Chinese Information Retrieval Based on Terms and Ontology

Yang Lingpeng, Ji Donghong, Tang Li
Institute for Infocomm Research
21, Heng Mui Keng Terrace
Singapore 119613
{lpyang, dhji, tangli}@i2r.a-star.edu.sg

Abstract

In this paper, we describe our approach for single language information retrieval (SLIR) on Chinese language of NTCIR4 tasks. Firstly, we automatically extract terms (short-terms and long terms) from document set and use them to build indexes; secondly, for a query, we use short terms in the query and documents to do initial retrieval; thirdly, we build an ontology for the query to do query expansion and implement second retrieval. Finally, we use long terms to reorder the top N retrieved documents. Experiments show that the method achieves good results for both T-run and D-Run SLIR tasks of Chinese language.

Keywords: Ontology, Term Extraction, Chinese Information Retrieval, Query Expansion, Term Coverage

1. Introduction

At NTCIR4, we participated in SLIR tasks of Chinese language in Cross Lingual Information Retrieval (CLIR) track, where both the query and the document set are in traditional Chinese language. We submitted two compulsory runs: a T-run with some nouns or noun phrases about topics as queries and a D run with short descriptions of topics as queries [4].

For Chinese Information Retrieval, many retrieval models, indexing strategies and query expansion strategies have been proposed [2, 9, 12, 14]. Chinese Character, bi-gram, n-gram (n>2) and word are the most widely used indexing units. The effectiveness of single Chinese Characters as indexing units has been reported in [7]. The comparison between the three kinds of indexing units (single Characters, bi-grams and short-words) is given in [6]. It shows that single character indexing is good but not sufficiently competitive, while bi-gram indexing works surprisingly well and it's as good as short-word indexing in precision. [5] suggests that word indexing and bi-gram indexing can achieve comparable performance but if we consider the time and space factors, it is preferable to use words (and characters) as indexes. It also suggests that a combination of the longest-matching algorithm with single characters is a good method for Chinese IR and if there is a module for unknown word detection, the performance can be further improved. Some other researches [11] give similar conclusions. Bi-gram and word are considered as the top two indexing units in Chinese IR and they are also used in many reported Chinese IR systems in NTCIR tracks.

Regarding retrieval models, two models are most widely used in Chinese Information Retrieval, i.e., Vector Space Model [13] and Probabilistic Retrieval Model [10]. They are also adopted in most Chinese language experiments in NTCIR tasks.

For query expansion, most strategies make use of the top *N* retrieved documents in initial retrieval [14]. Generally, it selects *M* indexing units from the top *N* documents according to some criteria and adds these *M* indexing units to original query to form a new query. In such a process of query expansion, it's supposed that the top *N* documents are related with original query. However in practice, such an assumption is not always true.

In NTCIR4, we adopt automatically acquired terms as indexing units and build query-specific ontologies for query expansion. Firstly, we automatically extract terms (short-terms and long terms) from document set and use them to build indexes; secondly, we use short terms in query and documents to do initial retrieval; thirdly, we build an ontology for each query to do query expansion and implement the second retrieval; finally, we use long term coverage to re-order the retrieved documents. Figure 1 demonstrates the processes of our Chinese IR system.

The rest of this paper is organized as following. In section 2, we describe how to

automatically extract terms from document set. In section 3, we describe the retrieval model and weighting scheme used in our system. In section 4, we describe how to build ontology and how to use it to do query expansion in our system. In section 5, we describe how to refine the final ranking documents by using term coverage. In section 6, we evaluate the performance of our proposed method on NTCIR4 and give out some result analysis. In section 7, we present the conclusion and some future work.

