A System to Mine Large-Scale Bilingual Dictionaries from Monolingual Web Pages

Guihong Cao[1], Jianfeng Gao[2], Jian-Yun Nie[1]

1 Département d'Informatique et de Recherche Opérationnelle, Université de Montréal
{ caogui, nie }@iro.umontreal.ca

2 Microsoft Research Redmond, WA
jfgao@microsoft.com

Abstract

This paper describes a system that automatically mines English-Chinese translation pairs from large amount of monolingual Chinese web pages. Our approach is motivated by the observation that many Chinese terms (e.g., named entities that are not stored in a conventional dictionary) are accompanied by their English translations in the Chinese web pages. In our approach, candidate translations are extracted using pre-defined templates. Transliterations and translation pairs are then identified using statistical learning methods. We compare several approaches to aligning transliterations and mining translations on more than 300GB Chinese web pages. In our experiments on MSN query log, we show that the mined bilingual dictionary greatly enlarges the coverage of an existing English-Chinese dictionary. It also improves query translation in cross-language information retrieval, leading to significantly higher retrieval effectiveness in on TREC collections.

1. Introduction

Bilingual dictionaries are valuable for many applications, such as machine translation, cross language information retrieval (CLIR), and information exchange in electronic commerce (Lu et al., 2004; Zhang and Vines, 2004; Cheng et al., 2004). However, traditional bilingual dictionaries are manually crafted, which is very expensive and time-consuming, and cannot be updated timely when new words appear. It is worthwhile to automatically construct live bilingual dictionaries from text collections. This perspective is more attractive than ever as the Web has become a huge data repository from which useful translation pairs can be extracted automatically. Much previous research has devoted to mining bilingual corpora from the Web (Nie et al., 1999; Lu et al., 2004; Zhang and Vines, 2004; Cheng et al., 2004; Huang et al., 2005; Lam et al., 2007). The parallel corpora can then be used to train a generic translation model (or bilingual dictionary).

Nie et al. (1999) tried to automatically discover parallel Web documents written in English and several other languages (such as French and Chinese). They exploit the common organization of parallel texts on the Web (e.g. common parent page, similar file names, etc.) to determine parallel Web pages. Although several parallel corpora have been mined, it turns out that it is difficult to extend the approach to some other languages such as Arabic. More often, we only have monolingual texts on the Web. In addition, the parallel texts cover relatively common terms. It is difficult to extract translations for some special items such as named entities and specialized concepts.

To extend the above approach, Lu et al. (2004) treated any two anchor texts[1] in different languages referring to one object as a translation pair. There are a variety of anchor texts in multiple languages that might link to the same pages from all over the world. This approach can then extract translations for specific terms. It has been reported to achieve high precision but low coverage since anchor texts own only a small percentage in web pages.

Other methods leveraged the quick response of search engines (Zhang and Vines, 2004; Cheng et al., 2004; Huang et al., 2005; Kuo et al., 2006). In these approaches, one side of a translation pair (say English terms) is given, and search engines are used to find the other side of the translation pair (say Chinese terms) from the Web. Though interesting, these approaches are not feasible to build very large bilingual dictionaries.

Different from the previous methods mentioned above, in this paper we try to mine translation pairs (English to Chinese) from monolingual Chinese web pages. Our method is motivated by the observation that many Chinese terms are accompanied by their English translations in the Chinese Web pages. Table 1 shows some segments extracted from Chinese web pages. In each segment, the underlined Chinese words in bold are a translation of the English terms within the parenthesis. This phenomenon is particularly frequent for special terms such as named entities (person or organization names) and specialized concepts. These special terms are usually covered poorly by generic dictionaries or translation models. Therefore, it is valuable to extract translations for them from such segments.

我的磁石(my Magnet)
人类最好的朋友(Man's Best Friend)
我的朋友**维尔克** (Velker)
就到了**财政花园**(TREASURY GARDENS)
而包括**通用电气塑料**(GE Plastics)
又称**特征检测**(Signature-based detection)
入侵检测系统的**特征检测**(Signature-based detection) …
街上全是梦中的店(Tiffani)

Table 1: Translation Pairs in Chinese Web Pages

Based on this observation, we propose an approach to mining bilingual dictionary from monolingual Chinese Web pages. It contains two phases:

Candidate extraction: Translation segments (as those shown in Table 1) are extracted from Chinese Web pages based on a set of predefined templates. However, not all the segments provide correct translation (e.g., the last one in table 1).

