Parsimonious Translation Models for Information Retrieval

Seung-Hoon Na
Div. of Electrical and
Computer Engineering
POSTECH, AITrc
nsh1979@postech.ac.kr

In-Su Kang
Div. of Electrical and
Computer Engineering
POSTECH, AITrc
dbaisk@postech.ac.kr

Jong-Hyeok Lee
Div. of Electrical and
Computer Engineering
POSTECH, AITrc
jhlee@postech.ac.kr

ABSTRACT

In the KL divergence framework, the extended language modeling approach has a critical problem estimating a query model, which is the probabilistic model that encodes user's information need. For query expansion in initial retrieval, the translation model had been proposed to involve term co-occurrence statistics. However, the translation model was a difficult to apply it, because term co-occurrence statistics must be constructed in the offline time. Especially in large collection, constructing such a large matrix of term co-occurrences statistics prohibitively increases time and space complexity. More seriously, reliable retrieval performance cannot be guaranteed because the translation model may comprise noisy non-topical terms in documents. To resolve these problems, this paper investigates an effective method to construct co-occurrence statistics and eliminate noisy terms by employing a parsimonious translation model. The parsimonious translation model is a compact version of a translation model that can reduce the number of terms containing non-zero probabilities by eliminating non-topical terms in documents. Through experimentation on seven different test collections, we show that the query model estimated from the parsimonious translation model significantly outperforms not only baseline language modeling but also non-parsimonious models.

Keywords

Information Retrieval, Language Models, Parsimonious Models

1. INTRODUCTION

Recently, the language modeling approach has become a popular IR model based on its sound theoretical basis and good empirical success [1, 3, 4, 5, 6, 7, 8, 9, 10, 11, 15, 17, 18, 19]. In contrast to the probabilistic model [12], it estimates individual document models that use unique probability distribution of words rather than explicitly inferring the relevance information. From the perspective of information retrieval, the language modeling approach provides a very flexible framework to deal with the complex relationship among terms. For example, adjacent two terms in a query has a dependency in the bi-term language model [17], whereas a term in a query can have a syntactic dependency with any terms within a query in the dependency language model [3], and a term in a query have dependency on parent in a sentence tree [10]. The language modeling approach can be incorporated with some advanced techniques such as query expansion [1, 7, 19], and cluster-based retrieval [6, 9], to resolve word mismatch problem.

This paper focuses on improving initial retrieval performance on language modeling framework by using automatic query expansion. Here, the initial retrieval means the retrieval process without using top retrieved documents. The performance of initial retrieval is critical, since post-processing such as pseudo relevance feedback is highly dependent on the initial retrieval performance. To do this, automatic query expansion based on word co-occurrence statistics is one good strategy, which is explored by two approaches in the language modeling approach: the statistical translation model [1] and the Markov chain translation model [7].

The statistical translation model is based on a document-query translation process by employing a translation model in the Machine Translation field [1]. In the traditional viewpoint, the operation of this model is similar to document expansion rather than query expansion because the expansion process is performed not on query terms but on document terms. The statistical translation model has trouble with two problems: requirement of relevance judgment data and time complexity at retrieval. Although the former problem is resolved by the title-language model [5], the latter problem is left unsolved. The problem is resolved in the Markov chain translation model [7], where the query-query translation process is adopted by estimating the query model using the translation model, instead of using the document-query translation process. Because expansion is performed on only query terms, the retrieval process can be done in a feasible time.

However, previous approaches raise problems of computational complexity and retrieval effectiveness for constructing the translation model. For the translation model, the naive constructing method of word co-occurrence statistics requires a prohibitive time-consuming process as well as large space overhead. More seriously, expansion terms can reduce retrieval performance because common words or stop words can be included in word co-occurrence statistics.

To simultaneously resolve these two problems, an additional process is necessary to extract only topical terms in documents and then construct word co-occurrence statistics using them. For extracting topical terms, one can consider traditional term weighting schemes such as TF*IDF. However, since these schemes are not based on the language modeling framework, the method from these schemes is ad-

MF/IR 2005 ACM SIGIR 2005 Workshop on Mathematical/Formal Methods in Information Retrieval

hoc. This paper concentrates on a well-founded framework to extract topical term and apply them in query expansion.

To do this, we adopt a parsimonious language model [4], where topical terms in a document are extracted and weighted at the EM framework. Unlike the MLE document model, the parsimonious document model eliminates the global common portion and leaves the document topic portion. The alternative translation model can be constructed using the parsimonious language model, resulting in the parsimonious translation model which the paper proposes.

Our work is a further exploration of Hiemstra's work. Hiemstra revolves three main processes for using parsimony: indexing time, retrieval time, and feedback time. However, he did not address the usage of it with a translation model to expand an original query model without feedback process. This paper presents empirical evidence that the parsimonious language model can significantly reduce time and space complexity, increasing retrieval performance compared with a non-parsimonious model when it can be used in query expansion based on the translation model.

The rest of the article is organized as follows. In Section 2 we briefly review KL divergence framework of the language modeling approaches and the query model estimation problem, and traditional translation model. In Section 3 we examine problems inherent to the traditional translation model, and that motivate the parsimonious model. In Section 4 and Section 5, we describe the parsimonious translation model and show results of experiments to evaluate it. Section 6 presents conclusions and outlines directions for future work.

