Extending Query Translation to Cross-Language Query Expansion with Markov Chain Models

Guihong Cao
Département d'Informatique et de Recherche Opérationnelle,
Université de Montréal
caogui@iro.umontreal.ca

Jianfeng Gao
Microsoft Research
Redmond, WA
jfgao@microsoft.com

Jian-Yun Nie, Jing Bai
Département d'Informatique et de Recherche Opérationnelle,
Université de Montréal
{nie, baijing}@iro.umontreal.ca

ABSTRACT

Dictionary-based approaches to query translation have been widely used in Cross-Language Information Retrieval (CLIR) experiments. Using these approaches, translation has been not only limited by the coverage of the dictionary, but also affected by translation ambiguities. In this paper we propose a novel method of query translation that combines other types of term relation to complement the dictionary-based translation. This allows extending the literal query translation to related words, which produce a beneficial effect of query expansion in CLIR. In this paper, we model query translation by Markov Chains (MC), where query translation is viewed as a process of expanding query terms to their semantically similar terms in a different language. In MC, terms and their relationships are modeled as a directed graph, and query translation is performed as a random walk in the graph, which propagates probabilities to related terms. This framework allows us incorporating different types of term relation, either between two languages or within the source or the target languages. In addition, the iterative training process of MC allows us to attribute higher probabilities to the target terms more related to the original query, thus offers a solution to the translation ambiguity problem. We evaluated our method on three CLIR benchmark collections, and obtained significant improvements over traditional dictionary-based approaches.

Categories and Subject Descriptors

H.3.3 [Information Storage and Retrieval]: Retrieval models

General Terms

Design, Algorithm, Theory, Experimentation

Keywords

Query Translation, Query Expansion, Cross-Language Information Retrieval, Markov Chain, Random Walk

1. INTRODUCTION

Cross-Language Information Retrieval (CLIR) has attracted a large number of studies, and a variety of methods for query translation have been proposed [1, 5, 10, 34, 16, 30, 31]. Many of these methods rely on dictionaries for query translation due to the simplicity of the methods and the increasing availability of machine readable bilingual dictionaries [10, 11, 14, 15, 33]. Some studies have shown that dictionary-based approaches can produce very good CLIR results. However, several problems have also been repeatedly observed in them, and remain unsolved: On the one hand, translation is strongly limited by the coverage of the dictionary, and a manual extension of the dictionary coverage is difficult. On the other hand, even when a dictionary contains all the possible translations for a word, we are still faced with the problem of translation ambiguities. A selection should be made in order to reduce noise (i.e., inappropriate translation candidates). However, dictionaries do not provide any translation reliability measure or context information that can help select the appropriate translations. In most previous studies, dictionaries have been used as the only resource to suggest translation candidates. Although this may result in reasonable suggestions in many cases, it is not sufficient for query translation in CLIR. In fact, unlike other translation tasks such as full text machine translation, a CLIR query can be translated not only by literal translation words (e.g., words that are stored in a dictionary), but also by semantically similar words. These latter have been found to be very useful to produce a desired query expansion effect [16]. For example, a literal Chinese translation of the English term "program" is "程序", but the Chinese term "算法" (algorithm) is semantically related to "program" and is also useful for retrieving more relevant documents about "program".

In order to enhance the expansion effect, several studies have used explicit query expansion before and after translation using pseudo-relevance feedback [1, 20]. However, in all the previous studies, the translation step and the expansion step(s) are performed separately, i.e., they are only loosely connected to the IR model. Many parameters have to be set heuristically. In such a case, it is difficult to determine automatically the best settings of these separate steps so as to maximize their global effectiveness. A better method is to define a single model in which both translation and expansion work together to determine semantically related target words, and to use a principled method to determine the parameters automatically.

In this paper we deal with these problems all together using Markov Chain (MC) models. Both monolingual (e.g. co-occurrences) and cross-lingual (e.g. dictionary translation) term relations are integrated into an MC model which is represented as a directed graph. The "translation" of a query is formulated as a random walk in the MC, where monolingual and cross-lingual term similarities are propagated among terms in both languages. This framework has several advantages: (1) It allows us to integrate both translation relations and monolingual relations such as co-occurrence statistics, by which the suggested terms can be translation terms or related target terms. Thus we are able to overcome the limitation by the coverage of the dictionary and to produce a query expansion effect; (2) The multi-step random walk

CIKM'07, November.6-8, 2007,Lisboa, Portugal

of MC allows us to extend similarity relations from query terms to other indirectly connected similar terms, which further extends the effect of query expansion; (3) The iterative training of MC will result in a stationary probability distribution, which represents better relations between terms than a coarse initial distribution. Truly related target terms are expected to receive higher probabilities after training. (4) There are several methods for automatic tuning of the parameters of MC [21, 29]. Therefore, the MC models provide a solution to all the problems mentioned above.

MC has been used in several recent studies for query expansion [19, 4]. The principle is similar to our work. However, in previous work, the MC was limited to monolingual terms, while we also integrate translation relations. To our knowledge, this study is the first attempt to apply MC to modeling cross-lingual query expansion.

We evaluated our approach on three TREC and NTCIR collections for English-Chinese CLIR. The experiments show that: (1) the use of MC can indeed lead to better translations than with the traditional approaches; (2) The integration of monolingual word relations can bring further improvements.

This paper is organized as follows. Section 2 describes the background of our method. Section 3 presents the MC models for query translation. Section 4 presents the estimation of model parameters. The experiments are presented in Section 5. Section 6 compares our approach with previously proposed methods. Conclusions and future work will be given in Section 7.

2. Background

Traditionally CLIR has been considered as a two–step procedure: query translation by an external component, and monolingual retrieval [10, 14]. Recent studies show that the separation of the two steps does not allow us to take into account effectively the uncertainties in each step, and an integrated approach is preferred [16, 34]. Language modeling has been shown to be an appropriate framework for such integration [16]. In this paper we follow the same principle, and consider query translation as a step embedded in the construction of the final query model in a language modeling setting. We use negative KL-divergence as the basic document ranking function [19], defined as follows:

$$score(q,d) = \sum_{w\in q} P(w|\theta_q) log \frac{P(w|\theta_d)}{P(w|\theta_q)} \propto \sum_{w\in q} P(w|\theta_q) log P(w|\theta_d) \quad (1)$$

where q and d are query and document respectively, and θ_q and θ_d are respectively the parameters of query and document models. By integrating query translation, the above equation is extended to the following one:

$$score(q,d) = \sum_{c\in V} P(c|\theta_q) log P(c|\theta_d) = \sum_{c\in V}\sum_{e\in q} P(c,e|\theta_q) log P(c|\theta_d) = \sum_{c\in V}\sum_{e\in q} P(c|e)P(e|\theta_q) log P(c|\theta_d) \quad (2)$$

where *c* is a term in document language (Chinese) and *e* a term in query language (English).

