Global and Local Term Expansion for Text Retrieval

Yuen-Hsien Tseng, Da-Wei Juang, and Shiu-Han Chen
Dept. of Library & Information Science,
Fu Jen Catholic University, Taipei, Taiwan, R.O.C., 242
tseng@lins.fju.edu.tw

Abstract

This paper describes our work at the fourth NTCIR workshop on the subtasks of monolingual information retrieval (IR). Global and local query expansions were explored. For global query expansion, co-occurred terms accumulated across the entire collection were selected and added to the initial query. For local query expansion, a method of blind relevance feedback (BRF) was implemented. Our experiments verified that BRF is effective and can be easily implemented without much parameter tuning. If best term selection can be achieved, global query expansion based on co-occurred terms can perform similarly well and combining both local and global expansion can outperform each method alone.

Keywords:
Chinese IR, relevance feedback, term association.

1. Introduction

In NTCIR-3, we participated in the Chinese, Japanese, and Korean single-language retrieval tasks (SLIRs) using an information retrieval system that dealt with these three languages in exactly the same way without using language-dependent knowledge or resources [1]. Results showed that our retrieval effectiveness does not show any difference among these three SLIRs. However, the effectiveness of our system was lower than the average of all runs submitted to the NTCIR by all participants. Post-analysis showed that the basic retrieval strategies used in our and others' systems did not match top-performing systems that used other sophisticated techniques such as blind relevance feedback (BRF), probabilistic retrieval model, hybrid term indexing, and title words re-weighting [2-4]. Among these techniques, BRF is the major approach that improves performance most. Thus in this year's NTCIR, we focus on the exploration of the relevance feedback techniques to see how effective they are under different implementations.

Relevance feedback is a technique that modifies the original query based on the initial retrieval results. If relevant (or irrelevant) terms can be identified from the initial results, adding them to (or subtracting them from) the original query for another run of retrieval often improves the retrieval effectiveness. However, since relevant terms are unknown to the system until inspected and feed backed by a searcher, most BRF methods under automatic retrieval mode simply assume that top-ranked documents retrieved from the initial query are relevant. Terms are then extracted from these "relevant" documents to add to the initial query. This way of query modification or query expansion is called "local expansion" since only a handful of documents "relevant" to the initial query are used. Information from the rest of the documents is not used at all. In contrast, if the feedback information comes from the entire collection, we call this way of relevance feedback "global expansion".

Both local and global expansions were implemented in our system. Next section will introduce our way of obtaining global relevant terms for query expansion based on term co-occurrence. Section 3 will describe our indexing and retrieval strategies including the details of the local and global expansion methods. In Section 4, we report our retrieval results submitted to the NTCIR and the post-run results. Finally we conclude and summarize our observations in Section 5.

2. Extraction of Global Relevant Terms

There are a number of approaches to extract terms relevant to the same topics from the entire document collection. One commonly used heuristic rule is based on term co-occurrence. Salton had proposed a framework that computes term similarity based on co-occurrence to reveal how two terms are relevant to each other [5]. His idea works as follows. Given a collection of n documents, an inverted term-document structure is first constructed, where

each term is denoted in a vector form whose elements are weights of the term in the documents, such as: $T_j=(d_{1j}, d_{2j}, \ldots, d_{nj})$. Similarities are then computed among all useful term pairs. A typical similarity measure is given by cosine similarity:

$$sim(T_j, T_k) = \sum_{i=1}^{n} d_{ij} d_{ik} \Big/ \sqrt{\sum_{i=1}^{n} d_{ij}^2 \sum_{i=1}^{n} d_{ik}^2}$$

If the weights of the terms were either 1 or 0, denoting the presence or absence of the terms in documents, the similarity becomes a value exactly proportional to the number of documents in which these two terms co-occur. With these pair-wise similarities, terms are clustered with some automatic processes. These term clusters can then be applied to document retrieval by expanding the query terms with their similar terms in the same clusters.

