Available online at www.sciencedirect.com

ScienceDirect

Information Processing and Management 43 (2007) 315–326

INFORMATION PROCESSING & MANAGEMENT

www.elsevier.com/locate/infoproman

Document reranking by term distribution and maximal marginal relevance for chinese information retrieval

Lingpeng Yang, Donghong Ji [*], Munkew Leong

Institute for Infocomm Research, Media Understanding, 21 Heng Mui Keng Terrace, Singapore 119613, Singapore

Received 25 May 2006; accepted 25 July 2006

Abstract

In this paper, we propose a document reranking method for Chinese information retrieval. The method is based on a term weighting scheme, which integrates local and global distribution of terms as well as document frequency, document positions and term length. The weight scheme allows randomly setting a larger portion of the retrieved documents as relevance feedback, and lifts off the worry that very fewer relevant documents appear in top retrieved documents. It also helps to improve the performance of maximal marginal relevance (MMR) in document reranking. The method was evaluated by MAP (mean average precision), a recall-oriented measure. Significance tests showed that our method can get significant improvement against standard baselines, and outperform relevant methods consistently.

Keywords: Relevance feedback; Term extraction; Term weighting; Maximal marginal relevance; Chinese information retrieval

1. Introduction

How to further improve the rankings of the relevant documents after an initial search has been extensively studied in information retrieval. Such studies include two main streams: automatic query expansion and automatic document reranking. While the assumption behind automatic query expansion is that the high ranked documents are likely to be relevant so that the terms in these documents can be used to augment the original query to a more accurate one, document reranking is a method to improve the rankings by re-ordering the position of initial retrieved documents without doing a second search. After document reranking, it is expected that more relevant documents appear in higher rankings, from which automatic query expansion can benefit.

Many methods have been proposed to rerank retrieved documents. Lee, Park, and Choi (2001) proposes a document reranking method based on document clusters. They build a hierarchical cluster structure for the whole document set, and use the structure to rerank the documents. Balinski and Danilowicz (2005) proposes a document reranking method that uses the distances between documents for modifying initial relevance

[*] Corresponding author.
E-mail addresses: lpyang@i2r.a-star.edu.sg (L. Yang), dhji@i2r.a-star.edu.sg (D. Ji), mkleong@i2r.a-star.edu.sg (M. Leong).

0306-4573/$ - see front matter © 2006 Elsevier Ltd. All rights reserved.
doi:10.1016/j.ipm.2006.07.011

weights. Luk and Wong (2004) uses the title information of documents to rerank documents, while Crouch, Crouch, Chen, and Holtz (2002) uses the un-stemmed words in queries to re-order documents. Xu and Croft (1996, 2000) makes use of global and local information to do local context analysis and then use the information acquired to rerank documents. Qu, Xu, and Wang (2000) uses manually built thesaurus to rerank retrieved documents, and each term in a query topic is expanded with a group of terms in the thesaurus. Bear et al. (1997) uses manually crafted grammars for topics to re-order documents by matching grammar rules in some segment in articles. Kamps (2004) proposes a reranking method based on assigned controlled vocabularies. Yang, Ji, and Tang (2004, 2005) use query terms which occur in both query and top $N(N \Leftarrow 30)$ retrieved documents to rerank documents.

One problem in automatic document reranking (also for query expansion) is how many top documents are regarded as relevance feedback in the first retrieval results, which is also faced by most methods mentioned above (Crouch et al., 2002; Kamps, 2004; Lee et al., 2001; Luk & Wong, 2004; Yang et al., 2004, 2005). Usually, a pre-defined smaller number of the documents (say top 10–30) are considered. However, in the cases that very few relevant documents fall within the range, the method will fail. On the other hand, if a larger scope (say 500, 1000) is considered, many irrelevant documents will come inside, and the noisy terms will dominate.

Another problem is that most methods mentioned above do not consider correlation between query terms. Mitra, Singhal, and Buckley (1998) uses maximal marginal relevance (MMR) to adjust the contribution of relevant terms. They argue that usually a document covering more aspects of a query should get higher score, which can be captured somehow by word correlation. The new score for a document is computed by summing the idf (inverse document frequency) of each query word where each word is normalized by correlation probability based on a large number of retrieved documents (say top 1000 documents). It is reported that their method achieves better result in reranking top 50–100 documents. But we find that within top initially retrieved documents, some really relevant terms do appear in larger portion of the documents, which will be unexpectedly assigned lower scores by idf scheme.