Equation (2) defines a general language modeling framework for CLIR. The key problem is the estimation of the translation probability $P(c|e)$. It is this estimation that makes our approach different from the others.

Due to the lack of the measurement of translation reliability in a dictionary, most previous studies based on dictionary used two naïve methods:

(1) $P(c|e)$ is assigned uniformly over all the candidates stored in the dictionary;

(2) $P(c|e)$ = 1 if *c* is the first translation of *e* in the dictionary and 0 for all other translations.

In some more recent studies, $P(c|e)$ is determined according to more sophisticated criteria such as the coherence between translation candidates [1, 10, 9]. However, as we mentioned earlier, in all the dictionary-based methods, the estimation of $P(c|e)$ is limited to the translation candidates stored in the dictionary. In order to produce an effect of query expansion, we argue that $P(c|e)$ should not be merely the literal translation probability of *c* given *e*, but a cross-lingual semantic similarity between *c* and *e*.

If $P(c|e)$ is estimated by a statistical translation model, such as one of IBM models [3], trained on a parallel corpus, it reflects cross-lingual term similarities implicitly [16]. However, the reliability for the model to represent such similarities depends on a large degree upon the quality and size of the parallel corpus. Two terms would not be considered as similar terms if they never appear in any parallel sentence pair. Nevertheless, the terms that often co-occur with a literal translation word in parallel texts will receive a small translation probability of the source word. Therefore, a statistical translation model has a capability of producing query expansion effect during translation, by distributing a part of the translation probability to the words that co-occur with the true translation(s).

However, parallel corpora are not widely available for many language pairs (e.g. Chinese-English). Although it is possible to mine parallel materials on the Web for some language [16], dictionaries still remain the most available resources for most language pairs. Therefore, we will use dictionaries in our study.

Notice that the ability of connecting related words in a translation model is a side effect rather than the desired goal of a statistical translation model – the translation model aims to capture literal translation relations. Its training process tries to limit the possibility of connecting related non-translation words rather than to favor it. This is contrary to our goal of query "translation" in CLIR, in which we would like to favor the connections to related non-translation terms as well. Therefore, it is desirable to include related words into query "translation".

Pre- and post-translation query expansions have been exploited as a means to perform such an extension [1, 11]. In pre-translation expansion, the original query is first expanded using a set of feedback documents retrieved in the source language. The expanded query is then translated (e.g. with the help of a dictionary). In post-translation expansion, the translation of the query is used to retrieve a set of feedback documents, which are then used to expand the translated query. In previous studies, both expansion processes have shown some effect on the retrieval effectiveness. However, the expansion steps have been considered to be separated from the retrieval process. They have been used as a means to produce a more appropriate "translation" of the initial query. In these steps, we have to set several parameters manually: the number of feedback documents to be used, the number of terms to be added into the query (or translation), and the weights

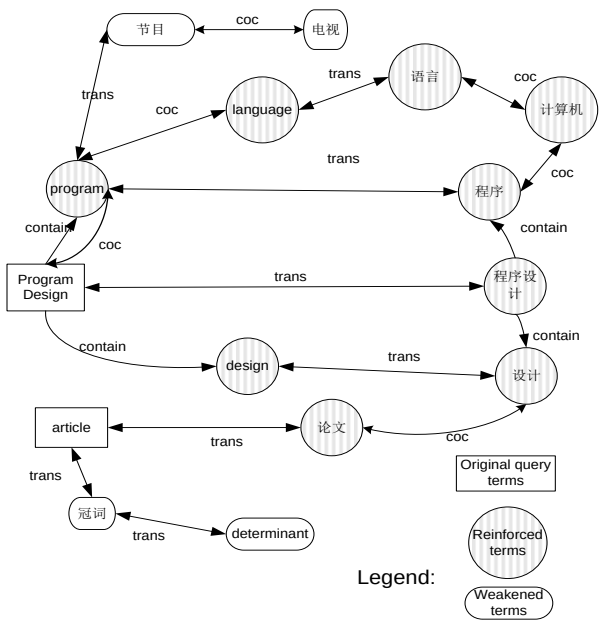

Figure 1 : Illustration of Query Translation via Random Walk

to be attributed to the additional terms (with respect to the original terms).

In addition to the above practical problems, pre- and post-translation expansion can only consider part of the term relations. As shown in [32], both global and local analyses can suggest useful terms to expand queries. Using pre- and post-translation expansions, we are indeed using a local analysis, which can suggest related terms appearing in the feedback documents, either in the source or the target language. As shown in [32], it would be beneficial to add global analysis in the expansion step. Following [32], the global analysis could be used as yet another external component outside the retrieval model. However, as we stated earlier, such a combination is highly dependent on the manually setting of parameters. An alternative is to integrate the term relations extracted from global analysis directly into the model, so that their parameters can be optimized together with those of the translation relations. This means that we extend the methodology of statistical model training to further extending the function $P(c|e)$ from term translation to term similarity relations. To achieve this goal, in this paper we propose to integrate explicitly different types of terms relation into a MC model.

The utilization of MC for query translation in CLIR is not new. [22] used MC to determine the best translation terms. However, only translation relations stored in a dictionary are modeled by the MC. In our case, we integrate other types of term relation in addition to translation relations.

In the following section we will describe the details of the model.