2. LANGUAGE MODELING APPROACH TO INFORMATION RETRIEVAL

2.1 Formalization

In the language modeling approach to information retrieval, one considers the probability of a query as being generated by a probabilistic model based on a document [11]. It defines a document language model for each document, and ranks a document by likelihood of query given the document language model. For a query $Q = q_1...q_m$ and document model θ_D, this probability is denoted by $P(Q|\theta_D)$.

$$P(Q|\theta_D) = P(q_1...q_m|\theta_D) \tag{1}$$

Further derivation of Eq. (1) would differ according to assumption of term dependency. As a simple assumption, we consider the case that terms in a query are independently generated, then query-likelihood

$$P(q_1...q_n|\theta_D) = \prod_{i=1}^{m} P(q_i|\theta_D)$$

characterizes term independence.

Lafferty and Zhai suggested KL divergence retrieval framework to incorporate query expansion or relevance feedback by introducing the concept of a query model [7], where the query model is a probabilistic model for user's information need. In KL divergence retrieval framework, the scores of documents are negative KL divergence between query model θ_Q and document models θ_D as in Eq. (2).

$$\begin{aligned} -KL(\theta_Q\|\theta_D) &= \sum_w P(w|\theta_Q)\log\frac{P(w|\theta_Q)}{P(w|\theta_D)} \\ &\propto \sum_w P(w|\theta_Q)\log P(w|\theta_D) + \text{const} \end{aligned} \tag{2}$$

Eq. (2) summarizes the two challenging problems: estimation of document model and estimation of query model.

There have been several studies done on estimation of document language models. Basically, a critical issue in estimating a document language model is smoothing. If we simply use maximum likelihood estimation for the document language model, then the estimated document language model will assign zero probabilities to unseen words in the document. To avoid this data sparseness problem, smoothing techniques are incorporated to add some probabilities to unseen words. The most popular smoothing method in language modeling is Jelinek's smoothing, with interpolation of the MLE (Maximum Likelihood Estimation) of the document language model with a background collection language model $P(w|\theta_C)$ using a coefficient λ to control the influence of the MLE document language model. See [18] for empirical studies of several smoothing methods.

The second problem is query model estimation. If θ_Q is an empirical distribution of the given query sample $Q = q_1...q_m$, then Kullback-Leiber divergence framework is identical with the original language modeling approach. Like the document language model, query model estimation has trouble with the data sparseness problem because most users' query samples are constructed ad-hoc and do not perfectly describe all related query terms. Smoothing for query model is called *semantic smoothing* [1, 7, 8], which is the method to give additional probabilities to semantically or topically related terms. Semantic smoothing corresponds to *automatic query expansion* in traditional methods for resolving word-mismatch problems.

Note that the smoothing query model can be obtained by smoothing document models. For example, the statistical translation model [1] expands terms in documents rather than expanding query terms. Differently from general document smoothing, we call this document smoothing semantic smoothing of document. In terms of traditional viewpoint the semantic smoothing of a document corresponds to the concept of document expansion [14], and in the language modeling approach the smoothing is explored by cluster-based smoothing [9]. Effects of two different smoothing, semantic smoothing of a query and document, may have a similar effect on retrieval performance. However, the semantic smoothing of document has trouble with large computational complexity when the model is applied as mentioned in [7]. Although the complexity will be resolved by constructing document models in advance in the offline stage, the offline construction is not easily handled due to its time complexity.

One estimation method of the query model is based on the relevance language model [8]. Without relevant document information, the relevance language model is approximated by the probability of generating other terms from documents containing query terms.

$$P(w|\theta_R) \approx P(w|q_1...q_m) = \frac{P(w, q_1...q_m)}{P(q_1...q_m)} \tag{3}$$

The relevance language model is effectively estimated by assuming conditional independence between query terms and w. This conditional independence is accepted in only top retrieved documents due to risk of assumption on whole document models. The relevance language model $P(w|\theta_R)$ is considered as a query model, so the relevance model estimated in Eq. (3) is used in the KL divergence framework. Another estimation of the query model is based on the Markov chain translation model (MCTM), of which a random walk process is performed on the Markov chain between words and documents [7]. Basically, MCTM is a variant of a statistical translation model that resolves several inherent limitations of the models. From the perspective of a language modeling framework, MCTM provides several advantages for automatic query expansion. The first reason is that MCTM provides another formulation of traditional measure such as co-occurrence statistics for automatic query expansion. Second, it generalizes restricted query expansion such as pseudo relevance feedback or cluster-dependent query expansion, where they are using top retrieved documents or cluster-based retrieval, respectively. Third, MCTM provides a learning mechanism, since the translation model in MCTM can be learned by using relevance judgment. Fourth, the translation model in MCTM is general, since it can be used in other similar methods such as relevance language model and statistical translation model for resolving word-mismatch problem. For these reasons, our study focuses on MCTM and its translation model.

2.2 Markov Chain Translation Model

2.2.1 Translation Model

Differently from the relevance language model using top retrieved document models, MCTM is estimated by using all document models, allowing query expansion process without initial retrieval result. The stochastic transition matrix T, where states are words, consists of state translation probabilities $T(q|w)$ in (w, q)-entry, which is the probability of generating q in documents containing w.

$$T_{w,q} = T(q|w) = \sum_D P(q|\theta_D)P(\theta_D|w) \tag{4}$$

In the above, the state transition probability refers by word-to-word *translation* probability. A random walk process on this Markov chain is performed which consists of several sequential state transition process. For each state transition, the stop event occurs with the probability of $(1-\alpha)$, and there is no more state transition after the stop event occurs. Elements of the transition matrix after state transition events of k times is

$$T^k(q|w) = (1-\alpha)\alpha^k T^k_{w,q}$$

For simplification and reliability, we only consider the case that the maximum of the random walk step is 1. In this case, the state word translation probability is of the form We have already seen several typeface changes in this sample.

$$t_\alpha(q|w) = \alpha T(q|w) + (1-\alpha)\delta(w,q) \tag{5}$$

where $\delta(w,q)$ is one if w equals to q and zero otherwise. In other words, $t_\alpha(q|w)$ indicates the probability of events, where the random walk process starts at word q and immediately stops, or stops at word q after one step when starting other term w (because all words stop after one random walk, we ignore the stop probability.). We obtain another Markov chain by using state translation probability $t_\alpha(q|w)$, where the single state translation process is performed instead of the random walk process.