Translation selection: The segments are further processed in the second phase to select the correct translation pairs. In previous research, this phase was divided into two independent steps: Chinese phrase boundary detection and translation determination (Lu et al., 2004; Cheng et al., 2004). In this paper, we integrate the two steps in order to minimize global errors. Given a translation pair, we do not fix the boundaries of Chinese

[1] An anchor text is the text shown with a link.

phrase. Rather, all the possible boundaries are taken into account, and a discriminative learning method is used to determine the correct translation pair. All the possible sequence of words or characters just before the English words are considered to be possible translation candidates, which is called a Chinese candidate hereafter Not all the sequences are meaningful, but only those consisting of complete Chinese words segmented using a Chinese word segmentor. For example, for the third segment in table 1 "我的朋友维尔克(Velker)", its Chinese part is segmented as：我/的/朋友/维尔克. We then consider only the following translation pair candidates:

维尔克, Velker

朋友维尔克, Velker

的朋友维尔克, Velker

我的朋友维尔克, Velker

The above constraint has at least two advantages: 1) it reduces the number of possible pairs; 2) it can improve the precision of translation selection. Hereafter, we call each of the above pair an instance (for our learning process). The correct pair is a positive instance; otherwise, a negative instance.

As many of the translation candidates concern named entities, it is important to handle transliteration. In the previous studies, various types of information, such as alignment probability and usage of special Chinese character for transliteration, have been considered. However, the information was combined in a heuristic way (Wan and Verspoor, 1998; Gao et al., 2004). In our work, we formulate the transliteration alignment as a binary classification problem. A more principled discriminative training framework is proposed to combine different types of information in a systemic way.

Our method is more scalable than previous approaches mentioned earlier and can be applied on larger data sets. As a matter of fact, our approach has been applied to a large set of Chinese Web pages, which amounts to more than 300GB. We expect to extract far more translation pairs than those in previous work. Another advantage of our method is that it does not require a list of predefined English terms, which is a prerequisite in approaches leveraging search engines (Zhang and Vines, 2004).

Two experiments are conducted to evaluate the quality of the mined bilingual dictionary. The first one investigates the coverage of the dictionary with respect to the English terms encountered in Web search query logs. The other is to translate queries for cross-lingual information retrieval. Both experiments show that the mined dictionary provides additional and useful translations beyond traditional bilingual dictionaries. To our knowledge, it is the first attempt to mining bilingual dictionary from such a large quantity of Web pages.

The remaining of the paper is organized as follows: in section 2, we will describe the architecture of the system. We give a brief introduction to the three major modules: data pre-processing, transliteration alignment and translation selection. The next three sections will provide the details of the three modules respectively. Section 3 presents the algorithms for pre-processing; section 4 describes a binary classifier based on averaged perceptron (Collins, 2002) to determine transliterations. The translation selection module is described in section 5. Section 6 presents the experiments using the mined dictionary. Section 7 concludes the paper.

2. Architecture of the Dictionary Mining System

The system of mining bilingual dictionary consists of three components: pre-processing, transliteration alignment, and translation selection.

The ***pre-processing module*** bridges the monolingual Chinese web pages and the translation selection module. The pre-processing module filters the HTML tags, normalizes character coding,, extracts translation segments based on some templates, and segments the Chinese string in each segment.

Many translation segments are a mixture of translation words and transliterations. As a consequence, we have to address the transliteration issues, which is the task of the ***transliteration alignment***. Given an English-Chinese pair, i.e., a translation segment, the module determines whether the pair or part of the pair is a transliteration. This module will be described in detail in Section 4, for which we use an averaged perceptron classifier (Collins, 2002) based on a set of features.

The third module is ***translation selection***. It also uses the averaged perceptron algorithm. However, different from the transliteration module, we do not use a binary classifier, but a ranker. As mentioned in section 1, each translation segment shown in table 1 can generate a set of instances. The ranker can cope with this problem. It is trained to maximize the score of the positive instances in training data. We will give the details of this module in section 5.

3. Data Pre-processing

The data pre-processing module bridges the raw Chinese web pages and the translation selection module. This module has two main functions: extracting translation segments, which are shown in table 1 and segmenting the Chinese string in each segment.

Previous studies also employed a similar module to extract translation segments from web pages. Zhang and Vines (2004) extracted all English terms surrounding a Chinese phrase, and considered each English term as a candidate translation to the Chinese phrase. However, this method cannot be used in our case because the method would generate many such segments from the Chinese web pages and most of them do not have translation pairs. In order to extract better translation segments, we observed that most translation pairs occurring in the Chinese web pages follow some templates. We defined four templates to extract the segments:

1). $c_1c_2..c_n$ (En)

2). $c_1c_2..c_n$, En, $c'_1c'_2..c'_m$

3). $c_1c_2..c_n$: *En*

4). $c_1c_2..c_n$ *是/即 (is/are) En*

where $c_1c_2..c_n$ refers to a Chinese sentence, and *En* refers to an English string.

Template	*Percentage*	*Precision*
1	17.65%	54%
2	68.35%	6.5%
3	9.05%	2.5%
4	4.94%	1%

Table 2: Comparing Precision of the Four Templates

The four templates lead to different precision of the extracted translation segments. It is interesting to investigate the quality of the templates. We randomly sampled 600 segments from the extracted segments and checked them one by one manually. If the Chinese string contains the translation of the English phrase, i.e., the past part of $c_1c_2..c_n$ is the translation of *En*, we consider the segment is correct, otherwise, it is incorrect. Table 2 shows the results.