3. Query Translation as a Random Walk

3.1 Principle

Instead of considering query translation as a traditional translation process, now we view it as a process of finding cross-lingual, semantically similar terms. The latter terms can be not only translation terms, but also semantically related terms. Similarly to the principle of pre- and post-translation expansion, related terms can be determined in two ways: they can be target language terms that are related to some translation terms (similar to post-translation expansion), or they can be terms that are translations of related terms in the source language (similar to pre-translation expansion). For example (see Figure 1), given an English (source language) query term "program", besides its literal translation "程序" in Chinese, the Chinese word "计算机" (computer) related to 程序" is also a useful Chinese query term. Similarly, the translation "语言" (language) of a related English term "language" can also be added.

The MC model that we propose tries to integrate the above relations within the source and target languages with translation relations. Our model follows the same principle as pre- and post-translation expansion; but we implement the idea in a very different way. Indeed, we try to determine the related terms in the source and the target languages using a global analysis, i.e. we make use of a global analysis of the whole document corpora, instead of relying on feedback documents which can only be determined on the fly during the retrieval process. As [32] showed, it is beneficial to combine global and local term relations. Therefore, even when global term relations are integrated with translation relations in our MC model, it is still possible to use blind feedback to perform local analysis, similarly to pre- and post-translation expansion.

Another major difference between the previous approaches using pre- and post-translation expansions and ours is the integration of the expansion and retrieval processes. In our case, both processes are integrated within the same framework, making it possible to optimize their parameters together.

An additional advantage of using MC is that, given a query, the word relations (either within one language or between two languages) that are strongly related to the query will be reinforced by each other. The final probability distribution after the iterative adaptation of MC is expected to be better for the query than the initial distribution. For example, suppose that the original English query is "articles about program design". A part of the MC is shown in Figure 1. The two key terms in this query "article" and "program design" can be respectively translated by the following words in Chinese:

article: 冠词 (determinant), 论文 (paper), 物品 (object), etc.

program design: 程序设计

In Figure 1, we also show some term relations within the same language (co-occurrence – *coc* and *contain*, see next section). We can see that through monolingual term relations, the correct translation candidates 论文 (paper) for the ambiguous word "article" is more tightly connected to the original query terms. Through the iterative updating, this term will be assigned a higher probability than the other irrelevant translation candidates. On the other hand, the probability of the words which are less related to the original query, such as "节目" ([TV] program), "电视" (TV), "冠词" (determinant) and "determinant", is reduced.

The above example shows that MC also offers a possible solution to the translation ambiguity problem. Indeed, the principle of mutual reinforcement during random walk is used (although to a very limited degree) in some previous approaches to query expansion. For example, [28] proposed to determine the expansion terms not according to the strength of their relation with one of the original query terms, but according to their

relations to all the query terms. An expansion term having relation with several original query terms will likely be preferred to another one related to only one query term (assuming that their strengths are similar). This approach has proven to be effective.

Transferring the same principle to CLIR, we want to favor translation candidates that are related to more original query terms. In Figure 1, we can see that the translation candidate 论文 (paper) is related to both original query terms (via direct or indirect links). Therefore, its probability is higher than another candidate, "冠词", which is related to only one of the original query terms. This preference is, however, not imposed by using heuristics. Rather, the updating process of MC [2] can naturally reinforce the more related translation candidates. This is another major advantage of using MC as our model.

3.2 Representing Word Relationships with a MC Model

In this section we describe the principle of modeling term similarity in a MC. Each MC model defines a set of states. A state is linked to other states by transitions with different probabilities. Two states are transitional if and only if the transition probability between them is non-zero. A MC model is usually represented as a weighted directed graph *G* as illustrated in Figure 1. It consists of a set of nodes and a set of weighted, directed edges. We use the following notations:

1. A node is denoted as v. We use nodes to represent terms.
2. An edge from v_i to v_j with a label (or relation type) l represents a transition of type l, denoted as: $v_i \xrightarrow{l} v_j$. Each type of edge corresponds to a type of term relation, which will be described later in this section.
3. Each edge $v_i \xrightarrow{l} v_j$ is also assigned a probability $\mathrm{P}(v_j|v_i, l)$. This probability will be determined according to different criteria described in Section 4.1.

If only translation relations are represented in a MC, the MC model can only assign a translation probability to the translation candidates stored in the dictionary. To extend translation to broader cross-lingual similarity relations, we incorporate two additional monolingual relations: *co-occurrence* and *contain*. The former connects frequently co-occurring terms. It has been shown to be useful for CLIR [1, 10]. The latter considers the relation between a longer term (e.g. "*program design"*) and a shorter constituent term (e.g. *"program"*). This relationship is particular useful for Chinese, which does not have any space between words. Therefore variable word segmentation can be produced for the same character sequence. For example, the sequence "程序设计" (program design) can be segmented either as a single word (in fact a phrase) or as two shorter words "程序" (program) and "设计" (design), depending on circumstances and segmentation programs. If "program design" is translated to "程序设计", it matches directly neither "程序" nor "设计" (for the latter will be considered as different indexes). By considering the *contain* relation, we can link "程序设计", and thus "program design", to "程序" and "设计". This is a way to propagate the translation relation to the constituent terms in the target language.

More types of relation can be integrated in this framework, but we limit our investigation in this paper to the three relations: *translation*, *co-occurrence* and *contain*. We will denote them by *trans*, *coc* and *contain*, respectively. The *trans* relations are defined between terms in different languages, the *coc* and *contain* relations between terms of the same language.

Given an MC model, random walk is a process that adjusts the transition probabilities iteratively as follows. In each iteration, we assume a 2-step process of moving from a node to another. First, from a node v_i one can select an edge (i.e., relation) l with probability $\mathrm{P}(l|v_i)$. We assume here that this selection is independent of v_i, so $\mathrm{P}(l|v_i) = P(l)$. Second, v_j is chosen by $\mathrm{P}(v_j|l, v_i)$. Considering a set *L* of all possible edge labels (i.e., relations), the probability to arrive at v_j from v_i is

$$\mathrm{P}(v_j|v_i) = \sum_{l \in L} \mathrm{P}(v_j|l, v_i) P(l) \quad (3)$$

with $\sum_{l \in L} P(l) = 1$.

For example, there are two relations between the terms "program design" and "program" in Figure 1: *coc* and *contain*. The similarity between them is then determined by:

$$\mathrm{P}(\text{program}|\text{program design})$$
$$= \mathrm{P}(\text{program}|\text{contain}, \text{program design}) \times P(contain)$$
$$+ \mathrm{P}(\text{program}|\text{coc}, \text{program design}) \times P(coc)$$

The estimation of $\mathrm{P}(v_j|l, v_i)$ and $P(l)$ will be described in Section 4.