2.2.2 Query Model Estimation

Query model estimation based on translation probability is one widely used technique, since the translation model reflects co-occurrence statistics in traditional automatic query expansion.

According to [7], the query model $P(w|\theta_Q)$ is estimated according to probabilities that the random walk process starting at w is stopped at any term of a given query. This intuition is formulated as follows.

$$P(w|\theta^B_Q) \propto \frac{1}{m}\sum_{i=1}^{m} t_\alpha(q_i|w)P(w|\phi)$$

where $P(w|\phi)$ is a prior model that acts to select the useful and frequent words having high IDF. This leads to

$$P(w|\theta^B_Q) \propto \sum_{q\in Q} t_\alpha(q|w)P(w|\phi)P(q|\hat{\theta}_Q) \tag{6}$$

where $\hat{\theta}_Q$ is MLE of given query terms. In this paper, we call θ^B_Q *backward query model*, of which estimation uses translation probability toward backward direction from expansion terms to query terms.

Unlike the method above, we can consider an other query model θ^F_Q , which is estimated by probabilities that the random walk process starting at an arbitrary term of a given query is stopped at term w. This new estimation method is formulated as follows.

$$P(w|\theta^F_Q) \propto \frac{1}{m}\sum_{i=1}^{m} t_\alpha(w|q_i)P(w|\phi)$$

This is rewritten by using MLE of query model.

$$P(w|\theta^F_Q) \propto \sum_{q\in Q} t_\alpha(w|q)P(w|\phi)P(q|\hat{\theta}_Q) \tag{7}$$

We call θ^F_Q *forward query model*, of which estimation uses translation probability toward direction from query terms to expansion terms. To see more clearly the difference between Eq. (6) and Eq. (7), we re-consider two estimation methods by simple modification based on the Markov chain theory. First, we derive the following equation by definition of transition probability.

$$t_\alpha(w|q)P(q) = t_\alpha(q|w)P(w)$$

where $P(w) = \sum_D P(w|\theta_D)P(\theta_D)$. By basic Markov chain theory [13], $t_\alpha(w|q)$ is the transition probabilities of the reversed Markov chain consisting of $t_\alpha(q|w)$ and $P(w)$ are stationary probabilities both on the forward chain and the reversed chain.

Then, the forward and the backward query model are modified as follows (For simplicity, we assume that prior probability $P(w|\phi)$ is uniformly distributed (i.e. $P(w|\phi) = 1/M$))

$$P(w|\theta^B_Q) \propto \sum_{q\in Q} \frac{Co_\alpha(w,q)}{P(w)} P(q|\hat{\theta}_Q)$$

$$P(w|\theta^F_Q) \propto \sum_{q\in Q} \frac{Co_\alpha(w,q)}{P(q)} P(q|\hat{\theta}_Q) \tag{8}$$

where $Co_\alpha(w,q) = t_\alpha(q|w)P(w) = t_\alpha(w|q)P(q)$ indicates co-occurrence information of w and q. In the backward query model expansion terms are weighted by topicality of expansion terms, while in the forward query model expansion terms are weighted on topicality of query terms co-occurred with them. However, the expansion terms of backward query model can have biased weights, since all topical terms co-occurred with any of query terms have high probabilities regardless of their relevancy to query terms. In the forward query model, terms co-occurred with informative query terms will have high probabilities rather than their topicality. Intuitively, the forward query model is more reasonable than the backward query model, since the topicality of expansion term can be reflected with IDF in ranking. In our preliminary experimentations, the forward query model improves the baseline language model, while the backward query model does not improve the baseline method and sometimes showing results in a worse performance. Motivated from this result, we selected the forward query model as our reference method due to its superiority over the backward one. The forward query model is simplified by the assumption of uniform distribution for the prior model.

$$\begin{aligned} P(w|\theta_Q^F) &= \sum_{q\in Q} t_\alpha(w|q)P(q|\hat{\theta}_Q) \\ &= \alpha\left(\sum_{q\in Q} T(w|q)P(q|\hat{\theta}_Q)\right) + (1-\alpha)P(q|\hat{\theta}_Q) \end{aligned}$$

We can regard $\sum_{q\in Q} T(w|q)P(q|\hat{\theta}_Q)$ as the expansion query model. As a result, we see that the final query model is interpolation of the expansion query model and the MLE query model. Interestingly, the relevance language model (i.e. Method 2 of [8]) can be reviewed though the perspective of the translation model, which also implicitly uses the translation model as follows.

$$\begin{aligned} P(w|\theta_R) &\approx P(w|Q) = \frac{P(w, q_1 ... q_m)}{P(q_1 ... q_m)} \\ &= P(w)\prod_i \frac{P(q_i|w)}{P(q_i)} \end{aligned}$$

where $P(q_i|w)$ corresponds to the translation model $t_\alpha(q_i|w)$, which is equal when parameter α is 1.

$$\begin{aligned} & P(w)\prod_i \frac{t_\alpha(q_i|w)}{P(q_i)} \\ &= P(w)\prod_i \frac{t_\alpha(w|q_i)}{P(w)} \end{aligned} \tag{9}$$

In fact, the above reflects an inner log term of MI measure of w and q_i.

$$= P(w)\prod_i \frac{Co_\alpha(w,q_i)}{P(w)P(q_i)}$$

where $Co_\alpha(w,q)/P(w)P(q_i)$ corresponds to inner log term of MI measure.