However, the above method requires a lot of computations. With m distinct terms in a collection of n documents, this can be an $O(m^2n)$ algorithm (n steps to calculate similarity between any of $O(m^2)$ term pairs). Normally m is often at least as large as n, making the algorithm consume a lot of computation resources. Besides, terms co-occur in the same document may virtually have no relationship if they are far apart from each other in the text. Calculating their term similarities in this way may turn out to be a waste of computation power.

Therefore, we proposed another method that is far more efficient. The major difference of our method from the above is to limit the terms to be associated to those that co-occur in the same logical segments of a smaller text size, such as a sentence or a paragraph. Association weights are computed in this way for each document and then accumulated over all documents. This changes it into a roughly $O(nk^2s)$ algorithm, where k is the number of selected keywords for association and s is the average number of sentences in a document.

Specifically, keywords or key terms extracted from each documents are first sorted in decreasing order of their term frequencies (or *tf* x *idf* or other criterion if the entire collection statistics are known in advance) and the first k terms are selected for term association analysis. A modified Dice coefficient was chosen to measure term pairs' association weights as:

$$wgt(T_{ij}, T_{ik}) = \frac{2 \times S(T_{ij} \cap T_{ik})}{S(T_{ij}) + S(T_{ik})} \times \ln(1.72 + S_i)$$

where S_i denotes the number of sentences (or paragraphs) in document i and $S(T_{ij})$ denotes in document i the number of sentences in which term T_j occurs. Thus the first term is simply the Dice coefficient similarity [5]. The second term $ln(1.72+S_i)$, where ln is the natural logarithm, is used to compensate for the weights of those terms in longer documents so that weights in documents of different length have similar range of values. This is because longer documents tend to yield weaker Dice coefficients than those generated from the shorter ones. Association weights larger than a threshold (1.0 in our experiments) are then accumulated over all the documents in the following manner:

$$sim(T_j, T_k) = \frac{\log(w_k \times n/df_k)}{\log(n)} \times \sum_{i=1}^{n} wgt(T_{ij}, T_{ik})$$

where df_k is the document frequency of term k and w_k is the width of terms k (i.e., number of constituent words in English or number of constituent characters in Chinese, Japanese, or Korean).

Computation of the similarities among all term pairs can be carried out as the index structure for the entire collection is constructed. Weights of term pairs from each document are calculated and accumulated just like the index terms accumulating their document frequencies and postings [6]. In this way, a global term relation structure can be obtained efficiently. For the 381,375 Chinese documents in the NTCIR-4 (469 MB of texts), it only takes 133 minutes on a notebook computer with a 1.7 GHz CPU and 512 Mega RAM for indexing, keyword extraction, and term association computation.

These associated terms not only can be added to the initial query in an automatic mode for global query expansion, but also can they be prompted to the searcher in an interactive mode for suggesting additional query candidates or for revealing the underlying knowledge among concepts, topics, or persons. An example is shown in Figure 1. The highlighted search term: "Akira Kurosawa" is from the topic 012 of this year's CLIR topic set. Twelve top-ranked co-occurred terms were shown, with more relevant terms in closer distance to the search term. In our implementations, at most 64 co-occurred terms for each keyword were kept in the term relation structure for later use.

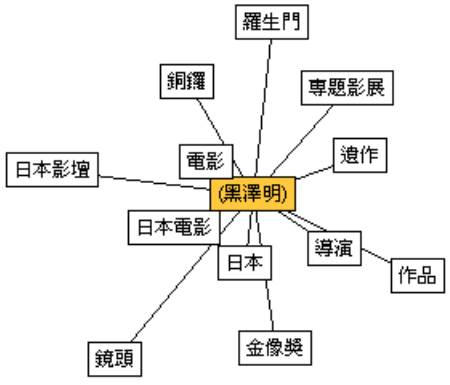

Figure 1. An example of global term expansion.

3. Indexing and Retrieval Scheme

The index terms in our system are bi-grams, dictionary words, and keywords extracted from each documents based on the algorithm proposed in [7]. The used dictionary covers over 120,000 words and in average 1/3 of the extracted keywords per document are unknown to the dictionary. Query strings are segmented by these index terms based on a longest-match and back tractable approach.

