# Comparing Different Units for Query Translation in Chinese Cross-Language Information Retrieval

Lixin Shi, Jian-Yun Nie, Jing Bai
Département d'informatique et de recherche opérationnelle, Université de Montréal
C.P. 6128, succursale Centre-ville, Montréal, Québec, H3C 3J7 Canada
{shilixin, nie, baijing}@iro.umontreal.ca

## ABSTRACT

Although both words and n-grams of characters have been used in Chinese IR, they have often been used as two competing methods. For cross-language IR with Chinese, word translation has been used in all previous studies. In this paper, we re-examine the use of n-grams and words for monolingual Chinese IR. We show that both types of indexing unit can be combined within the language modeling framework to produce higher retrieval effectiveness. For CLIR with Chinese, we investigate the possibility of using bigrams and unigrams as translation units. Several translation models from English words to Chinese unigrams, bigrams and words are created based on a parallel corpus. An English query is then translated in several ways, each producing a ranking score. The final ranking score combines all these types of translation. Our experiments on several collections show that Chinese character n-grams are reasonable alternative translation units to words, and they lead to retrieval effectiveness comparable to words. In addition, combinations of both words and n-grams produce higher effectiveness.

## Categories and Subject Descriptors

H.3.3 [Information Search and Retrieval]: Retrieval Models, Search Process

## General Terms

Algorithms, Experimentation, Performance

## Keywords

CLIR, Language Model, Parallel Corpus, Translation Model, Translation Unit

## 1. INTRODUCTION

Cross-language information retrieval (CLIR) is becoming increasingly important due to the rapid development of the Web. As the query and the documents are written in different languages, the main problem of CLIR is the automatic translation between query and document languages. The basic approach is to translate the query from a source language to a target language. There are three main techniques for query translation: using a machine translation (MT) system, using a bilingual dictionary, and using a statistical model trained on parallel texts. It has been shown that when used correctly, these approaches can lead to comparable retrieval effectiveness [10, 11, 13, 14, 19]. However, for CLIR involving Chinese, words are usually used as the translation units. Although n-grams of characters have been found to be reasonable alternatives to words in indexing [17, 18], no previous study has investigated the possibility of using Chinese character n-grams as translation units. In this study, we will investigate into this issue. Our investigation will make use of a parallel corpus.

The problem of indexing and translation units in Chinese stems from the fact that word boundaries are not explicitly marked in Chinese sentences. While in most European languages, one can be content with using words as units for both indexing and translation, in Chinese, we have to determine the units by an additional process – either using word segmentation or by cutting the sentence into n-grams (usually unigrams and bigrams). However, this process is not trivial due to ambiguities and unknown words: A Chinese sentence can often be segmented into several different sequences of words, and documents and queries can often contain unknown words (e.g. person's names, new words, etc.). These problems have an important impact on IR. For example, if a term is segmented differently in a document and in a query, then no match will be made between them based on this term.

One may argue that by using the same segmentation process, the same sequence of Chinese characters will likely be segmented in the same way, producing the same words. Therefore, from a linguistic point of view, the danger of producing different word segmentations is largely reduced. However, for two slightly different sequences of words, the danger of producing incomparable words still exists. For example, for the sequence 发展中国家 (developing country), it is well possible that it is segmented inconsistently into 发展 (development) 中 (middle) 国家 (country) or 发展 (development) 中国 (China) 家 (family), depending on the segmentation method used and the context.

In addition, for information retrieval (IR) purposes, it is not sufficient to perform word segmentation in a consistent way. We also encounter the problem of semantic similarity: two different words do not always have different meanings. They can be related, especially when the words share some common characters such as 办公室 (office) and 办公楼 (office building). If these two words are considered to be different indexes, then it is impossible to compare a document containing one word to a query containing another word. This problem of word similarity is widely spread in Chinese. To deal with it, another common approach to Chinese IR is to use characters or bigrams of characters as indexing units. Then the above two words will share a common characters or bigram. It has been shown that using words or bigrams of Chinese



*Conference Infoscale 2007,* June 6-8, 2007, Suzhou, China

characters as indexing units results in equivalent effectiveness, and combining them leads to better effectiveness [17, 18, 20].

However, for CLIR with Chinese, only words have been used as translation units. No study has investigated the possibility of using n-grams of Chinese characters as translation units or their combination with words. The main focus in this paper is to investigate the impact of using different Chinese units in CLIR. We will first re-examine the utilization of words and character n-grams in Chinese IR. Then we compare different approaches to query translation using different translation units. Our experiments on several large (NTCIR and TREC) test collections will show that in both Chinese monolingual and cross-language IR, it is always better to combine words and n-grams.

The remaining of the paper is organized as follows. In Section 2, we will describe the background of our study. Some related work will be described. Section 3 will describe our approaches using different translation models (TM) for Chinese CLIR. Section 4 describes the experimental setting and results. Conclusions and future work will be given in Section 5.

# 2. BACKGROUND

## 2.1 Chinese Monolingual IR

Chinese monolingual IR has been studied for more than one decade. The difference from IR in English and in Chinese lies in the fact that word boundaries are not marked in Chinese. In order to index a Chinese text, the latter has to be cut into indexing units. The simplest method is to use single characters (unigrams) or all adjacent overlapping character pairs (bigrams), such as in [7] [12]. Another method is to segment Chinese sentences into words, as in [15]. Several studies have compared the effectiveness of these two types of indexing unit in Chinese IR [17, 18, 20]. They all show that words and bigrams can achieve comparable performances. Both indexing methods have produced higher retrieval effectiveness than unigrams.