3. ANALYSIS OF MARKOV CHAIN TRANSLATION MODEL

3.1 Computational Complexity

To apply Eq. (8) at retrieval time, we need to calculate the translation matrix $T(w|q)$ in advance while offline. However, the calculation process is time consuming work as the number of document increases. To see the time complexity, first we rewrite $T(w|q)$ as follows.

$$T(w|q) = \sum_D P(w|\theta_D)P(\theta_D|q) = \sum_D P(w|\theta_D)P(q|\theta_D)/P(q) \tag{10}$$

Translation probability of (q,w) $(T(w|q))$ involves $\sum_D P(w|\theta_D)p(q|\theta_D)$ that corresponds to co-occurrence statistics of $q$and w among whole documents. We consider the following process for constructing co-occurrence statistics for all translation probabilities (i.e. translation matrix).

```
 Given N documents in collection $C$
Initially, set $T(w|v)$ to 0 for all pair $w$, $v$
For each $D \in C$
For all pair $w, v \in D$
Increase $T(w|v)$ by $p(w|\theta_D)p(v|\theta_D)$
```

In the above algorithm, co-occurrence statistics $T(w|v)$ are accumulated as the number of processed documents increase. Let the average number of unique terms in documents be K and the number of documents be N. The expected time complexity (TC) is obtained as follows.

$$TC = N\cdot(K-1)\cdot K/2 = O(K^2N) \tag{11}$$

The important fact is that the time complexity is proportional to the number of documents by linear time. Because recent retrieval environments require a large number of documents, its time complexity should be seriously considered. For a simple example, in the situation of ad-hoc document retrieval, i.e. K is 100, atomic operations of 5,000 (the operation marked with * in the above algorithm box) are required for each document. If N is only 200,000, the total number of atomic operations reaches 1G. Next, we will consider space complexity. Let the number of unique terms of each document D be L_D, and the number of co-occurred documents with w and v be $cdf(w,v)$. A is set of feasible term pairs defined by $\{(w,v)|cdf(w,v) > 0\}$. Then the following equation is easily derived by definition.

$$\sum_D L_D(L_D-1)/2 = \sum_{(w,v)\in A} cdf(w,v) \tag{12}$$

Let the number of unique term pair be R, then R is

$$R = |A| = \sum_{(w,v)\in A} 1$$

From Eq. (13) and Eq. (14), R is rewritten by

$$R = \sum_D L_D(L_D-1)/2 - \sum_{(w,v)\in A} (cdf(w,v)-1))$$

At this step, we denote expectation of $E(cdf(w,v))$ by $Ecdf$.

$$\begin{aligned} R &\approx E\left(\sum_D L_D(L_D-1)/2\right) - R(Ecdf-1) \\ &= NK(K-1)/2 - R(Ecdf-1) \end{aligned}$$

Thus, we derive final estimation of R.

$$R \approx \frac{1}{2\cdot Ecdf}NK(K-1) \tag{13}$$

A fundamental question is how the quantity $Ecdf$ is related to N. Is $Ecdf$ linear function of N or or $\log(N)$? Here, we assume that $Ecdf$ can be ignored given N, i.e. $Ecdf \ll N$. Strictly, this assumption is not true in real situation, since the number of co-occurred documents increases along with the number of documents. Considering the case that w and v are topically related, we expect that $cdf(w,v)$ very slowly increases compared with N. It is related to the probability that a new document belong to the topic dominantly related to w and v. Since the number of hidden topics also increases along with the number of documents, the probability given topic will be very small. Thus, our naive assumption that $Ecdf$ is almost a constant term given N ($Ecdf \ll N$) does not turn away from a real situation. By applying this naive assumption, R becomes $O(NK^2)$. As a result, we find that both the time and space complexity are $O(NK^2)$, thus when N is a very large value, time and space complexities are significantly dependent on the variation of K.

3.2 Retrieval Risk

There is a negative effect on retrieval performance in expansion terms from MCTM. To access the retrieval effectiveness, we first revisit the Eq.(5).

$$-KL(\theta_Q \| \theta_D) \propto \sum_w P(w|\theta_Q)\log P(w|\theta_D)$$
$$\propto \sum_w P(w|\theta_Q)\log\left(\frac{\lambda P(w|\theta_D)}{(1-\lambda)P(w|\theta_C)} + 1.0\right) \quad (14)$$

where λ is Jelinek-Mercer smoothing parameter for the document language model and θ_C is a collection language model. When we estimate θ_Q by using the query language model using MCTM, there are common words with high positive probabilities in θ_Q. The reason is that MCTM uses an MLE document model for θ_D. The MLE document model will assign high probability to common words, and translation probability $T(w|q)$, which corresponds to co-occurrence statistics $\sum_D p(w|\theta_D)p(q|\theta_D)$, enforces the final estimated θ_Q to assign high weights to common words. Intuitively, if the query model contains common terms with high weight, the risk of retrieval will significantly increase. Let us assume that two terms w and v where w is a common word and v is a topically relevant word, and $P(w|\theta_Q) = \kappa P(v|\theta_Q)$ where $\kappa > 1$. Retrieval risk of common word w can be resolved if the retrieval influence of w is less than that of v. Given the document D, let the retrieval influence of w be the increment of log-query likelihood $P(w|\theta_Q)\log P(w|\theta_D)$, and assume the following equation, ignoring the impact of document models.

$$\log\left(\frac{\lambda P(w|\theta_D)}{(1-\lambda)P(w|\theta_C)} + 1.0\right) \approx idf(w) = \log\frac{N}{df(w)} \quad (15)$$

where $df(w)$ means the document frequency.