3.3 Random Walk for Query Translation

This section describes how query translation is performed as a random walk in an MC model.

The query translation process can be stated as follows. Let θ_q^0 denote the distribution of an original English query, i.e., θ_q^0 gives non-zero probabilities to the nodes corresponding to the English query terms (words or phrases). The translation process corresponds to the propagation of these probabilities, through random walks, to other terms, especially in the target language. First, a term v_0 is chosen according to the initial distribution θ_q^0. Then one can decide whether to stay in this state (with probability γ) or to transit to another state (with probability 1- γ). The first choice retains the part γ of the probability in v_0, while the second choice transfers (1- γ) of its probability to the related nodes, according to the transition probability (similarity) to them. The process continues in this manner. After k steps of walk, we get a new probability distribution θ_q^k on terms. This latter can be interpreted as the measure of similarity of terms to the original query terms. In particular, the probabilities assigned to target language words are the cross-lingual similarities to the original query.

We notice that the above process interprets the procedure to construct a query-oriented MC model instead of a global MC considering all terms. We call the nodes having non-zero probability in θ_q^k active nodes. Each active node must either be a directly similar node to at least one node corresponding to a term in the original query, or be linked to a node in the query via intermediate nodes. The query-oriented MC model has at least two advantages comparing with the global MC model constructed independently from the query. First, it is much easier to manage and faster to update because the number of active nodes is much smaller than the number of all nodes (the sum of the number of English and Chinese terms). The time required to perform a random walk is thus much shorter. In our experiments, it only

takes a few seconds to update the probabilities and to translate one query. Second, the query-oriented MC model can reduce noise to some degree because it only considers the nodes related to the query so that it avoids distributing probabilities to non-related nodes.

More specifically, let M_{ij} be the probability of being at node v_j at step *t+1* given that one is at v_i at step *t* in the walk, we have:

$$M_{ij} = \begin{cases} (1-\gamma)\sum_{l \in L} \mathrm{P}(v_j|\mathrm{l}, v_i)P(l) & i \neq j \\ \gamma & i = j \end{cases} \quad (4)$$

where $\mathrm{L} = \{trans, coc, contain\}$ is the set of relationships. If we take *k-step* random walk, the similarity between terms is denoted by $M^{(k)}$, then we have:

$$M_{ij}^{(k)} = \left(\gamma \sum_{t=0}^{k} (1-\gamma) M^t\right)_{ij} \quad (5)$$

where *M* is the matrix consisting of M_{ij}. If we set *k* to be infinite, the MC will reach a stationary distribution, which is considered to be optimal [21, 29, 19]. Since $0 < \gamma < 1$, $M^{(k)}$ is guaranteed to converge. In our experiments, we only consider at most 4 iterations, as the convergence is very fast.

Assume that θ_q^0 is the initial probability distribution of the nodes, then the distribution after a *k-step* walk is proportional to

$$\theta_q^k = \theta_q^0 M^{(k)} \quad (6)$$

The document ranking formula for CLIR, i.e., Equation (2), can be re-written as:

$$score(q,d) = \sum_{c \in V} P(c|\theta_q^k) log P(c|\theta_d) \quad (7)$$

where θ_q^k is given by Equation (6) and θ_q^0 is the original parameter setting of query model, i.e., θ_q, in Equation (2).

Now, let us illustrate the mutual influence between similar terms during the random walk. Given the MC model in Figure 1, we assume the initial query terms to be "program design" and "article". The literal translations of these terms are "程序设计" (program design) and "论文" (thesis). Other words enclosed in circles are related terms (through mono-lingual relations). In the figure, we see that "article" has two translations "冠词" (article - in linguistics sense) and "论文" (thesis/paper). As "冠词" (article) does not have any other similar terms except "article", its probability will stay low during the random walk. On the other hand, the probabilities of "程序设计", "论文", "程序" and "设计" will be increased, because they will receive probabilities transmitted from related terms. These terms are strongly related to the query, so the effect is desired. This example shows that MC models naturally integrate query expansion and translation.

4. Parameter Estimation

In this section we describe in detail how we estimate the parameters of the MC models. We have seven parameters to estimate, i.e., three probabilistic models $P(v_j \mid v_i, l)$, with $l \in \{trans, coc, contain\}$, each for one of the three types of relationship, the three corresponding type selection probabilities $P(l)$, and the stopping rate γ. In section 4.1, we will describe how to estimate the three probabilistic models, and then in Section 4.2, we use line search algorithm to estimate the other parameters.

4.1 Probabilities of Relationships

In this study, we use a bilingual dictionary as the translation resource. The probability $\mathrm{P}(v_j|v_i, trans)$, i.e., the translation probability between two terms can be estimated in several ways given the bilingual dictionary:

(1) Uniform distribution: we assign equal probabilities to all candidates, that is:

$$\mathrm{P}(v_j|v_i, trans) = \frac{1}{|\{v_k | v_k \text{ is a translation of } v_i \text{ in the dictionary}\}|} \quad (8)$$

where $|\{.\}|$ is the number of unique elements in a set. This is one of the simple methods used in previous studies, but it may introduce much noise.

(2) Assignment by translation model (GIZA++): a bilingual dictionary can be treated as a parallel corpus: Each English word (or phrase) is aligned to the set of its translations, which is considered as a sentence. We thus can train a statistical translation model using tools such as GIZA++ [24]. We only trained IBM model 1 [3]. This method tries to determine translation probabilities so as to maximize the likelihood of the given sentence alignments. A translation that is appears in more aligned "sentences" will be assigned a higher probability than the one that appears in less aligned "sentences". Thus the probability indirectly reflects how often a translation is frequently used between two languages. It is usually more reasonable than the uniform assignment.