The previous studies have been carried out using different retrieval models: vector space model, probabilistic model, etc. No comparison has been made using language modeling (LM). In this study, we will re-examine the problem of indexing units for Chinese IR within the LM framework. Our conclusion will be slightly different from previous ones: our experiments will show that unigrams are more effective indexing units than words and bigrams alone.

## 2.2 Using Parallel Corpus for CLIR

Parallel texts are texts in one language accompanied by their translations in another language. Parallel corpora containing such texts have been used for CLIR in different manners.

A simple method is used in [8] [26]: a source language query is first used to retrieve source language documents in the parallel corpus; then the parallel texts in target language corresponding to the top retrieval results are used to extract some target language words; these latter are considered as a "translation" of the query. This method works in a way similar to "pseudo-relevance feedback" in information retrieval.

A more often used method trains a statistical translation model (TM) from a parallel corpus. Nie et al. [19] is among the first ones to use this method for CLIR. They build a probabilistic translation model from a parallel corpus. The top translation words proposed by the TM are kept as the translation of a query. This study showed that the retrieval effectiveness obtained is very close to that using a good MT system (Systran). A series of other papers, such as [10, 11, 13], follow the same direction to integrate TM to CLIR. In particular, Kraaij et al. [14] has tested the integration of query translation into a global language model. They showed that this integrated approach outperforms the existing machine translation system (Systran).

A translation model is a mathematical model, which gives the conditional probability $P(T|S)$, i.e. the likelihood of translation a source language string $S$ into a target language string $T$. Different TMs use different methods to align words between source and target languages. The main single-word-based alignment methods are IBM 1 to 5 [3] and Hidden-Markov alignment model [24]. These models use words as the basic translation units. For Chinese, it is assumed that a sentence is segmented into words. Then the same approach can be used for Chinese. Word-based translation approach has been used in all the previous studies on Chinese translation using parallel corpora. However, as shown in monolingual IR, a Chinese sentence can also be segmented into n-grams of characters (unigrams or bigrams). Therefore, an alternative query translation method is to use n-grams of Chinese characters as translation units. This possibility has not been studied previously. This is the focus of this paper.

## 2.3 Language Modeling Approach to CLIR

Statistical language modeling is an approach widely used in current IR research. Compared to the other approaches (e.g. vector space model), it has the advantage that different factors of IR can be integrated in a principled way. For example, unlike in vector space model, term weighting becomes an integral part of the retrieval model in language modeling. In addition, LM can also integrate easily query translation, as well as considering multiple indexing units in Chinese. Therefore, we will use an LM approach in this paper.

The basic approach of language modeling to IR is to build a statistical language model for each document, and then determine the likelihood that the document model generates the query [4] [21]. An alternative is to build a language model for each document as well as for the query. A score of document is determined by the difference between them. A common score function is defined by the negative Kullback-Leibler divergence or relative entropy as follows:

$$\begin{aligned} Score(D,Q) &= -KL(\theta_Q \,||\, \theta_D) \\ &= -\sum_{w \in V} p(w|\theta_Q) \log \frac{p(w|\theta_Q)}{p(w|\theta_D)} \\ &\propto \sum_{w \in V} p(w|\theta_Q) \log p(w|\theta_D) \end{aligned}$$

where $\theta_Q$ and $\theta_D$ are the parameters of language model for query $Q$ and document $D$ respectively, $V$ is the vocabulary of the language. The simplest way to compute query model $p(w|\theta_Q)$ is estimating probability by the maximum likelihood according to query text. For document model, it is necessary to use a certain smoothing method, such as absolute discounting, Jelinek-Mercer, Dirichlet prior, etc., to deal with the problem of zero-probability for the missing words in the document [27].

In CLIR, words in $Q$ and $D$ are in different languages. Query translation can be integrated into the query model $p(w|\theta_Q)$ formulas follows:

$$\begin{aligned} p(t_i \mid \theta_{Q_s}) &= \sum_{s_j} p(s_j, t_i \mid \theta_{Q_s}) \\ &= \sum_{s_j} t(t_i \mid s_j, \theta_{Q_s}) p(s_j \mid \theta_{Q_s}) \\ &\approx \sum_{s_j} t(t_i \mid s_j) p(s_j \mid \theta_{Q_s}) \end{aligned} \quad (1)$$

where $s_j$ is a word in source language, $t_i$ is a word in target language, $t(t_i|s_j)$ is a translation probability between $s_j$ and $t_i$. This probability is provided by a translation model trained on a parallel corpus. In our case, we use IBM model 1 [3] trained using GIZA++ toolkit[1]. We will provide some details about the model in section 4. A similar approach has been used in [14] for CLIR between European languages, in which $s_j$ and $t_i$ are words.

For CLIR with Chinese (as the target language), $t_i$ can either be words or n-grams. Therefore, we are faced with an additional problem of choosing between, or combining, different indexing units.

# 3 INDEXING AND TRANSLATION UNITS

## 3.1 Monolingual IR

Let us first re-examine the problem of monolingual IR in Chinese, as CLIR will strongly rely on it.

Several studies have compared the utilizations of words and n-grams as indexing units for Chinese IR [20] [17]. Most of them have been done in models other than language modeling. Here, we re-examine the impact of different indexing units within the language modeling framework.

Previous studies on Chinese word segmentation showed that segmentation accuracy in Chinese is usually higher than 90% [6, 16, 25]. This accuracy is shown to be satisfactory for IR [18]. So, in our study, we do not compare different Chinese word segmentation methods. We only use one segmentation method and focus on the differences between words and n-grams.