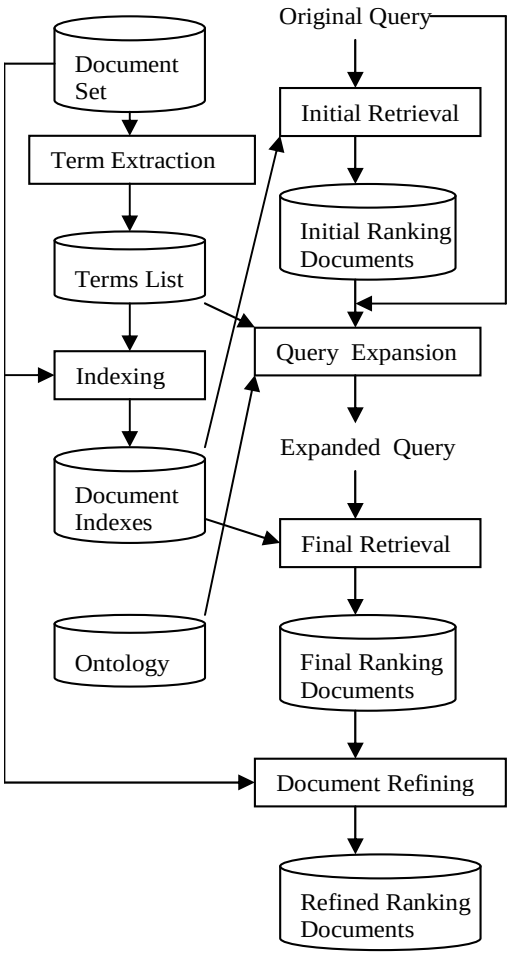

Fig. 1 Process of IR

2. Term Extraction

While bi-gram and word are most widely used as the indexing units in Chinese IR, we use automatically extracted terms as indexing units in our NTCIR4 track. The advantage is that we don't need a pre-defined dictionary.

We use a seeding-and-expansion mechanism to extract terms from documents (or document clusters). The procedure of term extraction consists of two phases, seed positioning and term determination. Intuitively, a seed for a candidate term is an individual word within the term, seed positioning is to locate the rough position of a term in the text, while term determination is to figure out which string covering the seed in the position forms a term.

To determine a seed needs to weigh the individual words to reflect their significance in the text in some way. To do so, we make use of a very large corpus *r* as a *reference*. Suppose *s* is the text of the collected summaries, *w* is an individual word in the text, let $P_r(w)$ and $P_s(w)$ be the probability of *w* occurring in *r* and *s* respectively, we adopt 1), *relative probability* or *salience* of *w* in *s* with respect to *r* [3], as the criteria for evaluation of seed words.

1) $P_s(w) / P_r(w)$

We call *w* a *seed* if $P_s(w) / P_r(w) \geq \delta$ ($\delta > 0$).

We have the following assumptions about a term.

i) a term contains at least a seed.
ii) a term occurs at least *N* times in the text.
iii) a *maximal word string* meeting i) and ii) is a term.
iv) for a term, a *real maximal substring* meeting i) and ii) without considering their occurrence in all those terms containing it is also a term.

Here a *maximal word string* meeting i) and ii) refers to a word string meeting i) and ii) while no other longer word strings containing it meet i) and ii). A *real maximal substring* meeting i) and ii) refer to a real substring meeting i) and ii) while no other longer real substrings containing it meet i) and ii).

Before term extraction, we roughly cluster the whole document set *r* into *K (K<2000)* clusters, then we regard each document cluster as one large document and apply the term extraction algorithm on it. The reason for document clustering is that some important terms may be statistically salient in a document cluster, which consists of similar documents, while they are not in a single document.

All the terms from different document clusters form a term list. We regard a term whose length is less than 4 Chinese Characters as a *short term*, and a term whose length is equal or greater than 4 Chinese Characters as a *long term.*

Here are some examples of short terms and long terms.

