectively, the Chinese translations can be represented as S={<c_1, r_c, c_2>| $c_1 \in CS_1$, $c_2 \in CS_2$, $r_c \in R$}, with R denoting the relation set.

Given an English collocation e_{col}=<e_1, r_e, e_2> and one of its Chinese collocation c_{col}=<c_1, r_c, c_2>$\in$S, the probability that e_{col} is translated into c_{col} is calculated as in Equation (1).

$$p(c_{col} \mid e_{col}) = \frac{p(e_1, r_e, e_2 \mid c_1, r_c, c_2)\, p(c_1, r_c, c_2)}{p(e_{col})} \quad (1)$$

According to Equation (1), we need to calculate the translation probability $p(e_{col}|c_{col})$ and the target language probability $p(c_{col})$. Calculating the translation probability needs a bilingual corpus. If the above equation is used directly, we will run into the data sparseness problem. Thus, model simplification is necessary.

2.3.2 *Translation Model*

Our simplification is made according to the following three assumptions.

Assumption 1: For a Chinese collocation c_{col} and r_e, we assume that e_1 and e_2 are conditionally independent. The translation model is rewritten as:

$$\begin{aligned} p(e_{col} \mid c_{col}) &= p(e_1, r_e, e_2 \mid c_{col}) \\ &= p(e_1 \mid r_e, c_{col})\, p(e_2 \mid r_e, c_{col})\, p(r_e \mid c_{col}) \end{aligned} \quad (2)$$

Assumption 2: Given a Chinese collocation <c_1, r_c, c_2>, we assume that the translation probability $p(e_i|c_{col})$ only depends on e_i and c_i (i=1,2), and $p(r_e|c_{col})$ only depends on r_e and r_c. Equation (2) is rewritten as:

$$\begin{aligned} p(e_{col} \mid c_{col}) &= p(e_1 \mid c_{col})\, p(e_2 \mid c_{col})\, p(r_e \mid c_{col}) \\ &= p(e_1 \mid c_1)\, p(e_2 \mid c_2)\, p(r_e \mid r_c) \end{aligned} \quad (3)$$

It is equal to a word translation model if we take the relation type in the collocations as an element like a word, which is similar to Model 1 in (Brown et al., 1993).

Assumption 3: We assume that one type of English collocation can only be translated to the same type of Chinese collocations[3]. Thus, $p(r_e| r_c) = 1$ in our case. Equation (3) is rewritten as:

$$\begin{aligned} p(e_{col} \mid c_{col}) &= p(e_1 \mid c_1)\, p(e_2 \mid c_2)\, p(r_e \mid r_c) \\ &= p(e_1 \mid c_1)\, p(e_2 \mid c_2) \end{aligned} \quad (4)$$

2.3.3 *Language Model*

The language model $p(c_{col})$ is calculated with the Chinese collocation database extracted in section 2.1. In order to tackle with the data sparseness problem, we smooth the language model with an interpolation method.

When the given Chinese collocation occurs in the corpus, we calculate it as in (5).

$$p(c_{col}) = \frac{count(c_{col})}{N} \quad (5)$$

where $count(c_{col})$ represents the count of the Chinese collocation c_{col}. N represents the total counts of all the Chinese collocations in the training corpus.

For a collocation <c_1, r_c, c_2>, if we assume that two words c_1 and c_2 are conditionally independent given the relation r_c, Equation (5) can be rewritten as in (6).

$$p(c_{col}) = p(c_1 \mid r_c)\, p(c_2 \mid r_c)\, p(r_c) \quad (6)$$

where $p(c_1 \mid r_c) = \frac{count(c_1, r_c, *)}{count(*, r_c, *)}$

$p(c_2 \mid r_c) = \frac{count(*, r_c, c_2)}{count(*, r_c, *)}$, $p(r_c) = \frac{count(*, r_c, *)}{N}$

$count(c_1, r_c, *)$: frequency of the collocations with c_1 as the head and r_c as the relation type.

$count(*, r_c, c_2)$: frequency of the collocations with c_2 as the modifier and r_c as the relation type

$count(*, r_c, *)$: frequency of the collocations with r_c as the relation type.

With Equation (5) and (6), we get the interpolated language model as shown in (7).

$$p(c_{col}) = \lambda \frac{count(c_{col})}{N} + (1 - \lambda)\, p(c_1 \mid r_c)\, p(c_2 \mid r_c)\, p(r_c) \quad (7)$$

where $0 < \lambda < 1$. λ is a constant so that the probabilities sum to 1.