The estimation of *contain* relation is similar to the uniform translation model. We count the number of terms v_j which can be a part of the term v_i, and assign the probability uniformly:

$$\mathrm{P}(v_j|v_i, contain) = \frac{1}{|\{v_k | v_k \text{ is a part of } v_i\}|} \quad (9)$$

Monolingual co-occurrence relations can be estimated on large monolingual corpora by counting the number of windows of a fixed size containing the two terms. The English corpora we used are AP88-90 and the Chinese corpora are the document collection that we use for CLIR experiments (see Section 5). For two terms v_i and v_j let $M(v_i, v_j)$ be a measure of closeness of the two terms. Then the relation between v_i and v_j is defined as follows:

$$\mathrm{P}(v_j|v_i, coc) = \frac{\mathrm{M}(v_i, v_j)}{\sum_k \mathrm{M}(v_i, v_k)} \quad (10)$$

$M(v_i, v_j)$ can be any statistical metric measuring the association between the two terms such as relative frequency, mutual information, information gain and log-likelihood ratio [8]. We use log-likelihood ratio because it produced the best results in our experiments. To filter noise, we only keep the 30 strongest co-occurring terms for each term.

4.2 Parameter Tuning

We estimate the probability of selecting each of the three relationships, *i.e.*, $P(l)$ *and* $\mathrm{l} \in \{trans, coc, contain\}$, and the stopping rate γ. For estimating these parameters, various methods can be used, such as Gradient descent-like approaches [7, 29G^H], Boosting algorithm [21], and so on. However, the objective functions used in these methods only are loosely related to the Mean Average Precision (MAP) which is used to measure the effectiveness of IR systems. Here we choose an alternative approach based on line search to optimize the parameters so as to maximize the MAP on training data directly. This approach has been used in [9, 23] and proven to be very effective. Let us denote the four parameters by a vector $\theta = [P(trans), P(coc), P(contain), \gamma]$, and each dimension of the model θ is denoted as $\theta_i, \mathrm{i} = 1,2,3,4$. Given a test collection with relevance judgments for a set of queries, the MAP resulting

Coll	Description	Size (MB)	#Doc	#Qry
TREC5&6	People's Daily (1991-1993) & Xinhua News Agency (1994-1995)	162	164,789	54
TREC9	HongKong Commercial Daily News, HongKong Daily News and Takungpao News	260	127,938	25
NTCIR3	Chinese Times, Central Daily News, China Daily and United Daily News	508	381,681	50

Table 1: Statistical Information of Dataset

from θ is denoted by MAP(θ). The learning approach can thus be formulated as:

$$\theta^* = \text{argmax}_\theta \text{MAP}(\theta) \quad (11)$$

The optimization problem can be cast as the multi-dimensional function optimization algorithm [25, 9]. The procedure works as follows: $\theta_i, i = 1,2,3,4$ are taken as a set of directions. Line search moves along the first direction while keeping the other unchanged, so as to maximize the MAP; then it moves from there along the second direction to maximize the MAP, and so on.

Cycling through the whole set of directions as many times as necessary, until the MAP stops to increase, we obtain the values of the parameters. This method is intuitive and efficient, but it may converge to different local maxima with different start points. Therefore, we perform the procedure multiple times with random start points, and select the parameters that produce the best MAP. $P(l)$ is normalized to become a probability.

5. Experiments

5.1 Experimental Setting

We evaluated the MC models with three benchmark English to Chinese CLIR collections: TREC5&6, TREC9 and NTCIR3. Table 1 shows the statistical information of these collections.

We conducted our experiments using cross-validation: The models evaluated on the TREC9 collection were learned on TREC5&6 datasets; the models evaluated on TREC5-6 collections were trained on the TREC9 dataset; the models evaluated on NTCIR3 were trained on both TREC9 and TREC5&6.

All Chinese documents and the translated queries are segmented using dictionary-based approach. The Chinese dictionary was compiled by UC Berkeley, which contains 137,613 words. When indexing document collections, we used all possible words in the dictionary and all single Chinese characters as indexing units [18]. All English queries are stemmed with Porter stemmer and the stop words are removed. Since we do not have a phrase recognizer, we only recognize phrases stored in our bilingual dictionary. Each query in TREC and NTCIR 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 defined different window sizes to extract term co-occurrences from English and Chinese corpora, respectively. The window width is 8 words for English, and 10 characters for Chinese.

The bilingual dictionary we used is a combination of two human compiled bilingual lexicons, including the LDC English-Chinese dictionary and a bilingual lexicon generated form a parallel corpus. The dictionary contains 123,747 English entries, including 108,799 words and 1,4948 phrases.

We have developed an experimental IR system based on Lemur 4.2 [26]. The main evaluation metric is the Mean Average Precision (MAP). Different from TREC evaluation, NTCIR uses two relevance judgments: *rigid relevance* which only considers highly relevant documents, and *relaxed relevance* which also considers partially relevant documents. We use rigid relevance for our evaluation. T-test is also conducted for significance test.

5.2 Does the MC Model work?

In this section we present comparison results of the MC models with other traditional CLIR models. Tables 2, 3 and 4 show the main results on the three collections using short and long queries. Two variants of the MC models are tested, in which the initial translation probability is respectively the uniform probability and the translation probability generated by applying GIZA++ on the dictionary. To evaluate the effectiveness of the MC model, four baselines are compared:

ML (Monolingual). In this model, the documents are retrieved with the manually translated Chinese query set provided in the collections. Its performance is usually considered as the upper bound of CLIR.

UM (Uniform Model). This model assigns a uniform distribution of translation probability to all the translation candidates stored in the dictionary. When translating an English query, if we encounter an English phrase in the query that exists in the dictionary, then the phrase translations are used; otherwise, translation of single words are used.

FM (First-one Model). The total translation probability is distributed to the first translation candidate. As UM, phrase translation is used in preference to word translation.

The above two methods may be too simplistic to serve as baseline methods. Nevertheless, we include them in the tables. A more reasonable baseline method is the following one:

GizaM (GIZA Model). The translation probabilities in this model are obtained with the GIZA++ toolkit, which extracts a statistical translation model from the bilingual dictionary, considered as a parallel corpus. GizaM model considers the frequency of translation of one word. If a translation appears several times, either as a translation item for the given word alone, or as a part of a translation of a compound term containing the given word, then the translation word will be assigned a higher probability. Some previous studies [12] have exploited the frequency of translation terms in a document collection in order to select the most frequent translation word. The GizaM model exploits a similar principle, by assuming that the more a translation word corresponds to a source word in the dictionary, the more it is a frequent one and thus should be favored. As we can see in Tables 2-4, this model is a reasonable baseline because it results in retrieval effectiveness comparable to most of the previous studies on the same test collections [10, 11, 13, 33, 34].