From the above table, we see that only template 1 achieves a satisfactory precision. The other three templates, especially template 4, have a very low precision. For the sake of efficiency, in this study we only use the template 1 and omit the other three templates. We leave the study of other templates to future work.

After extracting the translation segments, we used MSRSeg, a Chinese segmentor and NE recognizer developed at Microsoft Research (Gao et al., 2005), to segment the Chinese string in each segment.

4. Transliteration Alignment

Transliteration addresses a special case of translation, in which the source phrase is translated to target phrase not based on semantic clues, but phonetic clues. Transliteration is very important for mining bilingual dictionary. In general, it is difficult for human translators to translate unfamiliar named entities, such as person names, locations and organization names. Therefore, these named entities are usually transliterated. To make the description clearer, we will use "translation" to refer to word translations by meaning other than transliterations.

4.1 Separating Transliteration Units and Translation Units

In many cases, named entity translations use both translation and transliteration. For example, in the pairs (Little Smoky River, 小斯莫基河) and (Carnegie Mellon University, 卡内基梅隆大学), "little", "river", and "University" are translated as "小", "河", and "大学" respectively; and "Smoky", "Carnegie", and "Mellon" are transliterated into "斯莫基", "卡内基" and "梅隆". In order to deal with translation and transliteration separately, we need a mechanism to separate the transliteration units from the translation units in a phrase, i.e., to determine whether a word in a phrase is a translation or transliteration.

In Chen et al. (2006), a frequency-based approach was used to perform this separation. They assumed that most transliterations were named entities, meaning that they occurred less frequently in the text. Therefore, the words with frequency larger than a threshold were considered as translations; otherwise they were transliteration. However, some transliterations, such as "维多利亚" (Victoria) and "哈利波特" (Harry Potter), are highly frequent. This approach will fail in these cases.

In our work, we did not take the frequency of terms into account. We employed a rule-based approach to perform this separation. We found that most proper nouns in Chinese are formed with the structure: (translation unit) + transliteration unit + (named entity suffix). The parts in parentheses are optional. Here the *named entity suffix* represents some common words in Chinese that indicate a person name, location name or organization name, for example, "先生" (Mr.), "公司" (Ltd.), "河" (river), "大学" (university), "教授" (professor), etc. We used a list containing 40 commons named entity suffixes (shown in Appendix A). We also used a small Chinese to English bilingual dictionary. If a Chinese word exists in the dictionary, we assume it a translation unit. Therefore, given a Chinese sentence, we remove the suffix and translation units (if they exist) and the remaining part is the transliteration units.

After the transliteration units in the Chinese phrase are identified, the corresponding transliteration units in the English phrase can be detected via transliteration alignment. If the units in the English phrase can be aligned to any Chinese transliteration units, they are English transliteration units; otherwise, they are translation units.

4.2 Alignment Model for Transliteration

We assume that each Chinese transliteration unit consists of a Pinyin sequence and each English transliteration unit consists of a phoneme sequence. Then we have

$Ch = c_1c_2c_3 \dots c_n$ and $En = e_1e_2e_3 \dots e_m$

where *Ch* and *En* are a Chinese and English transliteration unit respectively. Different from the traditional word alignment model (Brown et al., 1993), the transliteration alignment is strictly monotonic. We denote an alignment as $A = a_1a_2 \dots a_l$, where a_i is an alignment unit which consists of a set of Chinese Pinyins and English phonemes. We denote them as Ch_i and En_i. Suppose that Ch_i contains n_i Pinyins and En_i contains m_i phonemes. The probability to align the Ch to En is:

$$P(Ch|En) = \sum_A P(Ch, A|En)$$
$$= \sum_A \prod_i P(Ch_i|En_i)P(n_i|m_i) \quad (1)$$

Using the so-called maximum approximation, we have:

$$P(Ch|En) \approx P(Ch, A^*|En) \text{ where}$$
$$A^* = argmax_A P(Ch, A|En)$$
$$= argmax_A \prod_i P(Ch_i|En_i)P(n_i|m_i) \quad (2)$$

where $P(Ch_i|En_i)$ represents the transliteration probability between the two transliteration units and it is estimated using maximum likelihood estimation (MLE) and $P(n_i|m_i)$ represents the length alignment probability. In a simple model (i.e., Basic Model), we consider $P(n_i|m_i)$ as a uniform distribution and then can be omitted in argmax. In a more sophisticated model (i.e., Length Model), we estimated the length probability with MLE. Based on equation 2, the alignment probability $P(Ch|En)$ can be calculated efficiently with dynamic programming. We used an EM algorithm to train the parameters in the transliteration model. For the sake of simplicity, we call one Pinyin or phoneme a phonetic unit. To reduce the search space, we defined four alignment templates (Gao et al., 2004). They are

(1). 1 phonetic unit to 1 phonetic unit;
(2). 1 phonetic unit to 2 phonetic units / 2 phonetic units to 1 phonetic units
3). 0 phonetic unit to 1 phonetic unit / 1 phonetic unit to 0
4). 2 phonetic units to 2 phonetic units.