(1) Short Terms
劳工 (Laborer)
抗议 (protest)
劳委会 (Council of Labor Affairs)
诉求 (Appeal)

(2) Long Terms
约翰走路 (Jonnie Walker)
高尔夫球 (Golf)
老虎伍兹 (Tiger Woods)
胚胎干细胞 (embryonic stem cells)

There have been many document clustering approaches reported so far. K-Means and hierarchical clustering are two commonly used approaches. In our Chinese IR system, we only need to roughly cluster document set into some document clusters. So we use a simple K-Means approach. Firstly, we randomly pick up 10**K* documents from document set *r;* secondly, we use K-Means approach to cluster these 10**K* documents into *K* clusters; finally, we insert other documents into the *K* clusters.

In practice, we cluster the whole document set (CIRB011: 132,173 documents and CIRB020: 249,508 documents) into 2000 document clusters, then we extract terms from each of these 2000 document clusters.

All of the terms extracted from the 2000 document clusters form a term list. Every term in the list is called a *global term* because it's extracted based on the whole document set *r*. In fact, the term list can be considered as an automatically acquired dictionary, and can be used to find terms in a single document or a query. To find terms in a single document or a query, we make use of a variant of word segmentation method to segment a document or a query into a term string. Unlike traditional word segmentation method, a global term and its sub-terms may be considered as a term in a document simultaneously. For example, if $g = cd$ is a global term where *c* and *d* are also global terms, then *g*, *c* and *d* are all considered as terms in a document. The terms acquired from single documents are regarded as *local terms*.

The local terms in documents are used as indexing units to build index.

3. Retrieval Model

We use vector space models to represent documents and queries. Each document or query is represented as a vector in vector space where each dimension of vector is a term extracted from the document or the query. The weight of term *t* in document *d* is given by the following TF/IDF weight scheme:

2) $w(t, d)=\log(\mathrm{T}(t, d)+1) * \log(N/\mathrm{D}(t)+1)$

where, $w(t, d)$ is the weigh given to *t* in *d*, $\mathrm{T}(t, d)$ is the frequency of *t* in *d*, *N* is the number of documents in document set, $\mathrm{D}(t)$ is the number of documents in document set which contain *t*.

The weight of term *t* in query *q*, $w(t, q)$, is given by the following weight scheme:

3) $w(t, q) = \mathrm{T}(t, q)$

where $\mathrm{T}(t, q)$ is the frequency of *t* in *q*.

In the initial retrieval, only short terms are used to construct document vectors and query vectors. The reason is that for long terms, their frequency in documents tends to be lower, and their *tf/idf* values tend to be higher, if we include them in document vectors in initial retrieval, they may dominate the retrieval results, which will affect the recall.

The similarity (distance) between a document *d* and a query *q* is calculated by the cosine of the document vector and the query vector.

4. Ontology Construction and Query Expansion

For each query, we built an ontology specific to the query for its expansion. Another option is to construct an ontology for the whole document collection. However, there may be too many terms from the document collection to cluster them in practice. The terms of the query-specific otology come from the query itself, top documents in the initial retrieval list and internet. The terms from the query and the document are those included in them. In practice for query expansion, we only consider the top 20 documents retrieved in the initial retrieval.

The reason why we also get some terms from internet is that some relevant terms may be not statistically obvious in the document collection, while they may be acquired by search engine results. The terms from internet are those we extract from the snapshots returned by Yahoo! with the terms from the query as the search terms. The extraction process is the same as that for the document collection. In order to avoid noises from the search results, we select those terms which are already in the term list from the

document collection, however not regarded as relevant term yet.

After getting the terms, we conduct term clustering, which includes four sub-steps:

i) build a term by term matrix based their co-occurrence frequency;
ii) Feature reduction based on latent semantics analysis [15]. The dimension is reduced to 50.
iii) Construct a feature filter, which is to select the features which minimize the point-to-point entropy in 4) [8]. Intuitively, an optimal feature set should make the cluster structure more clear, and the point-to-point entropy will be lower. This process is to remove the noisy features for term clustering.
iv) Clustering based on MDL criteria in 5) and 6) [1]. This method can determine the optimal number of the term clusters.