---

[2] Some English collocations can be translated into Chinese words, phrases or patterns. Here we only consider the case of being translated into collocations.

[3] Zhou et al. (2001) found that about 70% of the Chinese translations have the same relation type as the source English collocations.

2.3.4 *Word Translation Probability Estimation*

Many methods are used to estimate word translation probabilities from unparallel or parallel bilingual corpora (Koehn and Knight, 2000; Brown et al., 1993). In this paper, we use a parallel bilingual corpus to train the word translation probabilities based on the result of word alignment with a bilingual Chinese-English dictionary. The alignment method is described in (Wang et al., 2001). In order to deal with the problem of data sparseness, we conduct a simple smoothing by adding 0.5 to the counts of each translation pair as in (8).

$$p(e \mid c) = \frac{count(e,c) + 0.5}{count(c) + 0.5 * |trans_e|} \quad (8)$$

where $|trans_e|$ represents the number of English translations for a given Chinese word *c*.

2.3.5 *Collocation Similarity Calculation*

For each synonymous collocation pair, we get its corresponding Chinese translations and calculate the translation probabilities as in section 2.3.1. These Chinese collocations with their corresponding translation probabilities are taken as feature vectors of the English collocations, which can be represented as:

$$Fe^{i}_{col} = <(c^{i1}_{col}, p^{i1}_{col}), (c^{i2}_{col}, p^{i2}_{col}), \ldots, (c^{im}_{col}, p^{im}_{col})>$$

The similarity of two collocations is defined as in (9). The candidate pairs whose similarity scores exceed a given threshold are selected.

$$sim(e^{1}_{col}, e^{2}_{col}) = \cos(Fe^{1}_{col}, Fe^{2}_{col}) = \frac{\sum_{c^{1i}_{col} = c^{2j}_{col}} p^{1i}_{col} * p^{2j}_{col}}{\sqrt{\sum_{i} (p^{1i}_{col})^2} * \sqrt{\sum_{j} (p^{2j}_{col})^2}} \quad (9)$$

For example, given a synonymous collocation pair <turn on, OBJ, light> and <switch on, OBJ, light>, we first get their corresponding feature vectors.

The feature vector of <turn on, OBJ, light>:

< (<开, OBJ, 灯>, 0.04692), (<打开, OBJ, 灯>, 0.01602), … , (<依赖, OBJ, 光>, 0.0002710), (<依赖, OBJ, 光照度>, 0.0000305) >

The feature vector of <switch on, OBJ, light>:

< (<打开, OBJ, 灯>, 0.04238), (<开, OBJ, 灯>, 0.01257), (<打开, OBJ, 灯光>, 0.002531), … , (<开, OBJ, 信号灯>, 0.00003542) >

The values in the feature vector are translation probabilities. With these two vectors, we get the similarity of <turn on, OBJ, light> and <switch on, OBJ, light>, which is 0.2348.

2.4 Implementation of our Approach

We use an English-Chinese dictionary to get the Chinese translations of collocations, which includes 219,404 English words. Each source word has 3 translation words on average. The word translation probabilities are estimated from a bilingual corpus that obtains 170,025 pairs of Chinese-English sentences, including about 2.1 million English words and about 2.5 million Chinese words.

With these data and the collocations in section 2.1, we produced 93,523 synonymous collocation pairs and filtered out 1,060,788 candidate pairs with our translation method if we set the similarity threshold to 0.01.

3 Evaluation

To evaluate the effectiveness of our methods, two experiments have been conducted. The first one is designed to compare our method with two methods that use monolingual corpora. The second one is designed to compare our method with a method that uses a bilingual corpus.

3.1 Comparison with Methods using Monolingual Corpora

We compared our approach with two methods that use monolingual corpora. These two methods also employed the candidate generation described in section 2.2. The difference is that the two methods use different strategies to select appropriate candidates. The training corpus for these two methods is the same English one as in Section 2.1.

3.1.1 *Method Description*

Method 1: This method uses monolingual contexts to select synonymous candidates. The purpose of this experiment is to see whether the context method for synonymous word extraction can be effectively extended to synonymous collocation extraction.