In this paper, we propose a new term weighting scheme to deal with the two problems mentioned above. First, we consider document rankings, i.e., document positions in the ranking list, in the weighting scheme of the terms. Intuitively, a term gets a lower document frequency when occurring in a lower-ranking document, and a higher document frequency when occurring in a higher-ranking document (in contrast, the usual way for document frequency is that a document gets 1 count no matter where the document is located in the list). In this way, we can randomly choose a larger number of the documents as relevance feedback, without any worry about the irrelevant documents inside. Furthermore, we do not need to worry about the cases that top documents only contain very few relevant documents, since we can randomly set a larger scope as relevance feedback.

Second, the weighting scheme incorporates both local (feedback) and global distribution of the terms, and we use it to replace the idf scheme in MMR. If a term occurs in feedback documents more frequently than in the whole collection, it tends to have more contribution to document reranking; otherwise, it will be a noise.

Our method does not use word but uses the key terms extracted from queries and top retrieved documents. One motivation of this choice is that terms (including multi-word units) usually contain more complete information than individual words, and have more potential for improving the performance of information retrieval. Another motivation of this method is specifically for Chinese language information retrieval, where a word segmentation module is usually needed, which, however, generally requires some manual resources and suffers from the problem of portability. An automatic term extraction module could be a good alternative.

The rest of this paper is organized as the following. In Section 2, we describe key term extraction from documents. In Section 3, we talk about term weighting. In Section 4, we specify how to rerank the documents based on the key terms and their weighting together with MMR based on term correlation. In Section 5, we evaluate the method on NTCIR-3 CLIR Chinese SLIR document collection and give some analysis. In Section 6, we present the conclusion and future work.

2. Term extraction

Term extraction concerns the problem of what is a term. Intuitively, key terms in a document are some word strings which are conceptually prominent in the document and play main roles in discriminating the document from other documents.

We use a seeding-and-expansion mechanism to extract key terms from documents. 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 (or a Chinese character in the case of Chinese language, henceafter, we focus on Chinese language), 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 individual Chinese characters to reflect their significance in the text. We make use of a very large corpus r (LDC's Mandarin Chinese News Text) as a reference corpus. Suppose d is a document, c is an individual Chinese Character in the text, let $P_r(c)$ and $P_d(c)$ be the probability of c occurring in r and d, respectively, we adopt relative probability or salience of c in d with respect to r (Schutze, 1998), as the criteria for evaluation of seeds.

$$P_d(c)/P_r(c). \tag{1}$$

We call c a *seed* if $P_d(c)/P_r(c) \geqslant \delta(\delta \geqslant 1)$. That is, its probability occurring in document must be equal with or higher than its average probability in the reference corpus.

Although it is difficult to give out the definition of terms, we have the following definition about a key term in a document.

(i) A term contains at least one seed.
(ii) A term occurs at least L ($L > 1$) times in the document.
(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 the sub-string 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).

The above assumptions tell us a term is an independent maximal string which must contain a seed and occur at least 2 times in a document. For example, given a document d, suppose a Chinese character 博 (bo3) is a seed in d, 故宫博物院 (National Palace Museum) occurs 3 times in d, 博物院 (Museum) occurs 5 times in d, if we set the parameter L as 2, then both string 故宫博物院 (National Palace Museum) and 博物院 (Museum) are terms in d; but if we set the parameter L as 3, then 故宫博物院 (National Palace Museum) is term in d, but 博物院 (Museum) is not a term in d because its independent occurrence is 2 (excluding 3 occurrences as a sub-string in 故宫博物院 (National Palace Museum)).

3. Query term weighting based on term distribution

To rerank retrieved documents, we use the key terms in the documents, and suppose that these key terms will contribute to the reranking. Here, we only focus on the terms which also occur in the queries, which means that we do not use any query expansion. So, the terms can also be referred to as query terms. To weigh a query term, we consider the following three factors.

(i) Relative distribution: the ratio of document frequency of a term in the top K retrieved document against the document frequency of the term in the whole document collection.Intuitively, the more frequently a term occurs in the K documents relative with the whole collection, the more important the term tends to be.
(ii) Term length: the number of Chinese characters a term contains.Intuitively, the longer a term is, the more contribution to the precision the term may have.
(iii) Document ranking position: the serial number of a document in top K documents.

Intuitively, the higher ranking a document is, the more important the terms in it tend to be.

Given top K retrieved documents and query term t, the weight assigned to t is given by the following formula.