4) $$-\sum_{i}^{n}\sum_{j}^{n}(d_{i,j}\log d_{i,j}+(1-d_{i,j})\log(1-d_{i,j}))$$

5) $$-\log p(y\mid K,\theta)+\frac{1}{2}L\log(NM)$$

6) $$L=K(1+M+\frac{M(M+1)}{2})-1$$

After term clustering, we can detect the ISA relationship between term clusters. The ISA relationship is based on the string mapping of the terms. In detail, if a term *a* is sub-string of another term *b*, then the cluster including *a* is *super-ordinate* of the cluster including *b*. For example, 智慧卡 (IC) is a substring of 非接触式智慧卡 (non-touching IC), so the cluster containing 非接触式智慧卡 is *sub-ordinate*, while the cluster containing 智慧卡 is *super-ordinate*.

For any term in the query, we can find a cluster, which contain the term. During query expansion, we use the cluster and its sub-ordinate clusters to expand the term in the query.

5. Document Re-Ordering

Document re-ordering is used to adjust the top M (M<2000) document in the initial retrieval and final retrieval. In the process of document re-ordering, long terms play an important role, since they tend to be more significant for the retrieval precision than short terms.

For a term *a* in a query, another term *b* in a document *cover a*, if they belong to one same term cluster. So, we can define the *coverage rate* of a document for a query as the ration of the number of the covered terms in the query against the number of all the terms in the query.

According to the coverage rate, we can re-order the retrieved documents. Intuitively, suppose *c*, *d* and *e* are terms of query *q*, then document *f* is more likely to be relevant with *q* than document *g* if document *f* contains term *c*, *d*, and *e*, but document *g* only contains *c* and *e*.

6. Results and Evaluation

We submitted two compulsory runs to NTCIR4: a T run which only uses field TITLE as queries and a D run which only uses field DESC as queries. There are altogether 59 query topics.

Table 1 and Table 2 list the statistical result of mean average precision (MAP) for the 59 query topics under relax relevance judgment and rigid relevance judgment. Relax Relevance Judgment considers highly relevant documents, relevant documents and partially relevant documents, while Rigid Relevance Judgment only considers highly relevant documents and relevant documents.

In table 1 and 2, column [C-C-T] represents Chinese to Chinese T run, [C-C-D] represents Chinese to Chinese D run; Row [min] represents the minimum MAP among the submitted results, Row [max] represents the maximum MAP among the submitted results, Row [med] represents the medium MAP among the submitted results, Row [ave] represents the average MAP of the submitted results, and Row [I^2R] represents our MAP results.

Table 1. Statistics on Rigid Judgment

	C-C-T	C-C-D
min	0.1327	0.1251
max	0.3146	0.3255
med	0.1881	0.1741
ave	0.1943	0.1826
I2R	0.3146	0.3255

Table 2. Statistics on Relax Judgment

	C-C-T	C-C-D
min	0.1638	0.1548
max	0.3799	0.388
med	0.2356	0.2219
ave	0.2378	0.2328
I2R	0.3799	0.388