A Chinese word is composed of one, two, or more Chinese characters. Nie et al. [20] shows that the average length of Chinese words is 1.59 characters. It means that most Chinese words have only one or two characters. So, by considering bigrams, most Chinese words can be correctly covered. Although some longer words cannot be represented accurately by bigrams, the extension from bigrams to longer n-grams has a cost: there will be much more n-grams to be stored as indexes, and the complexity both in space and retrieval time will increase substantially. Therefore, limiting n-grams to length 2 is a reasonable compromise. So, besides words, we will consider only unigrams and bigrams.

Using a word segmentation method, a sentence can be transformed into a sequence of words. Then the same word-based method used for European languages can also be used for Chinese. For example, the sentence "国企增加研发投资" (National enterprises increase the investment in R&D) can be segmented into: "国企/增加/研发/投资".

However, this example also shows an important problem: the same meaning can be expressed in multiple ways. For example, 研发 (R&D) can be expressed as 研究和开发 (research and development). If only 研发 is used as index, then it will not be able to match against 研究和开发. This problem is similar to that of abbreviation in European languages (such as "R&D"). However, we argue here that the phenomenon is more frequent in Chinese. Very often new abbreviations are easily created. For example, 国营企业 (national enterprises) can be abbreviated to 国企 (as in our example). In addition, Chinese also has a large number of similar words to express the same meaning. For example, 增大, 猛增, 递增, 加大, etc. can all express the same (or a similar) meaning as (to) 增加(increase). A strategy that only uses words as indexing units will very likely miss the corresponding words.

We notice in the above example of "increase" that many similar Chinese words share some common characters. Therefore, a natural extension to word-based indexing of documents and queries is to add characters as additional indexing units. By adding 国, 企, 增, 加, 研, 发 as additional indexes, we will create partial matches with other words expression "national enterprises", "increase" and "R&D", thereby increase recall. Although this approach is unable to cover all the alternative expressions, it has been shown to be effective for Chinese IR [17, 20].

An alternative to word segmentation is to cut a Chinese sentence into overlapping bigrams such as: 国企/企增/增加/加研/研发/发投/投资. Compared to word segmentation, this approach has the advantage that no linguistic resource (such as dictionary) is required. In addition, new words can be better represented. For example, suppose 新译林 is a new word (possibly the name of a magazine), which is not stored in the dictionary. Then it is likely segmented into three separate characters 新/译/林 using a word segmentation approach. If we use bigrams, the sequence 新译/译林 will be generated. These latter can better reflect the sequence 新译林 than the three separate characters.

A possible problem with bigrams is that many of them do not correspond to valid semantics. In the earlier example, 企增,加研 and 发投 do not correspond to any valid meaning. However, it can be expected that their frequency of occurrences in documents will be much lower than the valid parts 国企, 增加, 研发 and 投资. Therefore, there is a natural selection of valid bigrams by the corpus statistics.

The above observation has been made in several previous studies [17] [20]. However, words and bigrams have often been used as two competitive approaches instead of combining them. In [20], it is found that the most effective approach is to segment sentences into words but also add the characters. For example, the sequence

[1] http://www.fjoch.com/GIZA++.html

国企增加研发投资 is segmented into 国企/增加/研发/投资/国/企/增/加/研/发/投/资. The addition of single characters (or unigrams) allows us to extend the words to related ones.

However, this is not the only possible approach to combine words and n-grams. Several alternative approaches are possible: we can create several indexes for the same document: using words, unigrams and bigrams separately. Then during the retrieval process, these indexes are combined to produce a single ranking function. In LM framework, this means that we build several language models for the same document and query. Each type of the model determines a score $Score_i$. The final score is a combination of the scores. So, in general, we define the final score as follows:

$$Score(D,Q) = \sum_i \alpha_i Score_i(D,Q)$$

where $Score_i$ is the score determined by a type of model (in our case, either unigram, bigram or word model) and $\alpha_i$ its importance in the combination (with $\sum_i \alpha_i = 1$). In particular, we can have the following possible basic indexing strategies:

- W (Word): segment sentences into words, and only use the word model for retrieval
- U (Unigram): segment sentences into unigrams (single characters), and only use unigram model for retrieval.
- B (Bigram): segment sentences into overlapping bigrams of characters.
- WU (Word+Unigram): segment sentences into both words and unigrams, as in [20].
- BU (Bigram+Unigram): segment sentences into both overlapping bigrams of characters and unigrams.

These strategies can then be combined according to Formula (2). For example, we can combine word and unigram models, bigram and unigram models, or word, bigram and unigram models, which we denote respectively by W+U, B+U and W+B+U as follows:

$$\begin{aligned} Score_{W+U}(D,Q) &= \lambda Score_W + (1-\lambda) Score_U \\ Score_{B+U}(D,Q) &= \lambda Score_B + (1-\lambda) Score_U \\ Score_{B+W+U}(D,Q) &= \lambda_b Score_B + \lambda_w Score_w + \lambda_u Score_U \end{aligned} \quad (2)$$

where $0 \le \lambda \le 1$, $\lambda_b+\lambda_w+\lambda_u=1$.

## 3.2 Creating Different Translation Models for CLIR

For CLIR, we use a TM to translate query $Q_s$ from source language to target language.

Here, we use maximum likelihood estimation to estimate the source terms in the query, that is: $P(s_j \mid \theta_Q) = \frac{c(s_j, Q_s)}{|Q_s|}$. The query model in Formula (1) becomes:

$$p(t_i \mid \theta_{Q_s}) = \sum_{s_j \in Q_s} t(t_i \mid s_j) \frac{c(s_j, Q_s)}{|Q_s|} \quad (3)$$

where $c(s_j,Q_s)$ is occurrence of term $s_j$ in query $Q_s$, and $|Q_s|$ is the number of terms in $Q_s$.