Table 1
Document frequency weighting

Scheme name	Scheme definition
W4	$f(i) = 1/\text{sqrt}(i)$
W5	$f(i) = 1 + 1/\text{sqrt}(i)$
W6	$f(i) = 1/(1 + \log(i))$
W7	$f(i) = 1$
W8	$f(i) = 1/i$
W9	$f(i) = 1 + 1/i$

$$\sqrt{\frac{\left(\sum_{i=1}^{K} df(t,d_i) \times f(i)\right)/K}{\mathrm{DF}(t,C)/R}} \times \sqrt{|t|}, \tag{2}$$

$$df(t,d_i) = \begin{cases} 1 & t \notin d_i \\ 0 & t \in d_i \end{cases} \tag{3}$$

where d_i is the ith$(i = 1,\ldots,K)$ document, R is the total number of documents in the whole collection C, $\mathrm{DF}(t,C)$ is the number of documents which contain t in C, $|t|$ is the length of term t, $f(i)$ is the document frequency weighting given to d_i. Table 1 lists 6 document frequency weighting schemes used in our experiments.

4. Document reranking

In the reranking phase, we consider queries with multiple aspects or concepts. To prevent from query drift, we prefer a document that matches the query on multiple independently concepts. In other words, we need to distinguish multiple query-document matches: matches on query terms related to the same aspect, or matches on query terms from different aspects of the query. Thus, a match on two independent query concepts should be considered more useful than a match on two strongly related query terms.

We use term correlation to measure the relatedness or independence of query terms and use MMR criteria to reduce redundancy of query terms while maintaining query relevance in reranking retrieved documents.

To estimate the relatedness or independence of query terms, we study their co-occurrence patterns in top K initially retrieved documents. If two query terms are correlated, then they are expected to occur together in many of these documents. Given the presence of one of the query terms in a document, the chance of the other occurring within the same document is likely to be relatively high. On the other hand, if two query terms deal with independent concepts, the occurrences of the query terms should not be strongly correlated. Given query term t_j, top K retrieved documents as document set S, we define the correlation in S between query term t_i and t_j regarding t_j as $P(t_i|t_j)$:

$$P(t_i|t_j) = \frac{\text{number of documents in } S \text{ containing query term } t_i \text{ and } t_j}{\text{number of documents in } S \text{ containing query term } t_j}. \tag{4}$$

To rerank each document d in top $M(M \Leftarrow K)$ retrieved documents, we first find out the query terms which occur in d, then we consider the matching query terms in decreasing order of query term weight. The first matching query term contributes its full weigh to $\text{Weight}_{\text{new}}$. The contribution of any subsequent match is deprecated on how strongly this match was predicted by a previous match – if a matching query term is highly correlated to a previous match, the contribution of the new match is proportionately down-weighted. Finally, we use $\text{Weight}_{\text{new}}$ and the initial similarity between d and query q to calculate a new ranking score and then use the new ranking score to re-order the M documents. More precisely, if $\{t_1,\ldots,t_m\}$ is the set of query terms presented in document d (ordered by decreasing query term weight), then $\text{Weight}_{\text{new}}$ and $\text{Score}_{\text{new}}$ are given by:

$$\text{Weight}_{\text{new}} = w(t_1) + \sum_{i=2}^{m} w(t_i) \times \min_{j=1}^{i-1}(1 - P(t_i|t_j)), \tag{5}$$

$$\text{Score}_{\text{new}} = (1 + \text{Weight}_{\text{new}}) \times \text{Sim}_{\text{old}}, \tag{6}$$

```
Given $q$ is a query, $K$ is the number of top initial retrieved documents from which
to collect term correlation and query term weighting, and $M$ (M<=K) is the number of
retrieved documents to be re-ordered in initial retrieval.
Step 1: Acquiring query terms in $q$ and their weights;
    1.1 Extract terms from each document $d$ in top $K$ retrieved documents; in
        practice, term extraction from each document is done only once and this
        process can be considered as a part of indexing.
    1.2 Collect terms that occur in $q$ and calculate their weights by formula (2) and
        (3);
Step 2: Acquiring query term correlation from top $K$ retrieved documents;
        Calculate query term correlation by equation (4);
Step 3: Re-order top $M$ documents;
    3.1. For each document $d_i$ in the $M$ documents, calculate its new ranking value
         $Score_i$ by (5) and (6);
    3.2: Re-order top $M$ retrieved documents by $\{Score_1, \ldots, Score_M\}$.
```

Fig. 1. The procedure of document reranking.

where $\mathrm{Sim}_{\mathrm{old}}$ is the original similarity value between document d and query q in initial retrieval, $w(t_i)$ is the weight of query term t_i.

The top M retrieved documents are re-ordered by their new ranking score $\mathrm{Score}_{\mathrm{new}}$. Fig. 1 gives out the pseudo code of the procedure of document re-ordering for query q and top M retrieved documents.