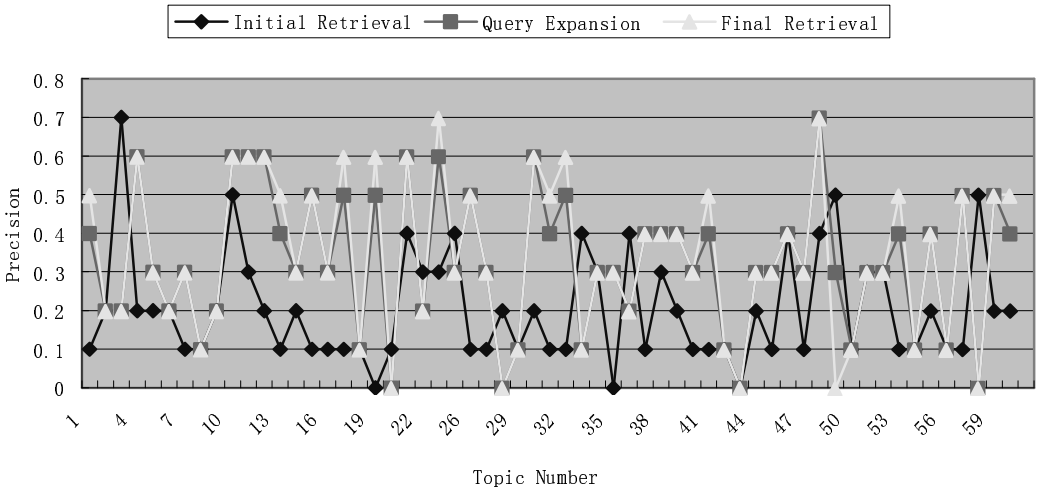

Fig. 2. Precision at top 10 documents (rigid)

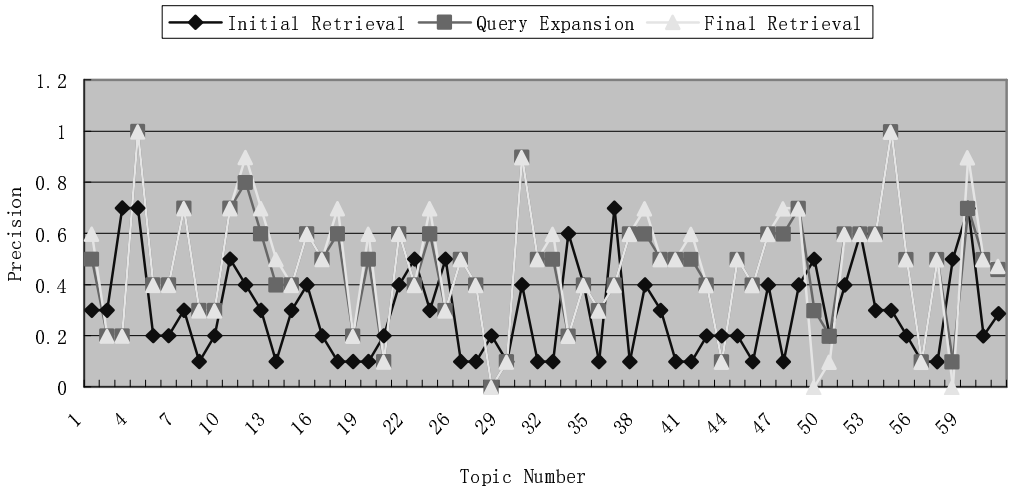

Fig. 3. Precision at top 10 documents (relax)

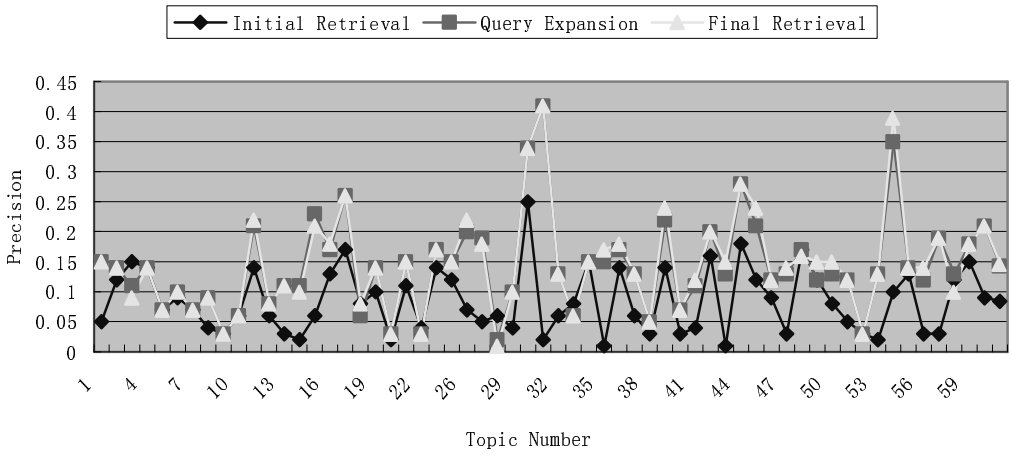

Fig. 4. Precision at top 100 documents (rigid)