The simplest TM is English-Chinese word-to-word translation model, which can be trained from English-Chinese parallel corpus (in which Chinese sentences are segmented into words). If only words are used, then we will have a TM translating English words into Chinese words. We denote this translation approach by W. To improve the retrieval coverage (recall) in CLIR, we can use the same method as in monolingual IR: we expand each Chinese word sequence in the parallel texts by adding the unigrams. The resulting translation model will suggest both Chinese words and characters as translations of English words. We denote this translation model by WU. The addition of single characters into parallel sentences aims to deal with the same problem as in monolingual Chinese IR. For example, if only 国家(country) is segmented as a word in a parallel sentence, then this word will be suggested as the only translation candidate for "country". In fact, 国 (country) is another reasonable alternative for the same meaning. Therefore, by adding single characters into the training sentence, the TM can also suggest 国 as another translation candidate to "country". This approach is simple. We only need to perform the following transformation of each parallel sentence:

$$e_1 e_2 ... e_n \parallel w_1 w_2 ... w_m \Rightarrow e_1 e_2 ... e_n \parallel w_1 w_w ... w_m c_1 ... c_k$$

where $e_i$ is an English word, $w_i$ is a Chinese word, $c_i$ is a Chinese character included in $w_1 ... w_m$. GIZA++ is then used to create an IBM 1 model. Now, the word "country" is translated into not only 国家(country): 0.2216, but also 国(country): 0.2501, 家(home): 0.1871, etc.

In the same way, if we append characters to bigrams, the resulting TM will translate an English word to Chinese bigrams and unigrams.

Now we show how these TM are used for CLIR. Firstly, we notice that the translation candidates with low probabilities usually are not strongly related to the query. They are more noise than useful terms. So, we remove them by setting a threshold δ: we filter out the items $t_i$ with $t(t_i \mid s_j) < \delta$. Then, the probabilities of the remaining translation candidates are re-normalized so that $\sum_{t_i} t(t_i \mid s_j) = 1$.

Then, we calculate the query model by Formula (3). To further reduce the noise, we use one of the following two methods to select translations:

(1) For each source term $s_j$, we select the top $N$ best translations.
(2) We sort of the translation candidates by $p(t_i \mid \theta_{Q_s})$ according to Formula 1 and select the top $N*|Q_s|$ terms as translation.

Here $N$ is a fixed parameter that we can tune manually.

## 3.3 Using Co-Occurrence Terms

Translation models are created for word translation. That is, the translation of a word only depends on the source word in isolation. In many cases, a single word is ambiguous. For example, the word "intelligence" has several meanings. It can be translated into Chinese as 智能, 情报, etc. In order to solve the

ambiguities, several studies have exploited the context words to determine the most appropriate translation candidates. For example, Gao et al [11] uses a cohesion measure between the translation candidates for different source words to select the ones with the highest cohesion. Ballesteros and Croft [1] uses co-occurrence statistics for translation disambiguation.

However, all these studies focus on the selection ambiguous translations in the target language afterwards. In [2], a different approach has been proposed to suggest related words for query expansion according to more than one query word at each time. For example, instead of using ambiguous term relations "Java→programming" and "Java→island", we include more than one term in the condition: "(Java, computer) → programming", where "(Java, computer)" means that the two words co-occur in some window. By adding more terms into the condition, the derived term is more strongly related to the query, and it is context-dependent.

In this study, we use the same idea but for query translation: In order to determine a target language translation, we make use of more than one source language word. For example, if "java" co-occurs with "computer", then the probability of translating them to 程序 (program) and Java 语言 (Java language) will be much higher than to 瓜哇岛 (Java island), i.e., $t($程序$|$ java,computer$)>>t($岛$|$ java,computer$)$.

In order to obtain such context-dependent translation relations, we perform a co-occurrence analysis on the parallel texts. As in [2], we also limit the condition part of the translation relations to two words.

The first question is what pair of words can be considered as meaningful pairs for translation. A meaningful pair of words is the one that brings more information than the two words separately. Several statistical measures have been proposed to determine such pairs [23], including t-score, Pearson's $\chi^2$, log-likelihood ratio, pointwise mutual information and mutual dependency. The results show that log-frequency biased mutual dependency (LFMD) and log-likelihood ratio (LLR) outperform the other methods. Therefore, we choose the LLR method for identifying meaningful co-occurrence words. LLR of words $w_1$ and $w_2$ is determined in as follows [9]:

$$LLR(w_1, w_2) = -2\log\frac{L(H_0)}{L(H_1)} = -2\log\frac{L(c_{12}, c_1, p)\cdot L(c_2 - c_{12}, N - c_1, p)}{L(c_{12}, c_1, p_1)\cdot L(c_2 - c_{12}, N - c_1, p_2)}$$

where $H_0$ is the hypothesis of $P(w_2|w_1)=p=P(w_2|\neg w_1)$, and $H_1$ is $P(w_2|w_1)=p_1\neq p_2=P(w_2|\neg w_1)$; $L(k,n,x)=x^k(1-x)^{n-k}$; $c_1$, $c_2$, and $c_{12}$ are the occurrences of $w_1$, $w_2$ and $w_1w_2$ respectively; $p=c_2/N$, $p_1=c_{12}/c_1$, $p_2=(c_2-c_{12})/(N-c_1)$. Usually, the co-occurrence of words should be limited within the same context (paragraph or sentence) and not far away from each other. We also limit word co-occurrences in the same sentence and within a fixed size of window – *win_size*. We apply a threshold to filter out word pairs with low LLR values, and keep the remains word pairs in a list of meaningful word pairs.