To obtain a constraint for $df(v)$ to prevent the retrieval risk of w, we further derive as follows.

$$\begin{array}{rcl} P(w|\theta_Q)\log(w|\theta_D) & < & P(v|\theta_Q)\log(v|\theta_D) \\ P(w|\theta_Q)\log(N/df(w)) & < & P(v|\theta_Q)\log(N/df(v)) \\ P(w|\theta_Q)\log(N/df(w)) & < & \kappa P(w|\theta_Q)\log(N/df(v)) \\ \left(\frac{N}{df(w)}\right) & < & \left(\frac{N}{df(v)}\right)^{\kappa} \end{array}$$

The final inequality is rewritten by.

$$df(v) < \left(\frac{df(w)^{\kappa-1}}{N^{\kappa-1}} df(w)\right) \quad (16)$$

$df(v) < df(w)$ is obtained at $\kappa = 1$, where v and w are the same in the query model. In this case, the upper bound of $df(v)$ is feasible from the fact that the document frequency of a topical term is smaller than one common term. However, as $\kappa > 1$, the upper bound for $df(v)$ is not tractable, since the bound is exponentially decreased according to . Specifically, if a query contains stop words or common words, it should be eliminated since their effects on expansion terms are very dangerous. One method for this elimination is to use heuristics such as the determination of automatic common words based IDF. However, such a method is ad-hoc and not unified in single a framework. More importantly, such criterion may be very vague for determining stop words and common words. Even though such method is successful, it cannot avoid risk of non-topical terms for documents due to its MLE document models. Co-occurrence statistics for translation model is reliable only if document represents single topic, because underlining basic of co-occurrence statistics is that it is to determine whether given two terms are topically co-occurred on whole document collection or not. Due to variety of document topics, there may be several different topical terms in document. Thus, co-occurrence evidence between different topical terms may be harmful for constructing accurate translation models. In this regard, the MLE document model itself does not reflect topicality of document terms, thus a more elaborated document model is necessary for estimating a translation model more accurately to assign high probabilities to main topical terms.

4. PARSIMONIOUS TRANSLATION MODEL

4.1 Motivation

In Section 3, we discussed the computational complexity and the retrieval risk of MCTM. To reduce this computational complexity, one possible strategy is to use only feasible terms and not all terms in the document. This strategy aims to decrease factor K. In such a strategy, it is well-known to restrict extraction of term pairs within the local context: small windows such as few words or phrase level or sentence level [15, 16, 17]. However, in most applications (e.g, word sense disambiguation) topical context and local context play different roles. Co-occurrence only from local context cannot completely substitute for co-occurrence from global context, and its effect will be different from the effect of using topical context. Especially in the query expansion problem, topically related terms should be selected and expanded. Co-occurrence statistics on topical context would be a primal resource for our problem, rather than those on local context.

Fortunately, in language modeling approaches, the parsimonious language model [4] provides a devise to handle topical terms in a document. The parsimonious language model enables us to build models that are significantly smaller than standard models within its model. In this model, it is assumed that top highly probable k terms are topical in the document and non-zero probabilities are assigned only to them. Other terms have zero probabilities. By applying a Markov chain on this parsimonious document model, a translation model can be constructed. We called this translation model the *parsimonious translation model*, discriminating it from the original translation model. In other words, parsimonious translation model is translation model estimated using parsimonious document models instead of MLE document models in Eq. (4).

4.2 Parsimonious Document Model

As noted in Section 2, document language models are constructed by mixing the MLE document language model and global collection language model. MLE for document is far from a document specific model because it contains global common words. To construct document specific topic model, we assume that documents are generated from mixture model with the document specific model and global collection model. For given document D, the likelihood of document is as follows.

$$P(D) = \prod_{w \in D} (\lambda P(w|\theta_D) + (1-\lambda)P(w|\theta_C)) \quad (17)$$

where $P(w|\theta_D)$ is document specific topic model for estimation. To maximize the document likelihood, we apply the EM algorithm [2].

E-step:

$$P[w \in D] = \frac{\lambda P(w|\theta_D)^i}{\lambda P(w|\theta_D)^i + (1-\lambda)P(w|\theta_C)}$$

M-step:

$$P(w|\theta_D)^{i+1} = \frac{c(w;D)P[w \in D]}{\sum_{w \in D} c(w;D)P[w \in D]}$$

where $P[w \in D]$ is the probability such that given w is a document specific term and i indicates the number of EM iterations. As iterations increase, the global collection model is not changed and only the document specific topic models are iteratively updated. For simplicity, let us denote $\tilde{\theta}_D$ to convergent document specific topic model.