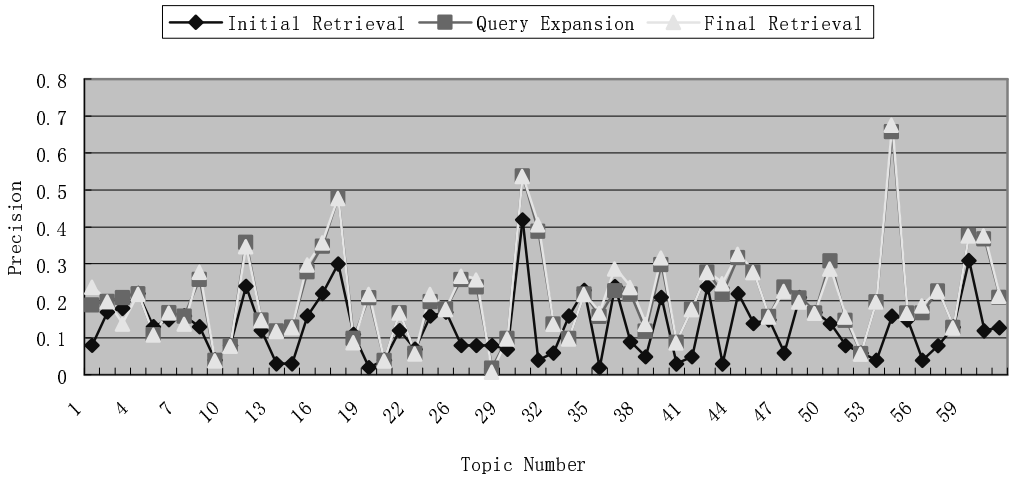

Fig. 5. Precision at top 100 documents (relax)

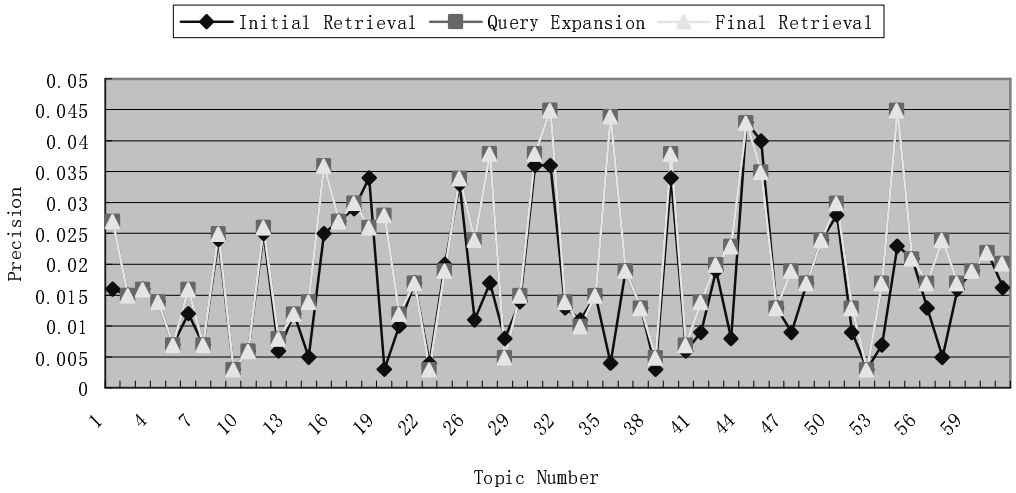

Fig. 6. Precision at top 1000 documents (rigid)