5. Experiments and evaluation

5.1. Data and design

We use NTCIR-3 CLIR Chinese SLIR document collection as our test dataset to rerank top 1000 retrieved documents. The dataset contains two Chinese document sets, CIRB011 (132,173 documents) and CIRB20 (249,508 documents). We use the officially released 42 Chinese–Chinese D-run query topics, and each query is a short description of a topic in Chinese language. As an example, the following is query topic 001:

⟨TOPIC⟩
⟨NUM⟩001⟨/NUM⟩
⟨DESC⟩
查询故宫博物院所举办之千禧汉代文物大展相关内容(Find information of the exhibition “Art and Culture of the Han Dynasty” in the National Palace Museum)
⟨/DESC⟩
⟨/TOPIC⟩

It is known that a Chinese sentence is a contiguous Chinese character sequence without space between Chinese words. It is suggested that word indexing and bi-gram indexing achieve comparable performance (Nie, Gao, Zhang, & Zhou, 2000). So, for initial retrieval, we use bi-gram as index unit. We use the vector space model and the OKAPI BM25 as retrieval models.

We also use NTCIR-3 CLIR Chinese SLIR's relaxed relevance judgment and rigid relevance judgment to evaluate the performance. Relaxed relevance judgments consider highly relevant, relevant, and partially relevant documents, while rigid relevance judgments only consider highly relevant and relevant documents. We use (relaxed) and (rigid) to represent the relaxed and rigid relevance judgments, respectively. We use Mean Average Precision (MAP) on 42 query topics to measure the overall retrieval performance.

In the vector space model, each document or query is represented as a vector in vector space where each dimension of the vector is a bi-gram. The weight of bi-gram b in document d is given by the following tf · idf weighting scheme:

$$w(b,d) = \log(T(b,d) + 1) \times \log(R/D(b) + 1), \tag{7}$$

where, $w(b,d)$ is the weight given to b in d, $T(b,d)$ is the frequency of b in d, R is the number of documents in document set, $D(b)$ is the number of documents in document set which contain b.

The weight of bi-gram b in query q, $w(b,q)$, is given by the following weight scheme:

$$w(b,q) = T(b,q), \tag{8}$$

where $T(b,q)$ is the frequency of b in q.

The similarities (distance) between a document d and a query q are calculated by the cosine of the document vector and the query vector. For the OKAPI BM25 model, we use the default parameter settings. The initial retrieval results (hereafter INI) under the vector space model and the OKAPI BM25 model are used as baselines in later experiments, respectively.

We will do two kinds of experiments. The first focuses on the performance with various parameter settings for term extraction and various document frequency weighting schemes. The second focuses on the comparison between our method and others.

5.2. Comparison on different parameter setting

Regarding term quality, there are two parameters (δ and L) in our term extraction method, and the following is the parameter setting in our experiments:

$$\delta = 1, 10;\ L = 2, 3, 4.$$

For document frequency weighting scheme, we test the six weighting schemes listed at Table 1.

The comparison of MAPs at different parameters settings under the vector space model is given at Tables 2 and 3. In Tables 2 and 3, each item in table represents the MAP value and its improvement over the baseline (INI) with the conditions expressed by (Column) and (Row). In the following, we use $+x\%$ to denote improvement of $x\%$ over the baseline.

From Tables 2 and 3, we see that the method achieves significant improvement against baseline (INI) in every parameter setting.

If only considering the effectiveness of term frequency to document reranking, ($L = 3$) or ($L = 4$) produce better results. If only considering the effectiveness of document frequency weighting schemes to document reranking, W5, W7 and W9 produce better results.

Table 2
MAPS on the vector space model

	$L = 2$		$L = 3$		$L = 4$	
	$\delta = 1$	$\delta = 10$	$\delta = 1$	$\delta = 10$	$\delta = 1$	$\delta = 10$
INI	0.1688	0. 1688	0. 1688	0. 1688	0. 1688	0. 1688
W4	0.2201 +30.4%	0.2210 +30.9%	0.2282 +35.2%	0.2274 +34.7%	0.229 +35.7%	0.2273 +34.7%
W5	0.2182 +29.3%	0.2214 +31.2%	0.2223 +31.7%	0.2233 +32.3%	0.2263 +34.1%	0.2309 +36.8%
W6	0.2180 +29.1%	0.2222 +31.6%	0.2232 +32.2%	0.2252 +33.4%	0.2261 +33.9%	0.2285 +35.4%
W7	0.2159 +27.9%	0.2209 +30.9%	0.2186 +29.5%	0.222 +31.5%	0.2237 +32.5%	0.2313 +37%
W8	0.2153 +27.5%	0.2147 +27.2%	0.2223 +31.7%	0.2218 +31.4%	0.2239 +32.6%	0.2223 +31.6%
W9	0.2181 +29.2%	0.2214 +31.2%	0.2218 +31.4%	0.2231 +32.2%	0.2236 +32.5%	0.2313 +37%

$\delta = 1$, 10; $L = 2$, 3, 4; rigid relevance.