These index terms are then used to compute the similarity between a query and each document based on the following vector space model:

$$Sim(d_i, q_j) = \frac{\sum_{k=1}^{T} d_{i,k} q_{j,k}}{(bytesize_{d_i})^{0.375} \sqrt{\sum_{k=1}^{T} q_{j,k}^2}}$$

where the *bytesize* denotes the number of bytes of a document. The use of the *bytesize* as the normalization factor is first introduced in the work of Singhal, et al [8] for OCR text retrieval, where the authors found that the commonly used cosine normalization factor has negative effects on retrieval when documents contain erroneous terms, such as those garbled by OCR errors or those mistyped or misspelled terms. Singhal et al also applied this bytesize normalization to large collections of TREC documents. They found that it also leads to better effectiveness than cosine normalization for ordinary documents. Besides, the bytesize normalization is easier to compute than the cosine normalization. Thus we use this formula in our retrieval experiment.

The document term weight $d_{i,k}$ in the above is calculated by the term frequency, i.e., *log(1+tf)*. The query term weight $q_{j,k}$ is calculated by the term frequency and the inverse document frequency, i.e., *log(1+tf)* x *log(1+n/df)*, where n is the collection size.

For comparison, we also implemented a probabilistic retrieval model that computes the okapi weight (*BM11*) of a document [9]:

$$BM11(d_i, q_j) = \sum_{k=1}^{T} tf_{j,k} \log\left(\frac{n - df_k + 0.5}{df_k + 0.5}\right)\left(\frac{tf_{i,k}}{tf_{i,k} + \frac{dl_i}{Avgdl}}\right)$$

where T is the number of query terms, dl_i is the document length of document i, and $Avgdl$ is the average document length of all documents.

As to the BRF, thirty best terms from six top-ranked documents retrieved by the initial query were used. These six documents were first concatenated into one text string and then the keyword extraction algorithm [7] was applied to extract maximally repeated patterns. The extracted terms were sorted in decreasing order of occurrence. The first 30 terms were then selected for local query expansion. The decision on the number of best terms and the number of top-ranked documents was quite arbitrarily. We chose these numbers from the beginning almost without any tuning.

For global query expansion, each query string was segmented by the index terms in the manner described above. High-frequency terms (whose df >= n/20) were discarded. Then associated terms (described in the previous section) of each resultant term were fetched. Associated terms having larger document frequency than its corresponding term were also discarded. Finally only those associated terms associated to at least two segmented terms were used for expansion. These limitations on the added terms were to avoid *topic drift*, a phenomenon that changes the topic of the original query as more (irrelevant) terms were added. But this also limits the number of terms for global query expansion such that most topics have only a few additional terms.

4. Experiment Results

The query topics prepared by NTCIR-4 consist of *title*, *description*, *narrative*, and *concept* fields. A total of 60 topics for Chinese SLIR were provided for retrieval. But due to the lack of sufficient relevant documents, one topic was discarded, leaving only 59 topics for evaluation. Average precision for each topic was calculated by the well-known trec_eval program [10]. They were then averaged over the 59 topics. The final average was denoted as MAP (Mean Average Precision). Participants can submit multiple results, each comes from the run that uses different fields of query topics and/or different retrieval strategies to see how MAP changes. Each submitted run is evaluated in two criteria, one is *relax*, meaning that the relevance judgment is done in a less strict way; the other is *rigid*, meaning that the relevance is judged in a more rigorous sense.

Our results for the Chinese SLIR (C-C runs) were shown in Table 1. In the RunID, the letter T denotes the run that submits the *titles* as queries, D denotes the *descriptions*, and C the *concepts*. The AP in the RunID denotes that the queries were expanded by associated terms, i.e., global expansion, while the BRF denotes the blind relevance feedback, i.e., the local expansion. The RunID appended with '(p)' denotes that the run used the probabilistic model, while all others used the vector space model. Both the results from the query topics of NTCIR-3 and NTCIR-4 are provided. Those runs whose RunID having a '*' are post-runs, that is, they are not the official runs submitted to NTCIR for evaluation. The maximum, average, and minimum of the MAPs among all similar runs submitted by all participants were also included for reference.