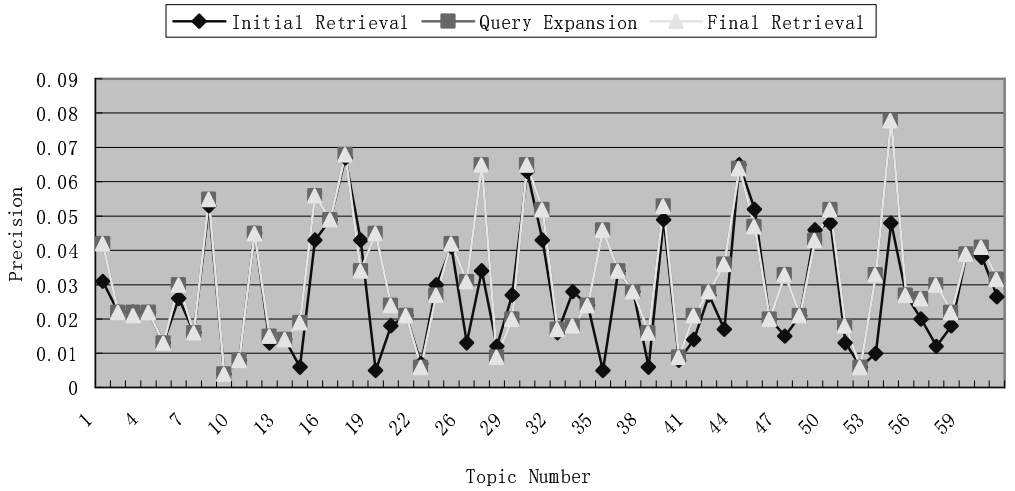

Fig. 7. Precision at top 1000 documents (relax)

From the statistical results, for T run, our group achieves 0.3146 and 0.3799 MAP on rigid and relax relevance judgment; for D run, our group achieves 0.3255 and 0.388 MAP on rigid and relax relevance judgment.

Fig. 2-7 gives the comparison between the precision of T-run in the three retrieval stages.

Although we get the best MAP results in average on both T run and D run under rigid relevance judgment and relax relevance judgment, we get poor results on several individual query topics. To find the reasons, we compare the final results with the initial search results without query expansion, and find that while query expansion based on ontology is useful for improving the precision of most queries, it affects some queries, which include query 9, 18, 28, 33 and 58.

The main reason for this lies in over expansion, i.e., irrelevant terms are introduced in the process of query expansion. As an example, for query 33:

<TITLE> 研究, 蛋白质 (Research, Protein) </TITLE>

We get some relevant terms for term 蛋白质 (Protein):

营养 (nutrition)
食物 (food)

These terms, although relevant with 蛋白质 (Protein) in some sense, are not relevant with the real topic of the title. So, they played negative roles on the retrieval results and reduced the precision of top *N* ranking documents.

As another example is query 58:

<TITLE> 非接触式智慧卡> (non-touching IC)</TITLE>

After query expansion, the precision of top 10 documents (PreAt10) becomes 0.0 from the original 0.5. Analysis shows that improper relevant terms caused such problem. The following is the list of the improper relevant terms for this query:

电子收费 (Electric pricing)
电子收费系统(Electric pricing system)
高速公路电子收费系统 (High way electric pricing system)

Another main reason for the poor performance is less expansion, i.e., the relevant terms are not captured by the expansion process. As an example, consider query 18:

<TITLE>青少年 社会问题(Social problems for young people)</TITLE>
<DESC>to retrieve the documents about the social problems caused by young people</DESC>

The really relevant terms for this query includes 青年(youth), 少年, 少男(young boy), 少女(young girl), 吸毒(drug), 抽烟(smoking), etc. However, the most relevant terms for 青少年 from the document collection are 健康 (health), and 营养 (nutrition), while the most relevant terms for 社会问题 from the document collection are 军事 (military), 股市(stock), etc.

7. Conclusion and Future Work

In this paper, we introduce our approach for Chinese IR and our experiments in the Chinese tasks of SLIR in NTCIR4. Our system achieves 0.3146 and 0.3799 MAP under rigid and relax relevance judgment for T run and 0.3255 and 0.388 MAP under rigid and relax relevance judgment for D run.