Table 3
MAPs on the vector space model

	$L = 2$		$L = 3$		$L = 4$	
	$\delta = 1$	$\delta = 10$	$\delta = 1$	$\delta = 10$	$\delta = 1$	$\delta = 10$
INI	0.2197	0. 2197	0. 2197	0. 2197	0. 2197	0. 2197
W4	0.2723 +23.9%	0.2733 +24.4%	0.2812 +28%	0.2798 +27.4%	0.2834 +29%	0.2828 +28.7%
W5	0.2700 +22.9%	0.2713 +23.5%	0.2773 +26.2%	0.2777 +26.4%	0.2799 +27.4%	0.2849 +29.7%
W6	0.2702 +23%	0.2740 +24.7%	0.277 +26.1%	0.279 +27%	0.2789 +26.9%	0.2838 +29.2%
W7	+0.2675 +21.8%	0.2714 +23.5%	0.2746 +25%	0.2753 +25.3%	0.2786 +26.8%	0.2855 +29.9%
W8	0.2690 +22.4%	0.2682 +22.1%	0.2764 +25.8%	0.2762 +25.7%	0.2798 +27.4%	0.2775 +26.3%
W9	0.2696 +22.7%	0.2714 +23.5%	0.2772 +26.2%	0.2777 +26.4%	0.2788 +26.9%	0.2854 +29.9%

$\delta = 1$, 10; $L = 2$, 3, 4; relaxed relevance.

If considering both term frequency and document frequency weighting, the parameter setting ($L = 3$ or $L = 4$, W5, W7 or W9) produces better results. Under such parameter settings, our method achieves 25%–29.9% improvement for MAP(relaxed), and 29.5%–37% improvement for MAP(rigid).

To explore the co-effects of the two parameters (L, δ) on the precision, we fix one parameter and see how the precision changes with the other.

One setting is that we fix L as 2, 3 or 4 and with δ changing from 1 to 10. One finding is that the MAP improves or keeps the same in most cases, while decreases in fewer cases. The reason is that the terms with lower salience seeds tend to be noises, and removing the noises leads to improvement of the precision. However, not all relevant terms do hold higher salience seeds, in addition, some documents, although containing good terms, but they are not relevant (due to different focus). However, this chance is rare, so in fewer cases, we can see that the precision decreases.

Another setting is that we fix δ as 1 or 10 and with L changing from 2 to 3. It demonstrates that whether $\delta = 1$ or 10, all the precision improves when L increases to 3 from 2. This means that when $L = 2$, there may be too many noisy terms, and when $L = 3$, some noisy terms can be removed. That is why all the precision improves.

Another setting is that we fix δ as 1 or 10 and with L changing from 3 to 4. It demonstrates that whether $\delta = 1$ or 10, all the precision improves when L increases to 4 from 3. This means that when $L = 3$, there may be still some noisy terms, and when $L = 4$, more noisy terms can be removed.

Regarding the effect of document ranking positions, it is noticed that with scheme W5, W7 or W9, it tends to get higher performance, while with scheme W8, it tends to get lower performance. The reason is that not all documents with top ranking are relevant in most cases. In particular, for the first retrieval, among the top 10 documents, there are only 3.6 relevant documents in average, while among the top 100 documents, there are 18.9 relevant documents in average. This means that many relevant documents are located outside the top 10, but within the top 100 in the first retrieval. With W5, W9 and W7, the terms in these documents get higher weights, and then the documents tend to move forward during the reranking process. On the contrary, with W8, the weights of the terms decrease dramatically as the rank goes down, and the terms in lower ranking documents get very lower weights. So the relevant documents containing the terms cannot move forward during the reranking.

To see whether the MAP difference between reranking and initial search (baseline) is significant, we conducted the paired t-tests. In our experiments, MAPs of the 42 topics are regarded as sampled observations. Tables 4–7 list the p-values and their significance (under sign) for each pair of reranking and initial search.

Table 4
Paired *t*-test results on the vector space model

	$L = 2$		$L = 3$		$L = 4$	
	p-value	Sign.	*p*-value	Sign.	*p*-value	Sign.
W4	2.0623e−4	**	4.9007e−5	**	1.1065e−4	**
W5	1.7622e−4	**	1.2800e−4	**	7.6106e−5	**
W6	1.9036e−4	**	1.0064e−4	**	7.9149e−5	**
W7	3.3612e−4	**	4.1109e−4	**	8.3740e−5	**
W8	2.0634e−4	**	1.3518e−4	**	1.7695e−4	**
W9	1.7396e−4	**	1.1533e−4	**	9.3889e−5	**

$\delta = 1$; $L = 2$, 3, 4; rigid relevance.