Now, we can extend the source sentences of parallel corpus. For all words $e_i$ and $e_j$, if the distance between them is less than *win_size* and they are in the list of meaningful word pairs, we add the pair $e_i\_e_j$ into the source sentence as follows:

*Original sentence pair:* $e_1 \ldots e_n \| w_1 \ldots w_n$
*Transformed pair:* $e_1 \ldots e_n\, e_{i\_}e_j \ldots \| w_1 \ldots w_n$

With the word pairs added, we train a translation model (IBM model 1), which include two types of translation: one is from English word to Chinese words, $TM_0$; another is from English word pair to Chinese words, $TM_{co}$.

The above approach can be viewed as a way to integrate the translation of compound terms. However, this approach is more flexible than that using compound terms – the determination of compounds usually require stricter syntactic constraints between compounds, while in our method words can freely group to form word pairs provided that they appear together often. Not only this method has a larger coverage, but also it can consider the influence of any useful context word in translation of a word without requiring them to form a compound term.

The final question is how these translation relations can be used for query translation. The basic idea is adjusting the probabilities of $TM_0$ according to $TM_{co}$ in the sentence context. The translation probabilities (in $TM_0$) should be boosted if the translations are also proposed by the co-occurrence translation model ($TM_{co}$), and decreased otherwise. The translation model in Formula (1) is then defined as follows:

$$t(c|e_i,\theta_Q) = \sum_{e_j\in Q}(1-\alpha(e_i,e_j))t_0(c|e_i) + \alpha(e_i,e_j)\cdot t_{co}(c|e_{i\_}e_j) \quad (4)$$

where the parameter $\alpha(e_i,e_j) \propto LLR(e_i,e_j)$, which is a value within the range [0,1], is a confidence factor measuring how strong the two words are related in the query. The final translation probability for each $e_i$ is then normalized so that $\sum_c t(c|e_i,\theta_Q) = 1$.

# 4. EXPERIMENTS OF ENGLISH-CHINESE CLIR

## 4.1 The Experiment of Chinese Monolingual IR

We use Lemur toolkit[1] with KL-divergence and Dirichlet prior smoothing method. We evaluated the monolingual IR and CLIR using two TREC collections and three NTCIR collections: TREC5, 6, and NTCIR3, 4, 5. The statistics are described in Table 1.

**Table 1. Collection and topic description**

| Collection | Description | Size (MB) | #Doc | #Topic |
|---|---|---|---|---|
| TREC5 | Peoples Daily & Xinhua news agency | 173 | 165K | 28 |
| TREC6 | Peoples Daily & Xinhua news agency | 173 | 165K | 26 |
| NTCIR3 | CIRB011&CIRB020 | 543 | 381K | 50 |
| NTCIR4 | CIRB011&CIRB020 | 543 | 381K | 60 |
| NTCIR5 | CIRB040 | 1106 | 901K | 50 |

[1] http://www.lemurproject.org

Table 2 gives the retrieval results measured in MAP (Means Average Precision), where for each of collection, we obtain two results: one with "title" of each topic as the query, the other with "title+description" as query. We use different index and retrieval units described in Section 3: word segmentation (W), bigrams (B), Unigrams (U), mixture of words and unigrams (WU), mixture of bigrams and unigrams (BU). In addition, we also tested several combinations of these indexing methods, by combining their ranking scores. Namely, we combined W and U indexes (W+U) as well as B and U indexes (B+U). We vary the combination factor of Formula (2) from 0.1, 0.2,…, to 0.9, and results show that when we attribute around 0.3 to W or to B and 0.7 to U, we obtain the best performances. When combining W, B, and U (W+B+U), we tune the parameters manually. On average, $\lambda_u$=0.6, $\lambda_w$=$\lambda_w$=0.2 gives best results.

We can observe that using words (W) or using bigrams (B) as indexing units, we obtain quite similar results. This is consistent with the observations in previous studies. What is surprising in our experiments is that using unigrams alone (U), we can also obtain very good results, which are even better than W and B. In some previous studies, unigrams have not been found to be as effective as bigrams [20]. We believe that the difference may be due to the use of different retrieval models: we use language modeling approach which is different from previous ones. The language modeling may have a capacity to extract discriminative unigrams higher than the other models. Even if characters are not always meaningful, their probabilities are assigned in LM in such a way that more meaningful characters are attributed more different probabilities in different documents. These characters will make more difference between documents, thus affect document ranking more. This capability of LM to consider discrimination values of indexes is analyzed in [27].

**Table 2. Comparing Chinese monolingual IR results**

| **Chinese Monolingual IR (Query: Title)** | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **Collec-tions** | **W** | **B** | **U** | **WU** | **BU** | **.3W +.7U** | **.3B+ .7U** | **W+ B+U** |
| TREC5 | .2585 | .2698 | .3012 | .3298 | .3074 | .3123 | .3262 | .3273 |
| TREC6 | .3861 | .3628 | .3580 | .4220 | .3897 | .4090 | .3880 | .4068 |
| NTCIR3 | .2609 | .2492 | .2496 | .2606 | .2820 | .2754 | .2840 | .2862 |
| NTCIR4 | .1996 | .2164 | .2371 | .2254 | .2350 | .2431 | .2429 | .2387 |
| NTCIR5 | .2974 | .3151 | .3390 | .3118 | .3246 | .3452 | .3508 | .3470 |
| **Average** | **.2805** | **.2827** | **.2970** | **.3099** | **.3077** | **.3170** | **.3184** | **.3212** |
| **(Query: Title + Description)** | | | | | | | | |
| TREC5 | .3240 | .3496 | .3433 | .3553 | .3553 | .3581 | .3693 | .3668 |
| TREC6 | .4909 | .5068 | .4709 | .5095 | .5165 | .5165 | .5116 | .5269 |
| NTCIR3 | .2822 | .2692 | .2672 | .2788 | .2766 | .3118 | .3080 | .3167 |
| NTCIR4 | .2122 | .2074 | .2390 | .2195 | .2170 | .2464 | .2443 | .2449 |
| NTCIR5 | .3386 | .3490 | .3741 | .3421 | .3516 | .3858 | .3942 | .3869 |
| **Average** | **.3296** | **.3364** | **.3389** | **.3410** | **.3434** | **.3637** | **.3654** | **.3684** |

When we mix up two types of indexing units in the segmentation step – W with U (WU) and B with U (BU), we can see that the results are generally better than when only one type of index is used. This observation is consistent with [20].