Therefore, a possible alignment unit may have 0, 1 or 2 phonetic units. Figure 1 shows the algorithm to train the Basic Model.

(1) Initialization: considering each possible alignment unit in the source language is aligned to all the alignment units in the target language whose distance to the source unit is less than 2 units. Then calculate the transliteration probability $\boldsymbol{P}(\boldsymbol{Ch_i}|\boldsymbol{En_i})$ between possible alignment units.
(2) E-step: Based on the current model, obtaining the best alignment according to equation 2
(3) M-step: Re-estimate the transliteration probabilities between possible alignment units among the best alignment.
(4) If the stopping criteria are reached, stop training; otherwise, go to step 2.

Figure 1: EM Algorithm to Train Basic Model

The training of the Length Model is similar to that of Basic Model. The only difference is that we estimate the length alignment probability $P(n_i|m_i)$, in addition to the transliteration probability, and we consider the length probability when searching for the best alignment.

4.3 Converting English Word into Phonemes

We can simply map a Chinese word to a Pinyin sequence based on a Chinese to Pinyin table. But the case of mapping English words to phonemes is more complicated. Previous work used a pronunciation dictionary (CMUdict, 1995). However, this approach is not applicable in our case because more than 70% named entities mined from the web pages are not covered by the dictionary.

However, we notice that we can approximate the corresponding phonemes of an English word by its syllables. In the system, we used a rule-based method to convert each English word into a sequence of syllables, and each syllable is further decomposed into several sub-syllables (Wan and Verspoor, 1998). Each sub-syllable contains only one phoneme. Therefore, this process can be viewed as transforming an English word into phoneme sequence.

4.4 Perceptron-based Binary Classifier

One straightforward measure to determine whether an English-Chinese pair is a transliteration is the alignment score calculated using equation 2. However, we found that this is insufficient. Consider the pair "Smoky" and "斯莫基". The transliteration alignment probability is fairly large, and they are transliteration for each other. However, the score for mapping "Smoky" to "是斯莫基的" is also very large using the English-to-Chinese transliteration model, but they are not transliteration. Fortunately, we found that the probability of the Chinese-to-English mapping in such cases is pretty low. Therefore, by combining these two directional probabilities, we are able to determine that "是斯莫基的" is not a transliteration of "Smoky".

As we know, some Chinese words, such as "斯", "列", "基", have a large probability to be a transliteration, while other words, such as "的", "走", "跑", rarely occur as a transliteration. In previous work (Chen et al., 2006), this information is integrated via rules for transliteration disambiguation. The coverage of such rules is not satisfactory. We integrate this information in a more principled way. We trained a character bigram language model on a dataset of Chinese transliterations. The correct transliteration is expected to have a high language model probability, while incorrect one with low probability. Adopting the character-based language model for transliteration selection is not new. Kuo et al. (2006) have done it. However, we combine other information besides the language model with a discriminative framework.

We integrate all the information mentioned above under a discriminative learning framework. In our work, we use a binary classifier trained by the averaged perceptron algorithm due to its simplicity and efficiency (Collins, 2002; Gao et al., 2005). The following features are used:

1) The logarithm of the transliteration probability from English to Chinese normalized by the number of alignment units. The probability is normalized because it decreases monotonically with the number of alignment units.
2). The normalized logarithm of the transliteration probability from English to Chinese.
3). The ratio of the number of English alignments and Chinese alignments in the English to Chinese transliteration.
4). The ration of the number of Chinese alignments and English alignments in the Chinese to English transliteration.
5). The perplexity of the Chinese string with respect to the Chinese transliteration language model. The feature is to test whether the Chinese contains unusual Characters used for transliteration.

4.5 Experimental Results

Model	*Accuracy*
BM	97.61%
LM	94.26%

Table 3: Results of Transliteration Identification

To evaluate the algorithm described above, we created a data set consisting of 36,012 English-Chinese pairs. Each pair may or may not be a transliteration. We separated the data set into three parts: training data, development data and test data. The training data contains 32,236 transliterations, which is used to train the English to Chinese and Chinese to English transliteration alignment models described in section 4.2. The development data has 2,517 pairs, in which there are 1,632 transliterations and 895 non-transliterations. The development data is used to train a binary classifier. The test data has 1,259 pairs, with 821 transliterations and 438 non-transliterations. We performed two experiments, one with Basic Model, and the other with Length Model. The

language model for Chinese transliteration is a character-based bigram model. Table 3 shows the results.

In this table, BM refers to the model in which the transliteration probabilities for both English to Chinese and Chinese to English are calculated with the basic transliteration model; while the LM refers to the one leveraging the Length Model. We found that both model produce good result. It is difficult to compare our results with other transliteration systems since other systems have two models, i.e., alignment model and generation model, while ours only has alignment model. Even though, the accuracy of the start-of-the-art transliteration systems is less than 50% (Gao et al., 2004), which is far from satisfactory. It is interesting to add a generation model to our transliteration system and compare ours with other systems. We leave it as one future work.