The similarity of two collocations is calculated with their feature vectors. The feature vector of a collocation is constructed by all words in sentences which surround the given collocation. The context vector for collocation *i* is represented as in (10).

$$Fe_{col}^{i} = < (w_{i1}, p_{i1}), (w_{i2}, p_{i2}), ..., (w_{im}, p_{im}) > \quad (10)$$

where $p_{ij} = \frac{count(w_{ij}, e_{col}^{i})}{N}$

w_{ij} : context word j of collocation i.

p_{ij} : probability of w_{ij} co-occurring with e_{col}^{i}.

$count(w_{ij}, e_{col}^{i})$: frequency of the context word w_{ij} co-occurring with the collocation e_{col}^{i}

N: all counts of the words in the training corpus.

With the feature vectors, the similarity of two collocations is calculated as in (11). Those candidates whose similarities exceed a given threshold are selected as synonymous collocations.

$$sim(e_{col}^{1}, e_{col}^{2}) = \cos(Fe_{col}^{1}, Fe_{col}^{2}) = \frac{\sum_{w_{1i}=w_{2j}} p_{1i} * p_{2j}}{\sqrt{\sum_{i} (p_{1i})^2} * \sqrt{\sum_{j} (p_{2j})^2}} \quad (11)$$

Method 2: Instead of using contexts to calculate the similarity of two words, this method calculates the similarity of collocations with the similarity of their components. The formula is described in Equation (12).

$$sim(e_{col}^{1}, e_{col}^{2}) = sim(e_1^1, e_1^2) * sim(e_2^1, e_2^2) * sim(rel^1, rel^2) \quad (12)$$

where $e_{col}^{i} = (e_1^i, rel^i, e_2^i)$. We assume that the relation type keeps the same, so $sim(rel^1, rel^2) = 1$.

The similarity of the words is calculated with the same method as described in (Lin, 1998), which is rewritten in Equation (13). The similarity of the words is calculated through the surrounding context words which have dependency relationships with the investigated words.

$$Sim(e_1, e_2) = \frac{\sum_{(rel,e) \in T(e_1) \cap T(e_2)} (w(e_1, rel, e) + w(e_2, rel, e))}{\sum_{(rel,e) \in T(e_1)} w(e_1, rel, e) + \sum_{(rel,e) \in T(e_2)} w(e_2, rel, e)} \quad (13)$$

where $T(e_i)$ denotes the set of words which have the dependency relation rel with e_i.

$$w(e_i, rel, e_j) = p(e_i, rel, e_j) \log \frac{p(e_i, rel, e_j)}{p(e_i \mid rel) p(e_j \mid rel) p(rel)}$$

3.1.2 *Test Set*

With the candidate generation method as depicted in section 2.2, we generated 1,154,311 candidates of synonymous collocations pairs for 880,600 collocations, from which we randomly selected 1,300 pairs to construct a test set. Each pair was evaluated independently by two judges to see if it is synonymous. Only those agreed upon by two judges are considered as synonymous pairs. The statistics of the test set is shown in Table 3. We evaluated three types of synonymous collocations: <verb, OBJ, noun>, <noun, ATTR, adj>, <verb, MOD, adv>. For the type <verb, OBJ, noun>, among the 630 synonymous collocation candidate pairs, 197 pairs are correct. For <noun, ATTR, adj>, 163 pairs (among 324 pairs) are correct, and for <verb, MOD, adv>, 124 pairs (among 346 pairs) are correct.

Table 3. The Test Set

Type	#total	#correct
verb, OBJ, noun	630	197
noun, ATTR, adj	324	163
verb, MOD, adv	346	124

3.1.3 *Evaluation Results*

With the test set, we evaluate the performance of each method. The evaluation metrics are precision, recall, and f-measure.

A development set including 500 synonymous pairs is used to determine the thresholds of each method. For each method, the thresholds for getting highest f-measure scores on the development set are selected. As the result, the thresholds for Method 1, Method 2 and our approach are 0.02, 0.02, and 0.01 respectively. With these thresholds, the experimental results on the test set in Table 3 are shown in Table 4, Table 5 and Table 6.