However, the best methods are those that create separate indexes for each type, and then combine the ranking score according to Formula (2). The result of combining word, bigram and unigram together shows that this approach can produce slightly better results than W+U and B+U, but the improvements are marginal. A possible reason is that words are usually formed with two characters. So there is a large overlap between words and bigrams. As a consequence, once words have been used, bigrams do not bring much new information, and vice versa.

Overall, comparing W to B, we obtain comparable effectiveness, either when they are used alone or they are combined each with U. Therefore, we can conclude that bigrams are reasonable alternative to words as indexing units. The combination between them does not seem to be interesting. This shows that both types of index captures about the same information. On the other hand, unigrams are complementary to them and it is useful to combine unigrams with either bigrams or words.

## 4.2 Parallel Corpus Preprocessing

Our model requires a set of parallel texts to train a TM. We have implemented an automatic mining tool to mine Chinese-English parallel texts from Web using a similar approach to [5]. Parallel texts are mined from six websites, which are located in United Nations, Hong Kong, Taiwan, and Mainland of China (Chinese pages encode in GB2312, Big5, and Unicode). It contains about 4000 pairs of pages and includes some noise (non-parallel texts).

After converting the HTML texts to plaint texts and mark the paragraph and sentence boundaries, we use a sentence alignment algorithm to align the parallel text to sentence pairs. Our sentence alignment algorithm is an extension of the length-based method, which also considers the known lexical-translation according to a bilingual dictionary. The idea is that if a pair of sentences contains many words that are mutual translations in the dictionary, then their alignment score is increased. Here we use CEDICT[1], which includes 28,000 Chinese words/phrases. After sentence alignment, we obtain 281,000 parallel sentence pairs. Another extension we made to the traditional TM training is to use sentence alignment score during TM training. A pair of sentences with a higher score is considered more important in the training process than a pair with lower score. This factor can be easily incorporated into the GIZA++ tool. Our previous experiments showed that these measures result in better translation models and higher CLIR effectiveness [22]. In this study, we use the same approach for TM training.

For English, we use a simple morphological analyzer[2] to remove the English language suffixes, such as *-s*, *-ed*, *-en*, *ase*, *-yl*, *-ide*, etc. For Chinese word segmentation, we use an existing segmentation tool[3]. The segmenter uses a version of the maximal matching algorithm based on a lexicon.

Once the parallel corpus has been pre-processed as above, GIZA++ is used to train translation models - IBM model 1.

## 4.3 Using Different Chinese Translation Units for CLIR

When preprocessing Chinese texts in the parallel corpus, different Chinese units have been created separately. We therefore obtain several types of translation model:

- W: English word to Chinese words;
- B: English word to Chinese bigrams;
- U: English word to Chinese unigrams (single characters);

[1] http://www.mandarintools.com/cedict.html
[2] http://web.media.mit.edu/~hugo/montylingua/
[3] http://www.mandarintools.com/download/segment.zip

- WU: English word to Chinese words and unigrams;
- BU: English word to Chinese bigrams and unigrams.

In our experiments, we set $N$=10 and use the second method introduced in Section 3.2, i.e. keep top $10 \cdot |Q|$ target words in query model. This method is slightly better than the first one. As for monolingual IR, when two function scores are combined using Formula (2), we set $\lambda$=0.3 for either W or B models. The CLIR results (measured in MAP) are shown in Table 3

We can observe that in general, CLIR effectiveness is much lower than monolingual effectiveness. This is normal and consistent with previous studies. Although we can expect a quite high effectiveness for CLIR between European languages, in general, the CLIR effectiveness between English and Chinese is much lower than monolingual effectiveness. So, the drop we observe here is not an exception.

What is important to observe is the comparison between different translation approaches.

As for monolingual IR, we see that using W or B as translation units, we can obtain similar results. Using U as translation units, we obtain generally better effectiveness. This result is also new compared to the previous studies. This shows that Chinese characters can be reasonable indexing and translation units for Chinese.

When we mix up Chinese units in TM (WU and BU), we can obtain further improvements. On the other hand, although it is still an interesting approach to translate the query into different units with different TMs and then combine their ranking scores by Formula (2), we do not observe any significant increase using this last approach over WU and BU, contrarily to monolingual IR.

**Table 3. CLIR results using different translation models**

| English→Chinese CLIR (Query: Title) | | | | | | | |
|---|---|---|---|---|---|---|---|
| **Collec-tions** | **W** | **B** | **U** | **WU** | **BU** | **0.3W+ 0.7U** | **0.3B+ 0.7U** |
| TREC5 | .1904 | .2003 | .1922 | .2448 | .2277 | .2158 | .2251 |
| TREC6 | .2047 | .2293 | .2602 | .2670 | .2772 | .2672 | .2822 |
| NTCIR3 | .1288 | .1017 | .1536 | .1628 | .1504 | .1619 | .1495 |
| NTCIR4 | .0956 | .0953 | .1382 | .1410 | .1308 | .1337 | .1286 |
| NTCIR5 | .1158 | .1323 | .1762 | .1532 | .1462 | .1682 | .1602 |
| **Average** | **.1470** | **.1518** | **.1841** | **.1938** | **.1865** | **.1894** | **.1891** |
| **(Query: Title + Description)** | | | | | | | |
| TREC5 | .2433 | .2637 | .2674 | .2984 | .2897 | .2848 | .2906 |
| TREC6 | .2910 | .3355 | .3624 | .3745 | .3866 | .3641 | .3793 |
| NTCIR3 | .1401 | .1189 | .1741 | .1878 | .1748 | .1977 | .1731 |
| NTCIR4 | .1021 | .0992 | .1463 | .1493 | .1390 | .1443 | .1395 |
| NTCIR5 | .1315 | .1430 | .2252 | .1851 | .1731 | .2051 | .2053 |
| **Average** | **.1816** | **.1921** | **.2351** | **.2390** | **.2326** | **.2392** | **.2376** |