The BM model outperforms the LM. One possible reason is that the probability $P(n_i|m_i)$ in the Length Model is poorly estimated. In the experiments, we found that 1-1 alignment dominate the length probability (more than 90%) so that the length probability does not provide any additional information.

5. Translation Selection

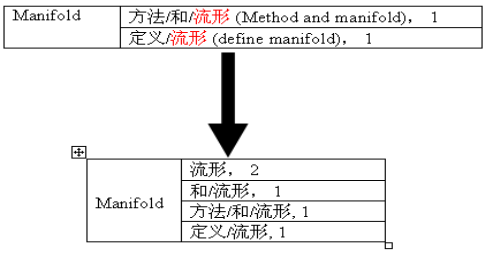

Figure 2: Process to Extract Instances from Translation Pair Candidates

The translation selection module is the core component of the mining system. Given segmented translation pair candidates, the module selects the correct ones according to some criteria. The mining process is divided into two phases: transliteration detection and translation pair scoring. When given a translation pair candidate, the system first uses the transliteration alignment module to determine whether it is a transliteration. If it is, the system adds it to the mined bilingual dictionary; otherwise, the system passes it to the translation selection module. We view the selection problem as a ranking task (Joachims, 2002), and choose the one with the largest score and discard all others. Because an English phrase may not have a translation, such as the last segment in table 1. We set a threshold for selection. If the score of the chosen one is greater than the threshold, we consider it as correct translation pair. This process can be illustrated with figure 2. There are two translation pair candidates shown in this figure, and they share the same English phrase. In the pre-processing phase, the Chinese parts have been segmented. Since the Chinese suffixes in the two candidates may contain a correct translation of "manifold", we generate four instances as shown in the bottom of the figure. Each instance consists of three fields: the English phrase, the Chinese candidates, and the co-occurrence frequency of them. We then apply the ranker to compute a score for each instance, and select the one with the largest score. If the score is greater than a threshold, we consider it to be the correct one. In next section, we will describe the ranking algorithm in detail.

5.1 Translation Selection with Multiple Nested Ranker

One approach to selecting translation pairs is, as what we do for transliteration, to train a binary classifier by assigning correct instances to a positive class and incorrect instances to a negative class. However, this approach does not work reasonably well in our case because there are much more negative instances in the data. As the example shown in figure 2, only one instance, "流形" is correct, and all the other three are incorrect. If the Chinese string is longer, there will be more negative instances with at most one positive instance. If a classifier were trained on such an unbalanced data set, it would classify most instances, including positive instances, as negative instances (Zhang and Mani, 2003).

As a consequence, we use an alternative technique based on a ranker, which is also based on the averaged perceptron algorithm. The ranker is trained to assign a higher score to positive instances and a lower score to negative instances. When mining translation pairs, we score all instances, which share one English phrase, and select the one with the highest score as the positive instance, i.e., the correct translation pair. Suppose there are *N* different English phrases in the training data, each can generate a set of instances, we denote the set of instances as S_i, and the instances in S_i as c_{ij}. The features of instance c_{ij} is denoted as $f_k(c_{ij})$, and the score of c_{ij} is $S(c_{ij})$. With the perceptron ranker, we have:

$$S(c_{ij}) = \sum_{k=0}^{M} \lambda_k f_k(c_{ij}) \quad (3)$$

where $\lambda_k, k = 1,2,\ldots,M$ represents the weights of M features and λ_0 is the bias. The ranker can be trained using the algorithm illustrated in figure 3.

Input: training samples, $S_i, i = 1,2,3,\ldots,N$

Output: parameter setting λ_k

1. **Initialization:** set $\lambda_0 = 1, \lambda_k = 0, k = 1,2,3,\ldots,M$
2. **For** $t = 1$ **to** T
   3. **For each** training sample S_i
      4. For instances c_{ij} and c_{il} in S_i
         If $S(c_{ij}) > S(c_{il})$ and c_{ij} is negative while c_{il} is positive, **then**
            5. $\lambda_k^{t+1} = \lambda_k^t + \eta(f_k(c_{il}) - f_k(c_{ij}))$
               $k = 0,1,2,3,\ldots,M$
6. $\lambda_k = \frac{\sum_{t=1}^{T} \lambda_k^t}{T}$

Figure 3: Training Averaged Perceptron Ranker

To deal with the issue of the unbalance data further, we use the multiple nested ranker (Matveeva et al., 2006).

The idea behind the nested ranker is intuitive: At each step of the training process, instead of using all the negative instances, we only select a subset of them, in which the percentage of positive instances is larger than that of the whole data set. The selected subset is supposed to be more balanced than the original data set. These instances are then used to train a new perceptron ranker,

which is in turn used to select a new subset of instances for the training of a new ranker at next step. Intuitively, the new ranker should perform better than the earlier ones. In our experiments, we keep top 50% pairs with higher scores at each step. We show the pseudo-code for training nested ranker in figure 4.