Although our system gets good results in both T run and D run, we find a lot of difficulties to push our approach into actual applications. The most difficult problem is how to acquire proper relevant terms and ontology knowledge.

Relevant terms are mainly based on the co-occurrence of terms in documents. Experiments show some relevant terms acquired in this way are not actually relevant with the given term. One possible solution in future is to detect relevant terms by co-occurrences of terms in paragraphs or in sentences. Another problem is that even if some terms are relevant with the query term, they are not relevant with the underlying topic described by the query. For this problem, we need to study how to get the topic-relevant terms, not only those just relevant with some query terms.

Another future work is to try to acquire more information-rich ontology, especially more semantic relationships between terms. Then, we can consider the contribution of these semantic relationships when doing query expansion.

Document re-ordering is very important for improving the precision. In future, we will study what kinds of terms are more significant for the precision, and how to re-order the documents based on these terms.

References

[1] A. C., Bouman, M., Shapiro, W. G., Cook, B. C., Atkins, and H., Cheng. 1998 Cluster: *An*

Unsupervised Algorithm for Modeling Gaussian Mixtures. http://dynamo.ecn.purdue.edu/bouman/software/cluster/.

[2] D.H. Ji, L.P. Yang, Y. Nie. *Chinese Language IR based on Term Extraction*. In The Third NTCIR Workshop.

[3] H. Schutze. 1998. T*he hypertext concordance: a better back-of-the-book index*. Proceedings of First Workshop on Computational Terminology. 101-104

[4] K. Kishida, K.H. Chen, S. Lee, K. Kuriyama, et al. *Overview of CLIR Task at the Fourth NTCIR Workshop*. In The Fourth NTCIR Workshop.

[5] J.Y. Nie, J. Gao, J. Zhang and M. Zhou. 2000. *On the Use of Words and N-grams for Chinese Information Retrieval*. In Proceedings of the Fifth International Workshop on Information Retrieval with Asian Languages, IRAL-2000, pp. 141-148

[6] K.L. Kwok. 1997. *Comparing Representation in Chinese Information Retrieval*. In Proceedings of the ACM SIGIR-97, pp. 34-41.

[7] Li. P. 1999.*Research on Improvement of Single Chinese Character Indexing Method*, Journal of the China Society for Scientific and Technical Information, Vol. 18 No. 5.

[8] M., Dash and H., Liu. 2000. *Feature Selection for Clustering*. Proceedings of Pacific-Asia Conference on Knowledge Discovery and Data Mining(pp. 110–121).

[9] M. Mitra., A. Singhal. and C. Buckley. *Improving Automatic Query Expansion*. In Proc. ACM SIGIR'98, Aug. 1998.

[10] N. Fuhr. *Probabilistic Models in Information Retrieval*. The Computer Journal. 35(3):243-254, 1992.

[11] Palmer, D. and Burger, J. *Chinese Word Segmentation and Information Retrieval*. AAAI Spring Symposium on Cross-Language Text and Speech Retrieval, Electronic Working Notes, 1997

[12] Chien, L.F. *Fast and quasi-natural language search for gigabytes of Chinese texts*. In: Proc. 18th ACM SIGIR Conf. On R&D in IR. Fox, E., Ingwersen, P. & Fidel, R. (eds.) ACM: NY, NY. Pp.112-120. 1995

[13] G. Salton and M. McGill. *Introduction to Modern Information Retrieval*, McGraw-Hill, 1983.

[14] S.E. Robertson and S. Walker. *Microsoft Cambridge at TREC-9: Filtering track*: In NIST Special Pub. 500-264: The Eight Text Retrieval Conference (TREC-8), pages 151-161, Gaithersburg, MD, 2001.

[15] T. K., Landauer and S. T., Dumais. 1997. *A solution to Plato's problem: The Latent Semanctic Analysis theory of the acquisition, induction, and representation of knowledge*. Psychological Review, 104 , 211-140.