## 4.4 Using English Word Pairs for Translation

To determine meaningful English word pairs, we use the monolingual English corpus, Associate Press (AP88-90). We filtered out the word pair which LLR less than 100, and kept 828,750 pairs.

The new translation method is compared to the translation method WU, which proved to be the most effective. Here, in addition to segmenting Chinese sentences into both words and unigrams, we also group English words to form an additional term. Finally, we trained a TM ($TM_{co}$) from English to Chinese that also contains translations of English word pairs. Using Formula (4), we can get the new model that we denote by $WU_{co}$ in the following table.

**Table 4. Comparing different translation approach (Documents are indexed by WU in both cases)**

| Collections | Query: Title | | | Query: Title + Description | | |
|---|---|---|---|---|---|---|
| | **WU** | **$WU_{co}$** | | **WU** | **$WU_{co}$** | |
| | **MAP** | **MAP** | **%of WU** | **MAP** | **MAP** | **%of WU** |
| TREC5 | .2448 | **.2463** | +0.6 | **.2984** | .2910 | -2.5 |
| TREC6 | .2670 | **.2912** | +9.1 | .3745 | **.3883** | +3.7 |
| NTCIR3 | .1628 | **.1656** | +1.7 | **.1878** | .1869 | -0.5 |
| NTCIR4 | .1410 | **.1448** | +2.7 | .1493 | **.1536** | +2.9 |
| NTCIR5 | .1532 | .1586 | +3.5 | .1851 | .2008 | +8.5 |

We can see that when meaningful English word pairs are considered in the translation model, the resulting retrieval effectiveness is slightly higher than the WU translation model. However, the improvements are not consistent in all cases.

For some queries, we observe that this new translation model can produce better translation. For example, for TREC6 topic CH45, The MAP of WU is 0.2514 and that of $WU_{co}$ is 0.6439. The English title is "*China red cross*". By the WU translation model this topic is translated to "红:0.5388 中国: 0.3842 中:0.3427 国:0.2650 两:0.1336 两岸:0.0837 跨:0.0760 十:0.0720 岸:0.0718 …" The underlined Chinese words are correct translations. Once we combine $TM_0$ and $TM_{co}$ by Formula (4), the translation becomes "中国:0.3842 中:0.3427 红:0.3007 国:0.2650 十: 0.2362 字:0.2292 红十字会:0.1662 两: 0.1025 会:0.0901 两岸 0.0642 …" We see that the translation is more related to the original query.

For some other queries, we observed decreases in effectiveness. This is the case for TREC6 topic CH24, for which the effectiveness drops from 0.3216 to 0.2437. The English title is "*Reaction to Lifting the Arms Embargo for Bosnian Muslims*". For this query, we have determined correctly "arm_embargo" as a word pair. Its translation should be "武器(weapon,arms)/禁运 (embargo)". However, due to the limitation of our parallel corpus, the translations of "arm_embargo" in $TM_{co}$ are " 运 (transport):0.1045 安全 (safe):0.1025 安 (safe):0.0813 全 (complete):0.0734 禁 (forbid):0.0654 表 :0.0576 禁运 (embargo):0.0386 生 :0.0348 发 :0.0339…" We see that the meaning of "weapon" is completely lost and the meaning of "embargo" is only reflected by two low probability translations. Therefore, the result becomes worse. We believe that this decrease is largely due to the limited size of our parallel corpus and its coverage of Chinese and English words. With a larger parallel corpus, the translation model with word pairs should be able to produce larger improvements in retrieval effectiveness.

Another factor that strongly impacts this method is that we have normalized the influence of each translation component in Formula (4). That is, when an English word is contained in a word pair, both types of translation are combined. If a word is not part of a word pair, then only word-based translation is considered. In

this case, the word-based translation will be attributed with a higher weight (because it is attributed the whole relative importance, or $\alpha(e_i, e_j) = 0$ in Formula (4)). This may raise some problem. Indeed, when a single word is translated, much ambiguity is introduced. Therefore, we should rather reduce our confidence on the translations from single English words. This is a problem that we will consider in our future research.

# 5. CONCLUSION AND FUTURE WORK

Chinese words and bigrams have been considered to be two competitive indexing units for Chinese IR. In this study, we further compared these approaches and combine them with unigrams (characters). We have found that Chinese unigrams are even more effective than either words or bigrams. This result is new in Chinese IR. We also show that by combining either words or bigrams with unigrams, we can obtain better retrieval effectiveness. This result is consistent with previous studies.

For CLIR with Chinese (as the target language), previous studies usually use words as translation units. In this paper, we have investigated the possibility to use bigrams and unigrams as alternative translation units. Our experiments showed that these translation units are as effective as words. In particular, unigrams have proven to be even more effective than words and bigrams.

Combining the above results, we can see that Chinese characters are very meaningful units, which can be used as both indexing and translation units.

When an English query is translated to both unigrams and words or bigrams, we observed slightly higher retrieval effectiveness. However, the increase is marginal.