Input: training samples, $S_i, i = 1,2,3, \dots, N$

Output: parameter setting for R rankers $\lambda_k^r, r = 1,2, \dots, R$

For r=1 to R

  For each training sample S_i

    Select 50% c_{ij} with the highest score in S_i

  Train Averaged Perceptron Ranker, and output parameter λ_k^r

Figure 4: Training Nested Ranker

In the mining phase, the system also keeps top 50% pairs and determines all the rest pairs as incorrect one. If the number of remaining pairs is less than 1, the system does not do any cutting. At the final step, the system outputs the one with the highest score and the score is greater than a threshold as a positive instance.

As mentioned above, the ranker relies on a set of feature to score instances. Below are the features we used.

1) the relative frequency of the Chinese candidate (Zhang and Vines, 2005)
2) length ratio of the English phrase and the Chinese candidate
3) length of the Chinese candidate (Zhang and Vines, 2005)
4) whether translation of the first English word after reordering is in the candidate
5) number of unique terms before the candidate
6) whether the word immediately before the suffix is a indicator word, such as "的", "和", "之", "与" …

The features are intuitive except for the No.4, which deserves a further explanation. In fact, feature 4 deals with one common problem in translation between Chinese and English. In many cases, an English noun phrase of the form "noun1 prep noun2" is translated to Chinese as "noun2 noun1" sequence. This phenomenon is very common. For example, "University of Victoria" is translated into "维多利亚大学", in which "University" is translated into "大学" and "Victoria" is translated into "维多利亚". In order to cope with this problem, we defined a simple template for reordering: A of/in/at B → BA. Here A and B are English words, and they are connected with a preposition word, such as "of", "in" or "at". When being translated to Chinese, they are reordered as "BA". If the translation of B occurs in a Chinese candidate, the candidate thus is more likely to be a translation of the phrase. For example, "University of Victoria" is reordered to "Victoria University", and the translation of "Victoria" occurs in a Chinese candidate, i.e., "维多利亚大学", which is the translation of "University of Victoria".

Translation %	Transliteration %	Accuracy %
53.55	46.45	90.15

Table 4: Accuracy of Mined Dictionary

5.2 Experimental Results

We used the system processed more than 300GB Chinese web pages. We obtained 834,329 translation pair candidates. 161,117 translation pairs are mined from the candidates. Table 4 shows the accuracy and some statistical information of the mined dictionary. To obtain the table, we randomly choose 402 pairs from the 161,117 mined pairs and check them one by one manually. The table show that the accuracy of the dictionary is quite high (90.15%). Therefore, the mined dictionary includes about 145,246 correct translation pairs. In the mined pairs, translation pair almost has the same percentage with the transliteration pairs.

We compared the 161,117 translation pairs with LDC2.0 English to Chinese bilingual dictionary, which is publicly available and is used by many cross lingual information retrieval experiments (Gao and Nie, 2006), and the results are shown in table 5.

	LDC2.0	**Mined Dict**
#pair	110,834	161,117
#En	109,745	127,145
#overlapped En	9280	9280
#pair: number of translation pairs #En: number of unique English terms #overlapped En: number of overlapped English terms between the two dictionaries		

Table 5: Comparing the Mined Dictionary with the LDC Dictionary

From the table, we observe that the dictionary mined by our system contains more translation pairs than the LDC dictionary, and there is a small overlap between the two dictionaries. It is reasonable that the two dictionaries are very different in nature. The LDC dictionary contains common words while the mined dictionary contains hot terms (including a lot of proper nouns) in the Web. In a sense, these two dictionaries complement each other pretty well. Therefore, it is reasonable to expect a significant benefit from a combination of them. In what follows, we will verify our speculation experimentally.

6. Evaluating the Mined Dictionary

In this section, we conducted two experiments to evaluate the quality of the mined bilingual dictionary. One is performed on web query logs and the other is performed on the cross lingual information retrieval task.

6.1 Coverage of Query Logs

haoshifu	realplayer	muice
bourges	boydell	canadian
welcome	photoshop	sql
bowmore	guipian	luxun
wajaa	baidu	yaoming
hongda	wangwei	mmvod
coogle	bud	powerdvd
tudou	xibu	spears

Table 6: English Query Terms Extracted from Chinese Search Engine Query Logs

We collected 80,885 popular query terms from MSN Chinese search engine. From them, we selected 9,065 English terms with highest frequency. We assume that the

Coll	*Description*	*Size (MB)*	*#Doc*	#Qry
TREC5&6	People's Daily (1991-1993)&Xinhua News Agencey (1994-1995)	162	164,789	54
TREC9	HongKong Commercial Daily News, HongKong Daily News and Takungpao News	260	127,938	25

Table 8: Statistical Information of Dataset

users used these English terms to retrieve relevant Chinese documents, which is the scenario of cross-lingual information retrieval on the Web. Table 6 shows some of the query terms.