Next, the selection process is performed, where only highly topical terms are selected, and non-topical terms are discarded. For non-topical terms w, its probability $P(w|\tilde{\theta_D})$ becomes 0. Discarded probability is re-distributed to topical-terms, uniformly. There are two possible techniques to select topical terms. One method is *select_top*(k), where terms are sorted by $P(w|\tilde{\theta}_D)$, and only top k ranked terms are selected (k is about between 50 and 100). Another method is *select_ratio*(P), where top terms are selected as much as summation of probabilities of selected terms is below limit probability P (P is between 0.0 and 0.1). After now, we will further explain the *select_ratio*(P). We will call P the *parsimony level*.

Let us define the parsimonious document language model consisting of topical terms selected by *select_ratio*(P), and Γ_D be a set of topical terms of the document D selected by *select_ratio*(P). Γ_D is a maximal subset of words of document D that satisfies the constraint $\sum_{w \in \Gamma_D} P(w|\tilde{\theta}_D) < P$.

$$P(w|\theta_D^s) = \begin{cases} Z_D P(w|\tilde{\theta}_D) & \text{if } w \in D \\ 0 & \text{otherwise} \end{cases} \quad (18)$$

where Z_D is a normalization factor with value $1/\sum_{w \in \Gamma_D} P(w|\tilde{\theta}_D)$.

4.3 Parsimonious Translation Model

As mentioned in Section 2, translation probability $T(w|q)$ is the probability generating w in the document that includes the given term q. Since the word translation model is a mixture model of different document models, it is one of the document language models. Substituting document language model of Eq. (4) into the summation of the document specific model and global collection model, we further derive a translation model.

where η is a smoothing parameter for mixing document specific model, and collection language model. Conceptually, although η corresponds to the smoothing parameter λ for initial retrieval, we treat η differently to λ. The translation model consists of three summation parts: Document specific co-occurrence model $\sum_w P(w|\theta_D^s)P(q|\theta_D^s)p(\theta_D^s)$, global co-occurrence model $P(w|\theta_C)P(q|\theta_C)$, and term topicality $P(w|\theta_D^s)P(\theta_D^s)$. PTM $t^s(w|q)$ is defined as model which divides the document specific co-occurrence model by global likelihood $P(q)$.

$$T^s(w|q) = \frac{1}{P(q)} \sum_D P(w|\theta_D^s)P(q|\theta_D^s)p(\theta_D^s) \quad (19)$$

In the offline indexing stage, of these quantities, we need to pre-calculate only the document specific co-occurrence model $\sum_w P(w|\theta_D^s)P(q|\theta_D^s)p(\theta_D^s)$. Other quantities can be calculated easily from information provided by basic language modeling. When using the *select_ratio*(P) method for document specific model, time complexity for constructing co-occurrence information is about $O(P^2N)$. Compared with K, the average number of unique terms in a document, P is very small. When P is 0.1, P^2 is 0.01. In this case, the reduction ratio of time complexity is about 100 times. The final query model is obtained by substituting $T(w|q)$ into $T^s(w|q)$ in Eq. (8)

$$P(w|\theta_Q^F) = \alpha \left(\sum_{q \in Q} T^s(w|q)P(q|\hat{\theta}_Q) \right) + (1-\alpha)P(q|\hat{\theta}_Q) \quad (20)$$