We also tested the possibility to determine Chinese translation from a pair of English words in order to reduce translation ambiguity. For some queries, the results are very interesting, but for some others, we observed rather a decrease. Therefore, the global effectiveness is only marginally changed. Despite of this fact, we believe that this new translation method can be further improved on the following aspects:

- Using a large parallel corpus, we can derive more useful translation from English word pairs;
- We can improve the way to combine the translation based on word pairs and those based on words. In our current implementation, we only considered the strength of link between the English words. This may not be reasonable. We have to define a better measure of confidence about the translations generated from single words or word pairs.

We will investigate these problems in our future research. It would be interesting to test our approaches also for other Asian languages such as Japanese and Korean.

# 6. REFERENCES


[1] L. Ballesteros and B. Croft. "Resolving ambiguity for cross-language retrieval," *SIGIR,* pp.64-71,1998

[2] J. Bai, J.Y. Nie, and G. Cao. "Context-dependent term relations for information retrieval," *EMNLP*, pp.551-559, 2006.

[3] P. F. Brown, S. A.D. Pietra, V. J.D. Pietra, and R. L. Mercer. "The mathematics of statistical machine translation: parameter estimation," *Computational Linguistics,* 19(2): 263-311, 1993.

[4] W. Bruce Croft. "Language Models for Information Retrieval," in *Proceeding of the 19th International Conference on Data Engineering,* pp.3-7, 2003.

[5] J. Chen and J.Y. Nie. "Automatic construction of parallel English-Chinese corpus for cross-language information retrieval," *ANLP*, Seattle, Washington, pp.21-28, 2000.

[6] K. Chen and S. Kiu. "Word identification for Mandarin Chinese sentences," In *5th International Conference on Computational Linguistics*: 101-107, 1992.

[7] L.F. Chien. "Fast and quasi-natural language search for gigabytes of Chinese texts," *SIGIR*, pp.112-120, 1995.

[8] M. W. Davis and W. C. Ogden. "QUILT: implementing a large-scale cross-language text retrieval system," *SIGIR,* pp.92-98*, 1997*.

[9] T. Dunning. "Accurate methods for the statistics of surprise and coincidence," *Computational Linguistics*, 19(1): 61–74, 1993.

[10] J. Gao, J.Y. Nie, E. Xun, and et al. "Improving Query Translation for Cross-Language Information Retrieval using Statistical Models," *SIGIR*, pp.96-104, 2001.

[11] J. Gao, J.Y. Nie, H. He, et al. "Resolving query translation ambiguity using a decaying co-occurrence model and syntactic dependence relations," *SIGIR,* pp.183-190, 2002.

[12] T. Liang, S.Y. Lee, and W.P. Yang. "Optimal weight assignment for a Chinese signature file," *Information Processing and Management: an International Journal,* 32(2): 227-237, 1996.

[13] R. Jin and J.Y. Chai. "Study of cross lingual information retrieval using on-line translation systems," *SIGIR* pp.619-620, 2005.

[14] W. Kraaij, J.Y. Nie, and M. Simard. "Embedding Web-based statistical translation models in cross-language information retrieval," *Computational Linguistics.* 29(3): 381-419, 2003.

[15] K.L. Kwok. "Comparing representations in Chinese information retrieval," *SIGIR,* pp.34-41, 1997.

[16] B. Li, S. Lien, C Sun, and M. Sun. "A maximal matching automatic Chinese word segmentation algorithm using corpus tagging for ambiguity resolution," *R.O.C. Computational Linguistics Conference (ROCLING-IV),* Taiwan, pp. 135-146 , 1991.

[17] R.W.P. Luk, K.F. Wong, and K.L. Kwok. "A comparison of Chinese document indexing strategies and retrieval models", *ACM Trans. Asian Lang. Inf. Process,* 1(3): 225-268, 2002.

[18] J.Y. Nie, M. Brisebois, and X. Ren. "On Chinese text retrieval," *SIGIR 1996*, pp.225-233.

[19] J.Y. Nie, M. Simard, P. Isabelle, and R. Durand. "Cross-language information retrieval based on parallel texts and automatic mining of parallel texts from the Web," *SIGIR 1999*, pp.74-81.

[20] J.Y. Nie, J. Gao, J. Zhang, and Zhou, M. "On the use of words and n-grams for Chinese information retrieval," In *Fifth International Workshop on Information Retrieval with Asian Languages, IRAL,* Hong Kong, 2000.

[21] J. Ponte and W.B. Croft. "A language modeling approach to information retrieval," *SIGIR*, pp.275-281, 1998.

[22] L. Shi, J.Y. Nie. "Filtering or adapting: two strategies to exploit noisy parallel corpora for cross-language information retrieval," *CIKM*, pp.814-815, 2006.

[23] A. Thanopoulos, N. Fakotakis, and G. Kokkinakis. "Comparative evaluation of collocation extraction metrics," in *Proceedings of 3rd International Conference on Language Resources and Evaluation (LREC'02)*, volume 2, pp.620-625, 2002.

[24] S. Vogel, H. Ney, and C. Tillmann. "HMM-based word alignment in statistical translation," *COLING*, pp.836-841, 1996.

[25] T. Yao, G. Zhang, and Y. Wu. "A rule-based Chinese automatic segmentation system," *Journal of Chinese Information Processing,* ,4(1): 37-43, 1990.

[26] Y. Yang, J. G. Carbonell, R. D. Brown, and R. E. Frederking. "Translingual information retrieval: learning from bilingual corpora," *Artificial Intelligence,* 103: 323-345, 1998.

[27] C. Zhai and J. Lafferty. “A study of smoothing methods for language models applied to ad hoc information retrieval,” *SIGIR*, pp.334-342, 2001.