Table 5
Paired *t*-test results on the vector space model

	$L = 2$		$L = 3$		$L = 4$	
	p-value	Sign.	*p*-value	Sign.	*p*-value	Sign.
W4	1.4907e−5	**	3.4839e−6	**	6.4448e−6	**
W5	5.6399e−6	**	5.0154e−6	**	3.8164e−6	**
W6	6.0715e−6	**	7.6713e−6	**	4.9134e−6	**
W7	1.6625e−5	**	1.0124e−5	**	3.3928e−6	**
W8	5.9705e−5	**	1.5246e−5	**	1.9163e−5	**
W9	5.5763e−6	**	3.9991e−5	**	3.6491e−6	**

$\delta = 1$; $L = 2$, 3, 4; relaxed relevance.

Table 6
Paired *t*-test results on the vector space model

	$L = 2$		$L = 3$		$L = 4$	
	p-value	Sign.	*p*-value	Sign.	*p*-value	Sign.
W4	2.1925e−4	**	7.2441e−5	**	1.9945e−4	**
W5	1.6007e−4	**	1.6356e−4	**	1.3081e−4	**
W6	1.8714e−4	**	1.0346e−4	**	1.5434e−4	**
W7	2.0560e−4	**	2.6117e−4	**	1.3040e−4	**
W8	2.4788e−4	**	1.8013e−4	**	2.9201e−4	**
W9	1.4644e−4	**	1.6801e−4	**	1.3081e−4	**

$\delta = 10$; $L = 2$, 3, 4; rigid relevance.

Table 7
Paired *t*-test Results on the vector space model

	$L = 2$		$L = 3$		$L = 4$	
	p-value	Sign.	*p*-value	Sign.	*p*-value	Sign.
W4	2.1651e−5	**	1.0905e−5	**	1.5435e−5	**
W5	2.6774e−5	**	1.3666e−5	**	1.1425e−5	**
W6	1.5914e−5	**	1.0202e−5	**	1.1529e−5	**
W7	3.0718e−5	**	3.4905e−5	**	1.0632e−5	**
W8	7.5830e−5	**	2.7544e−5	**	3.3322e−5	**
W9	2.3200e−5	**	1.2656e−5	**	1.1425e−5	**

$\delta = 10$; $L = 2$, 3, 4; relaxed relevance.

In following tables, $**, *$ and $\sim$ correspond to the *p*-value < 0.001, *p*-value $\leqslant 0.05$ and *p*-value > 0.05, which means that the difference is, respectively, strong significant, significant or not significant.

From Tables 4–7, we can see that for each parameter setting, document reranking can significantly improve the performance for both the rigid and relaxed relevance under the vector space model.

5.3. Comparison on MMR and non-MMR

To explore the impact of MMR in the reranking, Tables 8 and 9 show the comparison of MAPs with/without MMR under the vector space model and the p-values.

From the comparison, we can see that the MMR module helped to improve the performance by 8%–17.3% under the vector space model, which indicates that the correlation between query terms is useful for improvement of the performance.

5.4. Comparison with other document reranking methods

We first compare our method with Mitra et al.'s method (Mitra et al., 1998). Mitra et al. (1998) uses term correlation to re-order retrieved documents. If $\{w_1,\ldots,w_m\}$ is the set of query words presented in document d (ordered by decreasing idf), then the new ranking score between q and d is calculated by following formula:

$$\mathrm{Sim}_{\mathrm{new}} = \mathrm{idf}(w_1) + \sum_{i=2}^{m} \mathrm{idf}(w_i) \times \min_{j=1}^{i-1}(1 - P(w_i|w_j)), \tag{9}$$

where idf(w_i) is the inverse document frequency of word w_i in retrieved documents to be reranked, $P(w_i|w_j)$ is the word correlation between w_i and w_j in top K retrieved documents calculated by the same formula (4).

Figs. 2 and 3 show the comparison of performance between Mitra's and our method $\delta = 10$; $L = 4$; W5 under vector space model on MAP(rigid) and MAP(relaxed). In the experiments, we set K as 1000 and rerank top 50, 100, 200, 300, 400, 500, 600,700, 800, 900 and 1000 documents, respectively.

In Figs. 2 and 3, MMR represents the performance of our method, INI refers the initial results (baseline), and Mitra represents the performance of Mitra's method.