Table 4. Results for <verb, OBJ, noun>

Method	Precision	Recall	F-measure
Method 1	0.3148	0.8934	0.4656
Method 2	0.3886	0.7614	0.5146
Ours	0.6811	0.6396	0.6597

Table 5. Results for <noun, ATTR, adj>

Method	Precision	Recall	F-measure
Method 1	0.5161	0.9816	0.6765
Method 2	0.5673	0.8282	0.6733
Ours	0.8739	0.6380	0.7376

Table 6. Results for <verb, MOD, adv>

Method	Precision	Recall	F-measure
Method 1	0.3662	0.9597	0.5301
Method 2	0.4163	0.7339	0.5291
Ours	0.6641	0.7016	0.6824

It can be seen that our approach gets the highest precision (74% on average) for all the three types of synonymous collocations. Although the recall (64% on average) of our approach is below other methods, the f-measure scores, which combine both precision and recall, are the highest. In order to compare our methods with other methods under the same recall value, we conduct another experiment on the type <verb, OBJ, noun>[4]. We set the recalls of the two methods to the same value of our method, which is 0.6396 in Table 4. The precisions are 0.3190, 0.4922, and 0.6811 for Method 1, Method 2, and our method, respectively. Thus, the precisions of our approach are higher than the other two methods even when their recalls are the same. It proves that our method of using translation information to select the candidates is effective for synonymous collocation extraction.

The results of Method 1 show that it is difficult to extract synonymous collocations with monolingual contexts. Although Method 1 gets higher recalls than the other methods, it brings a large number of wrong candidates, which results in lower precision. If we set higher thresholds to get comparable precision, the recall is much lower than that of our approach. This indicates that the contexts of collocations are not discriminative to extract synonymous collocations.

The results also show that Model 2 is not suitable for the task. The main reason is that both high scores of $sim(e_1^1, e_1^2)$ and $sim(e_2^1, e_2^2)$ does not mean the high similarity of the two collocations.

The reason that our method outperforms the other two methods is that when one collocation is translated into another language, its translations indirectly disambiguate the words' senses in the collocation. For example, the probability of <turn on, OBJ, light> being translated into <打开, OBJ, 灯> (<da3 kai1, OBJ, deng1>) is much higher than that of it being translated into <取决于, OBJ, 光照度> (<qu3 jue2 yu2, OBJ, guang1 zhao4 du4>) while the situation is reversed for <depend on, OBJ, illumination>. Thus, the similarity between <turn on, OBJ, light> and <depend on, OBJ, illumination> is low and, therefore, this candidate is filtered out.

3.2 Comparison with Methods using Bilingual Corpora

Barzilay and Mckeown (2001), and Shimohata and Sumita (2002) used a bilingual corpus to extract synonymous expressions. If the same source expression has more than one different translation in the second language, these different translations are extracted as synonymous expressions. In order to compare our method with these methods that only use a bilingual corpus, we implement a method that is similar to the above two studies. The detail process is described in Method 3.

Method 3: The method is described as follows: (1) All the source and target sentences (here Chinese and English, respectively) are parsed; (2) extract the Chinese and English collocations in the bilingual corpus; (3) align Chinese collocations c_{col}=<c_1, r_c, c_2> and English collocations e_{col}=<e_1, r_e, e_2> if c_1 is aligned with e_1 and c_2 is aligned with e_2; (4) obtain two English synonymous collocations if two different English collocations are aligned with the same Chinese collocation and if they occur more than once in the corpus.

The training bilingual corpus is the same one described in Section 2. With Method 3, we get 9,368 synonymous collocation pairs in total. The number is only 10% of that extracted by our approach, which extracts 93,523 pairs with the same bilingual corpus. In order to evaluate Method 3 and our approach on the same test set. We randomly select 100 collocations which have synonymous collocations in the bilingual corpus. For these 100 collocations, Method 3 extracts 121 synonymous collocation pairs, where 83% (100 among 121) are correct[5]. Our method described in Section 2 generates 556 synonymous collocation pairs with a threshold set in the above section, where 75% (417 among 556) are correct.

If we set a higher threshold (0.08) for our method, we get 360 pairs where 295 are correct (82%). If we use |A|, |B|, |C| to denote correct pairs extracted by Method 3, our method, both Method 3 and our method respectively, we get |A|=100, |B|=295, and $|C| = |A| \cap |B| = 78$. Thus, the synonymous collocation pairs extracted by our method cover 78% ($|C|/|A|$) of those extracted by Method

[4] The results of the other two types of collocations are the same as <verb, OBJ, noun>. We omit them because of the space limit.

[5] These synonymous collocation pairs are evaluated by two judges and only those agreed on by both are selected as correct pairs.