5. EXPERIMENTATION

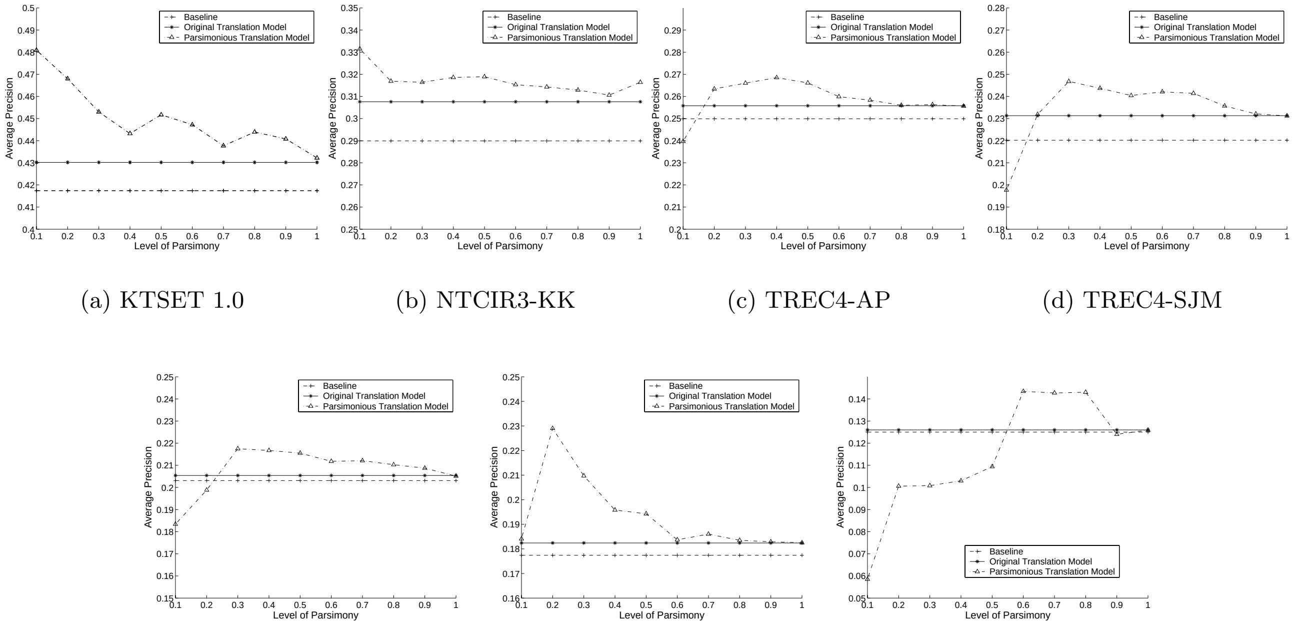
5.1 Experimental Setting

Our experimental database consists of two collections for Korean, and five TREC4 data collections for English. Table 1 summarizes the information of theseven data collections. The "# Doc" is the total number of documents, "# D.T." indicates the average number of unique terms of documents and "# Q" is the number of topics and "# R" is the number of relevant documents in each test set. To index Korean documents, we performed preliminary experimentations using various indexing methods (Morphology, word, and bi-character). It is well known that bi-character (n-Gram) indexing units are highly reliable for Korean or other Asian Languages, and our experimentations showed the same results. Thus, the bi-character indexing unit is used in this experimentation. For English documents, the typical preprocessing step is performed, where stop words are removed and then Poster stemming is applied.

For the baseline language modeling approach, we use Jelinek-Mercer smoothing, setting the smoothing parameter λ at 0.25. This smoothing parameter value is acquired empirically, by performing several experimentations across different parameters. This value is equally used for estimating parsimonious language models in Eq. (18).

5.2 Results of Retrieval Effectiveness

This section describes the retrieval results using the query model described in Section 3. For the query model, each



(a) KTSET 1.0

(b) NTCIR3-KK

(c) TREC4-AP

(d) TREC4-SJM

(e) TREC4-WSJ

(f) TREC4-ZIFF

(g) TREC4-FR

Figure 1: Average precision of OTM(Original Translation Model), PTM(Parsimonious Translation Model) and baseline language model in seven test collections

Table 1: Collection Summaries

Collection	*# Doc*	*# D.T.*	*# Q*	*# Term*	*# R*
KTSET 1.0	1,000	125.9	30	15,772	424
NTCIR3-K	66,147	176.3	30	180,294	3,868
TREC4-AP	158,240	156.6	49	268,411	3,055
TREC4-SJM	90,257	142	48	259,434	1,297
TREC4-WSJ	74,250	171.6	45	193,398	1,183
TREC4-ZIFF	217,940	90.6	32	578,607	512
TREC4-FR	19,860	124.1	27	98,567	416

Table 2: Comparison of performance between OTM and PTM

Collection	*Baseline*	*OTM*	*PTM**	*P**	*%chg*
KTSET 1.0	0.4174	0.4302	0.4809	0.1	11.79%
NTCIR3-KK	0.2899	0.3076	0.3315	0.1	7.7%
TREC4-AP	0.2499	0.2558	0.2685	0.4	4.96%
TREC4-SJM	0.2202	0.2313	0.2468	0.3	6.70%
TREC4-WSJ	0.2031	0.2054	0.2175	0.3	5.89%
TREC4-ZIFF	0.1774	0.1824	0.2290	0.2	25.55%
TREC4-FR	0.1250	0.1260	0.1434	0.6	13.81%

parameter is selected by empirical tuning. For all test collections, we fix parameters like $\eta = 1.0$ and $\alpha = 0.2$.

Figure 1 shows an average precision for three query models in seven different test collections by changing the parsimony level from 0.1 to 1.0. The three query models are: a baseline language model using MLE of query sample, a query model estimated from the original translation model (OTM, Eq. (8)), and a query model estimated from the parsimonious translation model (PTM, Eq. (20)).

As shown in Figure 1, in almost all parsimony levels, PTM significantly improves the baseline in six data collections. Remarkably, the performance of PTM is better than the performance of OTM at low parsimony levels. In OTM, some noise can occur because common query words can be expanded by common terms in document. Therefore, compared with the baseline, the high accuracy of PTM implies that it can effectively eliminate the noise of term expansion in OTM, and select good expansion terms for retrieval performance.

Concerning optimal parsimony level, while for the Korean collection the optimal parsimony level is 0.1, for the English test collection the optimal parsimony level is between 0.2 and 0.4. However, performance in TREC4-FR test collection is relatively exceptional. As shown in Figure 1, PTM does not perform well in parsimony levels not in the range 0.6 0.8. It seems that in TREC4-FR good expansion terms have relatively small probabilities value. The reason for this will be discussed in Section 5.3.

Table 2 summarizes the best performance of parsimonious models (PTM*) and OTM in seven different test collections. The last column with symbol %chg indicates improvement ratio of PTM* over OTM. P* is the parsimony level at the best. From this table, we know that PTM highly improves baseline from 5% to 25%, and especially 25.55% improvement is achieved at TREC4-ZIFF test collection.

5.3 Storage Size of Parsimonious Translation Model

Figure 2 shows that curves of the relative ratio to space size of MCTM among seven test collections. *Linear ratio* indicates the curve when the ratio is exactly same as the