Once the translation model is trained on the dictionary, we select the top 10 translations for each term for the short queries and top 3 for long queries. These same numbers are selected for the following two MC models.

UM+MC. The queries are translated with MC model. The initial translation probabilities are obtained from UM.

Model	Short Query				Long Query			
	MAP	% of ML	Imp. Over UM	Imp. Over GizaM	MAP	% of ML	Imp. Over UM	**Imp. Over GizaM**
ML	0.3754		-----	-----	0.4929	-----	-----	-----
UM	0.1281	34.12%	-----	-----	0.2708	54.94%	-----	-----
FM	0.1325	35.03%	3.43%	-----	0.2734	55.47%	0.96%	-----
GizaM	0.3414	90.94%	166.5%**	-----	0.4341	88.07%	60.30%**	-----
UM+MC	0.2918	77.73%	127.8%**	-17.45%	0.4463	90.55%	64.80%**	2.81%
GizaM+MC	0.3720	99.09%	190.3%**	8.96%*	0.4594	93.30%	69.64%**	5.82%

Table 2: Compare Different Model for TREC5&6 Collection

Model	Short Query				Long Query			
	MAP	% of ML	Imp. Over UM	Imp. Over GizaM	MAP	% of ML	Imp. Over UM	Imp. Over GizaM
ML	0.2819	-----	-----	-----	0.2961	-----	-----	-----
UM	0.0976	34.62%	-----	-----	0.1110	37.49%	-----	-----
FM	0.1220	43.28%	24.99%	-----	0.1354	45.73%	21.98%	-----
GizaM	0.2542	90.17%	160.5%**	-----	0.2693	90.95%	142.6%**	-----
UM+MC	0.2750	97.55%	181.7%**	8.18%	0.2622	88.55%	136.2%**	-2.63%
GizaM+MC	0.2897	102.77%	196.8%**	13.97%*	0.2730	92.20%	145.9%**	13.74%*

Table 3: Compare Different Model for TREC9 Collection

Model	Short Query				Long Query			
	MAP	% of ML	Imp. Over UM	Imp. Over GizaM	MAP	% of ML	Imp. Over UM	Imp. Over GizaM
ML	0.2222	-----	-----	-----	0.2840	-----	-----	-----
UM	0.0626	28.17%	-----	-----	0.1212	42.68%	-----	-----
FM	0.0611	27.50%	-2.40%	-----	0.1460	51.41%	20.46%	-----
GizaM	0.1422	63.99%	127.1%**	-----	0.1800	63.38%	48.51%**	-----
UM+MC	0.1442	64.90%	130.3%**	1.41%	0.1987	69.96%	63.94%**	10.38%
GizaM+MC	0.1489	67.01%	137.8%**	4.71%	0.2130	75%	75.74%**	18.33%*

Table 4: Compare Different Model for NTCIR3 Collection

GizaM+MC. This model is similar to UM+MC, but the initial translation probabilities are obtained from GizaM.

From Tables 2, 3 and 4, we find that UM performed the worst among all the methods. This is because it treated all translation candidates of a query term equivalently and introduced much noise (irrelevant translation terms). FM performed slightly better than UM in almost all runs except for short queries of NTCIR3. The reason is that FM only selects the first candidate, which can avoid including noise translation candidates to some degree. However, this "aggressive" selection can also remove relevant translation terms. GizaM can assign a translation probability between two terms according to how often one appears as a translation of another. The translation probabilities have been trained using the EM algorithm [6] to maximize the likelihood of translating each English term by its Chinese translations (the parallel sentence in the dictionary). The advantage of GizaM is that it can assign a strong probability to a translation term if the latter is a specific and unambiguous translation term of the former. However, in GizaM, the whole translation probability is still distributed only to the translations stored in the dictionary. In the above tables, we can see that GizaM performed fairly well. Its effectiveness is around 90% of that of ML in four runs (both short and long queries) of TREC5&6 and TREC9.

For MC models, we observe that the two MC variants are all promising: UM+MC model outperformed UM significantly in all the six runs. GizaM+MC outperformed GizaM in all runs, and it even outperforms the ML for short queries of TREC9. This result confirms the advantages of our MC approach. To see better where the superior effectiveness comes from, let us analyze the example shown in figure 2 for the query "forest railway in Mount Ali".

In this example, *mount* is translated by FM incorrectly as a verb. For the UM model, we only list the translations of "forest" and we can observe that many translations are unrelated to the query. GizaM seems to be able to distribute strong probabilities to related

English query: *Forest Railway in Mount Ali*

ML: 阿里 山 森林 火车

FM: 森林 (forest) 0.5; 林 (woods) 0.5;路线 (road) 0.5; 轨道 (rail) 0.5; 登上 (go up) 0.5 ….

UM: 森林 (forest) 0.5; 造林 (plant trees) 0.5; 山林 (forest in the mountain) 0.5; 植树 (plant trees) 0.5 ; 野 (wild) 0.5...

GizaM: 林 (woods) 0.657819;森 (forest)0.161144; 铁 (iron) 0.455182;铁路 (railway) 0.295346; 装 (set up) 0.395038; 安 (install) 0.222885; 阿里(Ali) 0.293588...

UM+MC: 铁路 (railway) 0.0565199; 森林 (forest) 0.0528217; 经铁路 (via railway) 0.0508112; 铁道 (railway) 0.0508112;阿里 (Ali)0.049161; 蒸汽 (steam) 0.0241262...

GizaM+MC: 林 (woods) 0.10024;铁路 (railway) 0.0651897; 阿里 (Ali) 0.0606648; 森 (forest) 0.0403337;森林 (forest)0.0367658; 嘉义 (Jiayi) 0.021745...

Figure 2: Translation obtained by Each Models for One Query

translation terms. Compared with UM and GizaM, the probabilities assigned by MC models seem generally more appropriate. In addition, they can also suggest some non-translation but related words such as 蒸汽 (steam) and 嘉义 (Jiayi) which is a city connecting Mount Ali. The example confirms the two advantages of MC that we expected:

1. The integration of more term relations can extend translation to broader similar terms, thus producing larger query expansion effect;
2. The iterative probability adjustment process can produce a better probability distribution.