Table 8
Comparison on the vector space model

	MAP		t-test (MMR versus non-MMR)	
	Non-MMR	MMR	p-value	Sign.
W4	0.2023 + 19.8%	0.2273 + 34.7%	2.8869e−3	*
W5	0.2172 + 28.7%	0.2309 + 36.8%	1.1883e−2	*
W6	0.2038 + 20.7%	0.2285 + 35.4%	2.9337e−3	*
W7	0.2162 + 28.1%	0.2313 + 37.0%	1.1934e−2	*
W8	0.1921 + 13.8%	0.2213 + 31.1%	3.0425e−3	*
W9	0.2164 + 28.2%	0.2313 + 37%	1.1162e−2	*

$\delta = 10$; $L = 4$; rigid relevance.

Table 9
Comparison on the vector space model

	MAP		t-test (MMR versus non-MMR)	
	Non-MMR	MMR	p-value	Sign.
W4	0.2545 + 15.8%	0.2828 + 28.7%	5.5169e−4	**
W5	0.2690 + 22.4%	0.2849 + 29.7%	1.6061e−3	*
W6	0.2555 + 16.3%	0.2838 + 29.2%	2.5677e−4	**
W7	0.2679 + 21.9%	0.2855 + 29.9%	1.3747e−3	*
W8	0.2444 + 11.2%	0.2775 + 26.3%	5.2614e−4	**
W9	0.2683 + 22.1%	0.2854 + 29.9%	1.4587e−3	*

$\delta = 10$; $L = 4$; relaxed relevance.

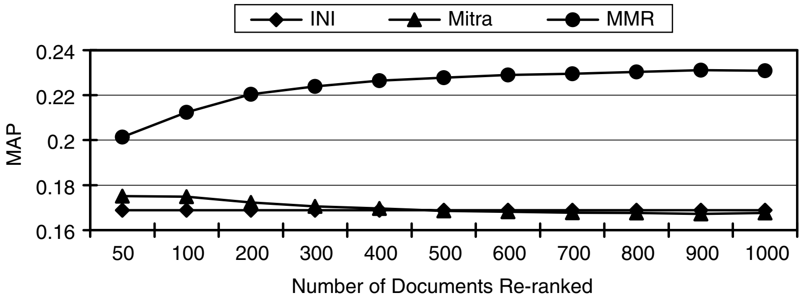

Fig. 2. Comparison on MAP (rigid relevance).

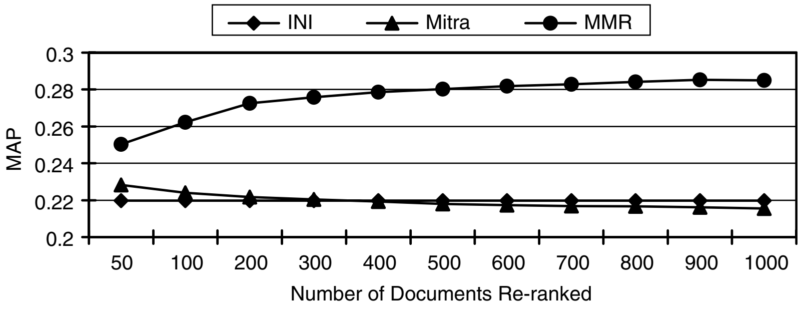

Fig. 3. Comparison on MAP (relaxed relevance).

From Figs. 2 and 3, we can see that our method achieves better performance than that of Mitra's for both MAP(rigid) and MAP(relaxed) consistently at every document number setting. For example, when reordering top 50, 100 or 1000 documents, Mitra(relax) is 0.2283, 0.224 and 0.2218, respectively, while our MMR(relax) is 0.2504, 0.2623 and 0.2725, respectively.

On the other hand, for our method, the improvement keeps or increases in a stable way as the number of documents to be reranked increases, while for Mitra's method, the improvement generally decreases as the document number increases. For example, Mitra(rigid) decreases from 0.1751 to 0.1749 and 0.1676 as document number increase from 50 to 100 and 1000, while our MMR(rigid) increases from 0.2014 to 0.2124 and 0.2309.

Another finding is that Mitra's method is only applicable to top (50–100) ranking documents, as is claimed in Mitra's paper, while our method is more robust and applicable to both smaller and larger scope of documents.

The reason for these findings is that our weighting scheme makes it possible to make use of the information of a larger scope of the retrieved documents, while resisting the impact of noisy documents by assigning lower weights for terms in lower ranking documents. In contrast, the idf-based weighting in Mitra's method assigns lower scores to some really relevant terms, and subjects to the noisy terms within a larger range of the documents.

We also compare our method with Yang et al. (2005), where a smaller top N (20, 25 or 30) documents are considered as relevance feedback. The comparison of MAPs under the vector space model is given at Table 10. In Table 10, MMR represents our proposed document reranking method at parameter setting ($K = 1000$; $L = 4$; $\delta = 10$).