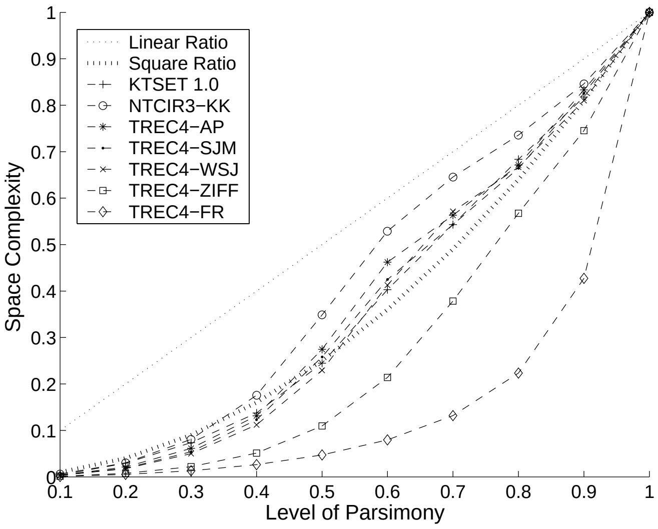


Figure 2: Space complexity (storage overhead) of PTM according to parsimony level on seven test collections

parsimony level and *Square ratio* indicates the curve when the ratio is exactly same as the square of parsimony level. When the parsimony level is 1.0, the ratio is 1.0. Here, square ratio reflects the relation of R of K in Section 3, where $R = O(NK^2)$. N is ignored because the ratio is a relative value to the full parsimonious model. If we assume that K is proportional to P, then R will be $O(NP^2)$. In Figure 2, the curve for each test collection has a shape similar to the square ratio. It means that estimation of space complexity in Section 3 is roughly consistent. Significantly ZIFF and FR draw very lower curves rather than the square ratio. This fact implies that the relation of P and K is very biased towards linear.

In collections other than ZIFF and FR, while the ratio curve follows the square ratio at the parsimony level below 0.4, it shows a tendency of leaning to the linear ratio in the levels over 0.4. In terms of retrieval performance, overall curves in Figure 2 are almost the same as the theoretical prediction in Section 3. Since in most test collections, the retrieval performance is the best when the parsimony level is less than 0.4, we will guarantee that at least the storage size in PTM is $0.4^2 = 0.16$ times of the storage size in MCTM.

6. CONCLUSION

Summing up, we propose an effective construction method for co-occurrence statistics using a parsimonious translation model. The parsimonious translation model involves an elegant method for selecting highly topical terms in documents, by document specific topic model. Basically, our idea is to use several state of the art methods in language modeling approaches for information retrieval. From experimentation on seven different collections, we show that a query model based on the parsimonious translation model preserves the effectiveness of traditional translation models, and remarkably reduces the time and space complexity of traditional translation models.

We have made interesting observations on the parsimony level in language of test collections. In Korean test collections the optimal parsimony level is very small, and in English test collections the optimal parsimony level is larger than one of Korean. In almost test collections, optimal parsimony levels are not over half of one.

7. ACKNOWLEDGMENTS

This work was supported by the KOSEF through the Advanced Information Technology Research Center(AITrc) and by the BK21 Project.

8. REFERENCES

[1] A. Berger and J. Lafferty. Information retrieval as statistical translation. In *Proceedings of ACM SIGIR*, pages 222–229, 1999.

[2] A. Dempster. Maximum likelihood from incomplete data via the em algorithm. *Journal of Royal Statistical Society*, 39(1):1–39, 1977.

[3] J. Gao, J. Nie, G. Wu, and G. Cao. Dependence language model for information retrieval. In *Proceedings of ACM SIGIR*, pages 170–177, 2004.

[4] D. Hiemstra, S. Robertson, and H. Zaragoza. Parsimonious language models for information retrieval. In *Proceedings of ACM SIGIR*, pages 178–184, 2004.

[5] R. Jin, A. Hauptmann, and C. Zhai. Title language model for information retrieval. In *Proceedings of ACM SIGIR*, pages 42–48, 2002.

[6] O. Kurland and L. Lee. Corpus structure, language models, and ad hoc information retrieval. In *Proceedings of ACM SIGIR*, pages 194–201, 2004.

[7] J. Lafferty and C. Zhai. Document language models, query models, and risk minimization for information retrieval. In *Proceedings of ACM SIGIR*, pages 111–119, 2001.

[8] V. Lavrenko and B. Croft. Relevance-based language models. In *Proceedings of ACM SIGIR*, pages 120–127, 2001.

[9] X. Liu. Cluster-based retrieval using language models. In *Proceedings of ACM SIGIR*, pages 186–193, 2004.

[10] R. Nallapati and J. Allen. Capturing term dependencies using a language model based on sentence trees. In *Proceedings of ACM CIKM*, pages 383–390, 2002.

[11] A. Ponte and J. Croft. A language modeling approach to information retrieval. In *Proceedings of ACM SIGIR*, pages 275–281, 1998.

[12] S. Robertson and K. S. Jones. Relevance weighting of search terms. *Journal of Royal Statistical Society*, 27(3), 1979.

[13] S. Ross. *Stochastic Process (2nd Edition)*. John Wiley and Sons, Inc., 1996.

[14] A. Singhal and F. Pereira. Document expansion for speech retrieval. In *Proceedings of ACM SIGIR*, pages 34–41, 1999.

[15] F. Song and W. Croft. A general language model for information retrieval. In *Proceedings of ACM SIGIR*, pages 279–280, 1999.

[16] R. Sperer and D. Oard. Structured translation for cross-language information retrieval. In *Proceedings of ACM SIGIR*, pages 120–127, 2000.

[17] M. Srikanth and R. Srihari. Biterm language models for document retrieval. In *Proceedings of ACM SIGIR*, pages 425–426, 2002.

[18] C. Zhai and J. Lafferty. A study of smoothing methods for language models applied to ad hoc information retrieval. In *Proceedings of ACM SIGIR*, pages 334–342, 2001.

[19] C. Zhai and J. Lafferty. Model-based feedback in the language modeling approach to information retrieval. In *Proceedings of ACM CIKM*, pages 403–410, 2002.