The above results are produced with a random walk of 4 steps for UM+MC and 2 steps for GizaM+MC. We observed that the performance was improved when increasing the steps. This indicates that iterative adjusting similarities between terms are useful for retrieval. We also observed that UM+MC outperformed GizaM in four runs (i.e., long query of TREC5&6, short query of TREC9 and two runs of NTCIR3) and achieved comparable results with UM+MC in the other two runs. This shows that MC models can capture the same characteristics as GizaM. Indeed, both models work with similar principles: They use an iterative learning procedure to assign a high probability to strong translation candidates. Therefore, UM+MC and GizaM performed similarly. However, GizaM+MC can further improve the performance of GizaM in most of the cases. The difference between them is directly attributed to the addition of more relations in GizaM+MC.

5.3 The impact of Different Relationships

In this section we investigate the impact of different relationships on the retrieval effectiveness. Tables 5 and 6 show the results of MC models with uniform translation probability (UM+MC) on the three collections. In the tables, UM is the uniform model mentioned in section 5.2; T represents the MC model only using translation relation; T+C represents model using translation relation plus co-occurrence relation; T+Con represents the model using both translation relation and contain relation; T+C+Con represents the model using all three relations. The T model is indeed equivalent to the one used in [22]. The tables show that the T model outperforms UM substantially. This can be explained by two reasons. First, after propagating the similarity via a random walk, the translation distribution in the T model is changed from uniform distribution to the one which assigns a higher probability to a term if it is connected to many query terms. Second, we only select the top *m* terms as query translation. This helps filter out some noise (which typically has a low probability).

Relation	TREC5&6		TREC9		NTCIR3	
	MAP	Imp. Over T	MAP	Imp. Over T	MAP	Imp. Over T
UM	0.1281	-----	0.0976	-----	0.0626	-----
T	0.2761	-----	0.2616	-----	0.1257	-----
T+C	0.2902	5.10%*	0.2719	0.11%	0.1431	13.84%
T+Con	0.2829	2.46%	0.2746	1.10%	0.1267	7.95%
T+C+Con	0.2918	5.68%*	0.2750	1.25%	0.1442	14.71%

Table 5: Different Relation Combinations for short queries

Relation	TREC5&6		TREC9		NTCIR3	
	MAP	Imp. Over T	MAP	Imp. Over T	MAP	Imp. Over T
UM	0.2708	-----	0.1110	-----	0.1212	-----
T	0.4372	-----	0.2431	-----	0.1904	-----
T+C	0.4458	1.97%	0.2618	7.69%	0.1927	1.21%
T+Con	0.4391	0.43%	0.2578	6.05%	0.1987	4.36%
T+C+Con	0.4463	2.08%	0.2622	7.86%	0.1987	4.36%

Table 6: Different Relation Combinations for long queries

Indeed, as we mentioned earlier, UM is too simplistic to serve as a baseline method. However, T is a reasonable baseline method, which corresponds to the state of the art [22].

We observed that when more relationships are added into the MC model, the effectiveness is further improved. The best model is the one that uses all the three relationships.

On the other MC model, GizaM+MC, we have observed a similar behavior. Due to the space limitation, we do not present the details here.

The experimental results confirm our hypotheses: 1) Integrating more term relations than translation can improve query translation in CLIR; 2) Using an iterative random walk process in MC leads to a more reasonable probability distribution.

6. Related Work

The MC model we used here integrates both query translation and query expansion in a unified framework. Query expansion has been investigated in the context of CLIR in a number of previous studies. Ballesteros and Croft [1] explored query expansion methods for CLIR by combining pre- and post-translation expansion, and they found that the method can effectively improve retrieval effectiveness. McNamee and Mayfield conducted a series of experiments to compare CLIR query expansion techniques [20]. They also found similar results to [1]. The pre- and post-translation expansions are conceptually similar to our addition of more term relations. Thus our experiments confirm their observation.

However, our work is different from the above two in the following aspects:

1). Pre- and post-translation expansions have been separated from translation. In fact, as illustrated in [20], their models are divided into three phases (pre-translation expansion, query translation, and post-translation expansion), that have been handled independently.

In contrast, our MC model incorporates the three phases together within the same framework.

2). Comparing to the expansion process, our MC model-based approach is theoretically more sound, and easier to extend. We also used a principled way to optimize all parameters.

MC models have been used for many other tasks. [21] used the random walk model to disambiguate person's names in e-mails, but the relationships in their model are binary. In IR, infinite random walks have been used for document or webpage re-ranking [17, 27]. The idea of representing semantic similarities by a graph has also been used in NLP and IR. [19, 4] used a random walk model for monolingual query expansion. But they only use one type of relationship. [29] presented a MC model for pp-attachment disambiguation. [22] used MC for query translation in CLIR. However, the MC is built on a dictionary, so translation suggestions are bounded by the dictionary. In our case, we extended the translation relations to cross-language semantic similarity relations. In so doing, we can create more effect of query expansion.

7. Conclusion

Dictionary-based approaches are widely used for CLIR because of their simplicity and the availability of machine-readable dictionaries. However, we are faced with several problems: limited coverage and lack of a measurement for the reliability of the translation candidates. In this paper we proposed a method based on MC models, which integrate several types of monolingual term relation, in addition to the translation relation. As a result, query translation is extended to cross-lingual query expansion.

The MC models also adjust the probabilities of terms automatically through a random walk. We showed in our experiments that the final distribution produces higher retrieval effectiveness than the original one. This shows that the random walk can effectively adjust terms' cross-lingual similarity to the query so that strongly related target terms are assigned higher probabilities.

In this paper we only investigate three types of relation: translation, co-occurrence and containment. However, the method can be easily extended to include more types of relations. Among other useful relations are synonymy, hyponymy and hypernymy.

A possible way of improving our approach is to consider dependency between terms. In our current model the resulting translation candidates are considered independently once they have been generated. In fact, other criteria, such as the coherence between the candidates, can also be useful to help select better candidates [10]. We leave it to future work to integrate these criteria into MC models. Currently, the estimation of transition probabilities is made according to the whole collection. It might be more reasonable to estimate them using local contexts related to a given query. This leads to a query-dependent MC model – another area of our future work.