NTCIR3					NTCIR4				
RunID	Rigid		Relax		RunID	Rigid		Relax	
	MAP	% imp	MAP	% imp		MAP	% imp	MAP	% imp
C-C-D	0.1858	-	0.2281	-	*C-C-D	0.1449	-	0.1960	-
*C-C-D+AT	0.1894	1.94	0.2432	6.62	C-C-D+AT	0.1486	2.55	0.2011	2.60
*C-C-D+BRF	0.2246	20.88	0.2796	22.58	C-C-D+BRF	0.1689	16.56	0.2267	15.66
*C-C-D+BRF(p)	0.2474	33.15	0.3009	31.92	*C-C-D+BRF(p)	0.1701	17.39	0.2200	12.24
Max of C-C-D	0.3933		0.4990		Max of C-C-D	0.3255		0.3880	
Avg of C-C-D	0.2130		0.2670		Avg of C-C-D	0.1826		0.2328	
Min of C-C-D	0.0347		0.0443		Min of C-C-D	0.1251		0.1548	
C-C-C	0.1997	-	0.2403	-	*C-C-T	0.1636	-	0.2052	-
*C-C-C+AT	0.2048	2.55	0.2358	-1.87	C-C-T+AT	0.1685	3.00	0.2091	1.90
*C-C-C+BRF	0.2377	19.03	0.2981	24.05	C-C-T+BRF	0.1881	14.98	0.2356	14.81
*C-C-C+BRF(p)	0.2524	26.39	0.3026	25.93	*C-C-T+BRF(p)	0.1956	19.56	0.2380	15.98
Max of C-C-C	0.2386		0.2929		Max of C-C-T	0.3146		0.3799	
Avg of C-C-C	0.2104		0.2605		Avg of C-C-T	0.1943		0.2378	
Min of C-C-C	0.1831		0.2403		Min of C-C-T	0.1327		0.1638	

Table 1: Mean average precisions (MAP) of FJUIR in the C-C track.
Runs marked with '*' are post-runs. That is, they are not the official runs submitted to the NTCIR for evaluation. Percentage improvements in MAP (% imp) are calculated from the nearest rows with "-" as basis.

As can be seen in Table 1, the probabilistic retrieval model performs slightly better than the vector space model regardless of long or short queries, verifying past experiments in NTCIR. Another salient result is that local expansion performs far better than global expansion.

Despite the different implementation from others, our results confirm that BRF can boost the effectiveness substantially. This shows that BRF is quite a reliable approach that is not sensitive to detailed implementations or different parameter tuning. On the contrary, the global expansion yields almost no improvement. This may be due to the failure of our expansion method. We choose a very conservative approach to avoid topic drift. This leads to very few terms added to the query and the results make little difference. To see how effectiveness changes if we have a best selection for global expansion, we run a small experiment by manually selecting the global relevant terms from those associated to the query terms. Table 2 shows an example of running the topic 012 with various expansion combinations. The title of this topic is "director, Akira Kurosawa". It has a total of 8 and 15 relevant documents in rigid and relax assessment, respectively. First row is the results of global expansion using our automatic method. No additional terms were added since associated terms from "director" and "Akira Kurosawa" do not pass the limitation rules. Next run is manual selection of the associated terms from these two query terms. All the terms in Figure 1 (i.e., the best twelve associated terms of "Akira Kurosawa") were added, yielding 93.20% and 69.16% improvement in rigid and relax assessment, respectively. Compared to the BRF, this manual selection of terms performs obviously better. If manual selection was combined with BRF, the effectiveness climbs even higher, implying that neither one is optimal and that one can be re-enforced by the other. Those cases using the probabilistic retrieval model have the similar results, only they are even better than those using the vector space model.