From the table, we find that some terms are Chinese person names (wangwei, yaoming), some are names of products (powerdvd) or software (photoshop, realplayer), and some terms are meaningless probably due to spelling errors (coogle). At the first glance, it seems very difficult to translate these terms using any pre-complied bilingual dictionary. We compared the coverage of the mined dictionary with LDC dictionary, and the result is shown in table 7.

Dictionary	**Coverage**	**Improvement**
LDC	0.2030	
MinedDict	0.3457	+70.29%
MinedDict+LDC	0.3688	+81.67%

Table 7: Compare the Coverage of Two Dictionaries

We find that the coverage of mined dictionary is much larger than the LDC dictionary. This is expected because the mined dictionary contains many hot terms in the web which rarely occur in common dictionaries. Taking the high accuracy (90.15%) of the mined dictionary into account, the experiment shows that in the Web context it makes more sense to use the mined dictionary than the manually crafted dictionary.

6.2 Evaluation on Cross Lingual Information Retrieval

We used two benchmark English to Chinese CLIR collections: TREC5&6, TREC9 in our experiments. Table 8 shows the statistical information of these collections.

All Chinese documents and the translated queries are segmented using dictionary-based approach. The dictionary was compiled by UC Berkley, which contains 137,613 entries. When indexing the document collections, we used all possible words in the dictionary and all single Chinese characters as indexing units. All English queries are stemmed with Porter stemmer and the stop words are removed. Since we do not have a phrase recognizer, the phrases in the English queries are simply detected in a vocabulary-based manner. To build the phrase vocabulary, we take all consecutive English word sequences occurring in the bilingual dictionary to be phrases. Each English query in TREC collection has three fields: title, description and narrative. We used two versions of queries: short queries that contain only titles and long queries that contain all the three fields.

We trained a statistical translation model for query translation. The translation probabilities in this model are obtained using the GIZA++ toolkit (Xu et al., 2001). GIZA++ considers the bilingual dictionary as a parallel corpus and learned a statistical translation model. GIZA++ implements several translation models, we used IBM model 1 (Brown et a., 1993) for its simplicity and efficiency. In our experiments, this approach has proven to be more effective than simple approaches such as selecting the first translation candidate or select all the candidates. From the translation model, we selected the top 10 translations for each term for the short queries and top 3 for long queries. Tables 9 and 10 show the results.

In the two tables, ML refers to the monolingual retrieval, and is usually considered as the upper bound of the cross lingual IR performance. LDC refers to the CLIR run using LDC dictionary only and LDC+Mined uses LDC dictionary together with the mined dictionary. #UNK refers to the number of unknown words in the query terms and #total is the total number of query terms. In fact, the usefulness of the mined dictionary could be underestimated when we evaluated it using the TREC data because TREC queries are different from real web queries. For example, many web queries contain more named entities, which also amount a large portion in the mined dictionary. The TREC queries however consist of common words. Moreover, TREC queries were produced several years ago and relating to out-of-date topics, but the mined dictionary contains recent popular words. Even though, from the two tables, we found that the mined dictionary can reduce the number of unknown words and improve the retrieval effectiveness. When the mined dictionary is combined with the LDC dictionary, the retrieval effectiveness is significantly improved. In the case of short queries, we see that the reduction of number of unknown words is not large; nevertheless, the addition of the mined dictionary has been very useful. This can be

	TREC5&6			**TREC9**		
	MAP	% of ML	#UNK/#total	MAP	% of ML	#UNK/#total
ML	0.3754			0.2458		
LDC	0.2839	75.63	12/300	0.2020	82.18	2/95
LDC+Mined	0.2963	78.93	8/300	0.2367	96.30	1/95

Table 9: CLIR Results of Short Queries

	TREC5&6			**TREC9**		
	MAP	% of ML	#UNK/#total	MAP	% of ML	#UNK/#total
ML	0.4929			0.2814		
LDC	0.3810	77.3	286/2414	0.2235	79.42	85/721
LDC+Mined	0.4092	83.02	169/2414	0.2266	80.53	78/721

Table 10: CLIR Results of Long Queries

explained by the fact that even for a word covered by the dictionary the mined dictionary can propose new appropriate translations. These latter can be more reasonable translation candidates.

7. Conclusion

In this paper, we describe a system to mine English-to-Chinese bilingual translations/transliterations from monolingual Chinese Web pages. The system consists of three main modules: data pre-processing, transliteration alignment and translation selection.

The transliteration module can be treated as a stand-alone system, which can be used to determine whether an English-Chinese pair is a transliteration. We also conducted some experiments to evaluate the system and the result is encouraging.

We ran the system over more than 300GB Chinese web pages. From the web pages, 834,329 translation pair candidates were extracted. We mined a bilingual dictionary containing 161,117 translation pairs from the candidates. We compared the mined dictionary with LDC2.0 dictionary and found very small overlap between them.