3 while those extracted by Method 3 only cover 26% ($|C|/|B|$) of those extracted by our method.

It can be seen that the coverage of Method 3 is much lower than that of our method even when their precisions are set to the same value. This is mainly because Method 3 can only extract synonymous collocations which occur in the bilingual corpus. In contrast, our method uses the bilingual corpus to train the translation probabilities, where the translations are not necessary to occur in the bilingual corpus. The advantage of our method is that it can extract synonymous collocations not occurring in the bilingual corpus.

4 Conclusions and Future Work

This paper proposes a novel method to automatically extract synonymous collocations by using translation information. Our contribution is that, given a large monolingual corpus and a very limited bilingual corpus, we can make full use of these resources to get an optimal compromise of precision and recall. Especially, with a small bilingual corpus, a statistical translation model is trained for the translations of synonymous collocation candidates. The translation information is used to select synonymous collocation pairs from the candidates obtained with a monolingual corpus. Experimental results indicate that our approach extracts synonymous collocations with an average precision of 74% and recall of 64%. This result significantly outperforms those of the methods that only use monolingual corpora, and that only use a bilingual corpus.

Our future work will extend synonymous expressions of the collocations to words and patterns besides collocations. In addition, we are also interested in extending this method to the extraction of synonymous words so that “black” and “white”, “dog” and “cat” can be classified into different synsets.

Acknowledgements

We thank Jianyun Nie, Dekang Lin, Jianfeng Gao, Changning Huang, and Ashley Chang for their valuable comments on an early draft of this paper.

References

Barzilay R. and McKeown K. (2001). *Extracting Paraphrases from a Parallel Corpus.* In Proc. of ACL/EACL.

Brown P.F., S.A. Della Pietra, V.J. Della Pietra, and R.L. Mercer (1993). *The mathematics of statistical machine translation: Parameter estimation.* Computational Linguistics, 19(2), pp263- 311.

Carolyn J. Crouch and Bokyung Yang (1992). *Experiments in automatic statistical thesaurus construction.* In Proc. of the Fifteenth Annual International ACM SIGIR conference on Research and Development in Information Retrieval, pp77-88.

Dragomir R. Radev, Hong Qi, Zhiping Zheng, Sasha Blair-Goldensohn, Zhu Zhang, Waiguo Fan, and John Prager (2001). *Mining the web for answers to natural language questions. In* ACM CIKM 2001: Tenth International Conference on Information and Knowledge Management, Atlanta, GA.

Fung P. and Mckeown K. (1997). *A Technical Word- and Term- Translation Aid Using Noisy Parallel Corpora across Language Groups.* In: Machine Translation, Vol.1-2 (special issue), pp53-87.

Gasperin C., Gamallo P, Agustini A., Lopes G., and Vera de Lima (2001) *Using Syntactic Contexts for Measuring Word Similarity.* Workshop on Knowledge Acquisition & Categorization, ESSLLI.

Grefenstette G. (1994) *Explorations in Automatic Thesaurus Discovery.* Kluwer Academic Press, Boston.

Kiyota Y., Kurohashi S., and Kido F. (2002) *"Dialog Navigator": A Question Answering System based on Large Text Knowledge Base.* In Proc. of the 19th International Conference on Computational Linguistics, Taiwan.

Koehn. P and Knight K. (2000). *Estimating Word Translation Probabilities from Unrelated Monolingual Corpora using the EM Algorithm.* National Conference on Artificial Intelligence (AAAI 2000)

Langkilde I. and Knight K. (1998). *Generation that Exploits Corpus-based Statistical Knowledge.* In Proc. of the COLING-ACL 1998.

Lin D. (1998) *Automatic Retrieval and Clustering of Similar Words.* In Proc. of the 36th Annual Meeting of the Association for Computational Linguistics.

Shimohata M. and Sumita E.(2002). *Automatic Paraphrasing Based on Parallel Corpus for Normalization.* In Proc. of the Third International Conference on Language Resources and Evaluation.

Wang W., Huang J., Zhou M., and Huang C.N. (2001). *Finding Target Language Correspondence for Lexicalized EBMT System.* In Proc. of the Sixth Natural Language Processing Pacific Rim Symposium.

Zhou M., Ding Y., and Huang C.N. (2001). *Improving Translation Selection with a New Translation Model Trained by Independent Monolingual Corpora.* Computational Linguistics & Chinese Language Processing. Vol. 6 No, pp1-26.