RunID	Rigid		Relax	
	MAP	% imp	MAP	% imp
C-C-T+AT	0.2119	-	0.3217	-
C-C-T+MT	0.4094	93.20	0.5442	69.16
C-C-T+BRF	0.2881	35.96	0.3912	21.60
C-C-T+MT+BRF	0.4795	126.29	0.5962	85.33
C-C-T+AT(p)	0.2472	16.66	0.3892	20.98
C-C-T+MT(p)	0.4174	96.98	0.5918	83.96
C-C-T+BRF(p)	0.3602	69.99	0.5576	73.33
C-C-T+MT+BRF(p)	0.6707	216.52	0.6779	110.72
Max of C-C-T	0.7145		0.7492	
Avg of C-C-T	0.5083		0.5954	
Min of C-C-T	0.2119		0.3217	

Table 2: Average precisions of topic 012, where MT denotes manual selection of associated terms for query expansion.

5. Conclusions

We have verified that blind relevance feedback is an effective and stable approach to boost retrieval effectiveness. It only uses local information, i.e., the documents at the top-ranked positions retrieved by the initial query. In this year's NTCIR, we attempt to use global information for query expansion by adding co-occurred terms accumulated across the entire collection in an automatic manner. But our attempt has failed. However preliminary experiments show that if best term selection can be made, the retrieval effectiveness can be better than that of BRF. Furthermore, using both local and global information together for query expansion can be better than using each alone. Our future studies will focus on effective methods of automatic term selection from globally co-occurred terms for query expansion.

This year our performance has improved. But our best performing runs only match the average of all similar runs. The high performance of other systems is still a mystery to us. Without further information on how these systems work at the time this article is written, we wonder which retrieval strategies else make this difference. Anyway, we expect that our system will continue to improve itself in the near future after learning some lessons from this workshop.

Acknowledge

This work is supported in part by NSC under the grant number NSC 91-2413-H-030-012-.

References

[1] Da-Wei Juang and Yuen-Hsien Tseng, "Uniform Indexing and Retrieval Scheme for Chinese, Japanese, and Korean," Proceedings of the Third NTCIR Workshop on Evaluation of Information Retrieval, Automatic Text Summarization and Question Answering, Oct. 8-10, 2002, Tokyo, Japan, pp.137-141.

[2] K. L. Kwok, "NTCIR-3 Chinese, Cross Language Retrieval Experiments Using PIRCS," Proceedings of the Third NTCIR Workshop on Evaluation of Information Retrieval, Automatic Text Summarization and Question Answering, Oct. 8-10, 2002, Tokyo, Japan, pp. 45-49.

[3] Masaki Murata, Qing Ma, and Hitoshi Isahara, "Applying Multiple Characteristics and Techniques to Obtain High Levels of Performance in Information Retrieval," Proceedings of the Third NTCIR Workshop on Evaluation of Information Retrieval, Automatic Text Summarization and Question Answering, Oct. 8-10, 2002, Tokyo, Japan, pp.87-92.

[4] Robert W. P. Luk, K. F. Wong, and K. L. Kwok, "Different Retrieval Models and Hybrid Term Indexing," Proceedings of the Third NTCIR Workshop on Evaluation of Information Retrieval, Automatic Text Summarization and Question Answering, Oct. 8-10, 2002, Tokyo, Japan, pp.93-100.

[5] Gerard Salton, Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer, Addison-Wesley, 1989.

[6] William B. Frakes and Ricardo Baeza-Yates, *Information Retrieval: Data Structure and Algorithm*, Prentice Hall, 1992.

[7] Yuen-Hsien Tseng, "Automatic Thesaurus Generation for Chinese Documents", Journal of the American Society for Information Science and Technology, Vol. 53, No. 13, Nov. 2002, pp. 1130-1138.

[8] Amit Singhal, Gerard Salton, and Chris Buckley, "Length Normalization in Degraded Text Collections," Proceedings of Fifth Annual Symposium on Document Analysis and Information Retrieval, April 15-17, 1996, pp. 149-162.

[9] S. E. Robertson and S. Walker, "Some Simple Effective Approximations to the 2-Poisson Model for Probabilistic Weighted Retrieval," Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval, August 1994, pp. 232-241.

[10] Chris Buckley, ftp://ftp.cs.cornell.edu/pub/smart/trec_eval.v3beta.shar