REFERENCES

[1] Ballesteros, L. and Croft, W.B. (1998). Resolving ambiguity for cross-language retrieval. In *Proceedings of ACM SIGIR.* pp. 64-71.

[2] Brémaud, P. (1999) *Markov chains: Gibbs fields, Monte Carlo simulations, and queues.* Springer-Verlag.

[3] Brown, P., Della Pietra, S., Della Pietra, V., and Mercer, R. (1993). The Mathematics of Statistical Machine Translation. *Computational Linguistics*, 19(2): 243-311.

[4] Collins-Thompson, K, and Callan, J. (2005). Query Expansion Using Random Walk Models. *In Proceedings of CIKM*, pp.704-711.

[5] Davis, M.W., and Ogden, W.C. (1997). Free resources and advanced alignment for cross-language text retrieval. In *the Proceedings of TREC6.* NIST, Gaithersburg, MD.

[6] Dempster, A., Laird, N., and Rubin, D. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, 39: 1-38.

[7] Diligenti, M., Gori, M., and Maggini, M. (2005). Learning web page scores by error back-propagation. In *the Proceedings of IJCAI*. pp. 684-689.

[8] Dunning, T. 1993. Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics.* 19: 61-74.

[9] Gao, J., Qi, H., Xia, X., and Nie, J.-Y. (2005). Linear discriminative model for information retrieval. *In Proceedings of ACM SIGIR*, pp. 290-297

[10] Gao, J.F., Nie, J.Y. (2006). A Study of Statistical Models for Query Translation: Find a Good Unit of Translation. In *Proceedings of ACM SIGIR*, pp. 194-201

[11] Gao, J.F., Nie, J.Y., Xun, E.D., Zhang, J., Zhou, M., and Huang, C.N. (2001). Improving query translation for cross language information retrieval using statistical models. In *Proceedings of ACM SIGIR*, pp. 96-104.

[12] Grefenstette, G. (1999). The World Wide Web as a resource for example-based machine translation tasks, In *Proc. ASLIB translating and the computer 21 conference*.

[13] He, H.Z, Gao, J.F. (2001). NTCIR-3 CLIR Experiments at MSRA In *the Proceedings of NTCIR3*.

[14] Hedlund, T., Airio, E., Keskustalo, H. Pirkola, A., Jarvelin, K. (2004) Dictionary-based Cross Language Information Retrieval: Learning Experiences from CLEF 2000-2002. *Information Retrieval*, 7: 99-119.

[15] Hull, D. and Grefenstette, G. (1996). Querying across languages: A dictionary-based approach to multilingual information retrieval. In *Proceedings of ACM SIGIR*, pp.49-57.

[16] Kraaij, W., Nie, J.Y., and Simard, M. (2003). Embedding Web-Based Statistical Translation Models in Cross-Language Information Retrieval. *Computational Linguistics*, 29(3): 381-420.

[17] Kurland, O., and Lee, L. (2005). Pagerank without hyperlinks: Structural re-ranking using links induced by language models. In *Proceedings of ACM SIGIR*. pp. 306-313

[18] Kwok, K.L. (2000). Exploiting a Chinese-English bilingual wordlist for English-Chinese cross language information retrieval. In *the Proceedings of the Fifth International Workshop on Information Retrieval with Asian Languages, IRAL-2000*. pp. 173-179.

[19] Lafferty, J. and Zhai, C. (2001). Document language models, query models, and risk minimization for information retrieval. In *Proceedings of ACM SIGIR* , pp. 111-119.

[20] McNamee, P. and Mayfield, J. (2002). Comparing Cross-Language Query Expansion Techniques by Degrading Translation Resources. *In the Proceedings of ACM SIGIR*, pp. 159-166.

[21] Minkov, E., Cohen, W., and Ng, A. (2006). A Graphical Framework for Contextual Search and Name Disambiguation in Email. In *the Proceedings of ACM SIGIR*, pp. 27-34.

[22] Monz, C., and Dorr, B., (2005). Iterative translation disambiguation for cross-language information retrieval. In *the Proceedings of ACM SIGIR*, pp. 520-527.

[23] Morgan, W, Strohman, T., and Henderson, J. (2004). *Dicrect maximization of average precision by hill-climbing with a comparison to a maximum entropy approach*. Technical report. MITRE.

[24] Och, F., and Ney, H. (2000). Improved statistical alignment models. In *Proceedings of ACL*. pp. 440-447.

[25] Och, F.J. (2003). Minimum error rate training in statistical machine translation. In *Proceedings of ACL*. pp. 160-67

[26] Ogilvie, P. and Callan, J. (2001). Experiments using the lemur toolkit. *In the Proceedings of TREC-10*, pp.103 – 108.

[27] Page, L., Brin, S., Motwani, R., and Winograd, T. (1998). *The PageRank citation ranking: Bringing order to the web*. Technical Report, Computer Science department, Stanford University.

[28] Qiu, Y.G., and Frei, H.P. (1993). Concept query expansion. In *the Proceedings of ACM SIGIR*, pp. 160-169.

[29] Toutanova, K., Manning, C. and Ng, A. (2004). Learning Random Walk Models for Inducing Word Dependency Distributions. *In the Proceedings of the 21st International Machine Learning Conference.*

[30] Wang, J. and Oard, D. (2006). Combining Bidirectional Translation and Synonymy for Cross-Language Information Retrieval. In the *Proceedings of ACM SIGIR.* pp. 202-209.

[31] Xu, J., and Weischedel, R. (2000). Cross-lingual information retrieval using Hidden Markov models. In *the Proceedings of SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*. pp. 95-103.

[32] Xu, J.X., and Croft, B. (1996). Query expansion using local and global document analysis. In *the Proceedings of ACM SIGIR*. pp. 4-11.

[33] Xu, J.X., Weischedel, R. (2005). Empirical studies on the impact of lexical resources on CLIR performance. *Information Processing and Management.* 41: 475-487.

[34] Xu, J.X., Weischedel, R., and Nguyen, C. (2001). Evaluating a Probabilistic Model for Cross-lingual Information Retrieval. In *Proceedings of ACM SIGIR*, pp. 105-110.