We evaluated the mined dictionary in two real world applications. One aims to test the coverage of the mined dictionary over English queries extracted from the Chinese query log data and the other aims to test the effectiveness of the mined dictionary in the CLIR tasks on the TREC benchmark data. In the query log experiments, the mined dictionary can improve the coverage of query terms up to 70%. Our experiments on CLIR showed that the mined dictionary is complementary to a manually constructed bilingual dictionary. When both dictionaries are used, we observed large increase in retrieval effectiveness. To our knowledge, this work is the first attempt to mining a bilingual dictionary from monolingual Web pages in a large scale.

Bibliographical References

Brown, P., Della Pietra, S., Della Pietra, V., and Mercer, R. (1993). The Mathematics of Statistical Machine Translation. Computational Linguistics, 19(2)

Chen, H.H., Lin, W.C. and Yang, C.h. (2006). Translation-Transliterating Named Entities for Multilingual Information Access. Journal of the American Society for Information Science and Technology, 57(5):645-659

Cheng, P., Teng, J., Chen, R., Wang, J., Lu, W., and Chien, L. (2004). Translating Unknown Queries with Web Corpora for Cross-Language Information Retrieval. In the Proceedings of SIGIR2004, pp.162-169.

CMUdict, 1995. The CMU Pronouncing Dictionary 0.6. ftp://ftp.cs.cmu.edu/project/speech/dict.

Gao, J.F. and Nie, J.-Y. 2006. Study of Statistical Models for Query Translation: Finding a Good Unit of Translation. In the Proceedings of SIGIR 2006.

Gao, J.F., Li, M., Wu, A., and Huang, C.N. 2005. Chinese word segmentation and named entity recognition: a pragmatic approach. Computational Linguistics, 31(4)

Gao, J.F., Qin, H., Xiao, X., and Nie, J.Y. (2005). Linear Discriminative Model for Information Retrieval. In the Proceedings of SIGIR2005.

Gao, W., Wong, K.F. and Lam, W. (2004). "Phoneme-based Transliteration of Foreign Names in Cross Language Information Retrieval", IJCNLP2004, pp374-381.

Huang, F., Zhang, Y., and Vogel, S. (2005). Mining Key Phrase Translations from Web Corpora. In the Proceedings of HLT-EMNLP2005.

Joachims, T. (2002). Optimizing Search Engines Using Click through Data, Proc. of the 8th ACM Conference on Knowledge Discovery and Data Mining.

Lam, W., Chan, S.K., and Huang, R. (2007). Named Entity Translation Matching and Learning: With Application for Mining Unseen Translations. ACM Transactions on Information Systems, 25(1)

Lu, W. and Lee, H. (2004). Anchor Text Mining for Translation of Web Queries: A Transitive Translation Approach. ACM Transactions on Information Systems, Vol.22, April 2004, pages 242-269.

Matveeva, I., Burges, C., and Burkard, T. 2006. High Accuracy Retrieval with Multiple Nested Ranker. In the Proceedings of SIGIR2006.

Michael Collins. 2002. Discriminative training methods for hidden Markov models: Theory and experiments with perceptron algorithms. In Proc. EMNLP, pages 1–8.

Nie, J., Simard, M. Isabelle, P. and Durand, R. (1999). Cross-language Information Retrieval Based on Parallel Texts and Automatic Mining of Parallel Texts from the Web. In the Proceedings of SIGIR1999, pp. 74-81.

Wan, Stephen and Verspoor, Cornelia Maria, 1998. Automatic English-Chinese name transliteration for development of multilingual resources. In the Proceedings of the COLING/ACL.

Xu, J.X., Weischedel, R., and Nguyen, C. (2001). Evaluating a Probabilistic Model for Cross-lingual Information Retrieval. In Proceedings of SIGIR2001

Zhang, J.P., and Mani, I. (2003). kNN Approach to Unbalanced Data Distributions: A Case Study involving Information Extract. In the Proceedings of Workshop Learning from Imbalanced Dataset, ICML, Washington DC.

Zhang, Y. and Vines, P. (2004). Using the Web for Automated Translation Extraction in Cross-Language Information Retrieval. In the Proceedings of SIGIR2004, pp.162-169.

Kuo, J.S., Li, H., and Yang Y.K (2006). Learning Transliteration Lexicon from the Web. In the Proceedings of COLING/ACL2006, pp1129-1136

Appendix A: Common Chinese Suffixes

县(County), 路(Road), 区(District),弄(street), 里(streat), 寨(village), 村(village), 乡 (village), 道(road), 郡 (district), 洞 (district), 门(gate), 桥(Bridge), 塔 (Tower), 园(Park), 市(City), 城(City), 省(Province), 山 (Mountain), 岭(Mountain), 峰, 河(River), 江(River), 海 (Sea), 溪(creek), 楼(Building), 湖(Lake), 水(Water), 潭 (lake), 沟(creek), 渠(arcduc), 站(Station), 组件 (Component),国际(International), 公司(Company, Ltd.), 大学 (University), 协会(Association), 先生 (Mr.), 小姐 (Miss), 女士(MS)