From Table 10, we see that our method achieves better result with more than 10% improvement against Yang's method on both MAP(rigid) and MAP(relaxed). One possibility is that Yang's method only uses information in top 20–30 documents while we use information in top 1000 documents. When there are fewer relevant documents falling in top 20–30 documents, their method cannot capture enough information for reranking.

Table 10
Comparison: Rerank top 1000 documents with $L = 4$ and $\delta = 10$

	MAP(rigid)	MAP(relax)
INI	0.1688	0.2197
Yang ($N = 20$)	0.2085 + 23.5%	0.2624 + 19.4%
Yang ($N = 25$)	0.2072 + 22.7%	0.2603 + 18.5%
Yang ($N = 30$)	0.2057 + 21.9%	0.2591 + 17.9%
MMR	0.2313 + 37%	0.2854 + 29.9%

To see whether the MAP difference between our method and Yang et al.'s is significant, we conducted the paired *t*-tests. Table 11 lists the *p*-values.

From Table 11, we can see that the difference is significant for both rigid and relaxed relevance under the vector space model.

5.5. Experiments on the Okapi BM25 model

We also do experiments on the OKAPI BM25 model and use the default parameter setting. The comparison is given in Table 12. From Table 12, we see that the method achieves 18.9%–21% improvement against (INI) at MAP(rigid) and achieves 13.7%–15% improvement against (INI) at MAP(relaxed).

To see whether the MAP difference between reranking and initial search under the OKAPI BM25 model is significant, we conducted the paired *t*-tests. Table 13 lists the *p*-values and their significance for each pair of reranking and initial search.

Table 11
Paired *t*-test Results between our and Yang et al.'s method

	MAP(rigid)		MAP(relaxed)	
	p-value	Sign.	*p*-value	Sign.
Yang ($N = 20$)	4.5125e−2	*	4.2584e−2	*
Yang ($N = 25$)	3.4629e−2	*	2.2859e−2	*
Yang ($N = 30$)	2.9355e−2	*	1.6272e−2	*

Table 12
MAP on ($\delta = 10$; $L = 4$; OKAPI BM25 model)

	INI	W4	W5	W6	W7	W8	W9
MAP(rigid)	0.1899	0.2286 +20.4%	0.2298 +21%	0.2278 +20%	0.2296 +20.9%	0.2257 +18.9%	0.2298 +21%
MAP(relaxed)	0.2363	0.2717 +15%	0.2714 +14.9%	0.2696 +14.1%	0.2699 +14.2%	0.2686 +13.7%	0.2700 +14.3%

Table 13
Paired *t*-test results on ($\delta = 10$; $L = 4$; OKAPI BM25 model)

	Rigid relevance		Relaxed relevance	
	p-value	Sign.	*p*-value	Sign.
W4	4.6829e−3	*	4.3986e−2	*
W5	5.9408e−3	*	1.9589e−2	*
W6	6.2873e−3	*	4.9454e−2	*
W7	6.4251e−3	*	1.8269e−2	*
W8	5.9348e−3	*	4.7379e−2	*
W9	6.1871e−3	*	1.8379e−2	*

From Table 13, we can see that for parameter setting $\delta = 10$, ($L = 4$), document reranking can significantly improve the performance for both rigid and relaxed relevance under the OKAPI BM25 model. Similar results are also found for other parameter settings under the OKAPI BM25 model.

6. Conclusion and future work

In this paper, we propose a new term weighting scheme and use it in document reranking. The weighting scheme for terms is based on their local and global distribution in top retrieved documents and the whole document set respectively, which combines the information regarding relative document frequency, document ranking positions as well as term length. The scheme allows randomly setting a larger portion of documents as relevance feedback, and helps to improve the performance of MMR model in document reranking.

Our experiments based on NTCIR-3 CLIR Chinese SLIR task show that our proposed approach achieves significant improvement against the baseline. Compared with other document reranking methods, our method also gets higher performance on NTCIR-3 CLIR Chinese SLIR document collection. Furthermore, the performance of the approach generally improves or keeps as the number of the reranking documents increases, which shows that it is robust against the noisy documents included.

The experimental results support our assumptions: key terms in top K retrieved documents can be used to improve precision; long key term may contain more precise information and can be used to improve precision; document frequency distribution of query term in top K retrieved documents against the whole retrieved document set implies the importance of query term.

As the basis of this method, the term extraction module is very simple, which is a purely statistical method. In future, we will also consider more effective approaches for term extraction.

Our experiments are all based on Chinese information retrieval. In fact, our method is language independent. In the future, we will do further tests on other languages.

References

Balinski, J., & Danilowicz, C. (2005). Re-ranking method based on inter-document distance. *Information Processing and Management, 41*, 759–775.

Bear J., Israel, D., Petit J., & Martin D. (1997). Using Information Extraction to Improve Document Retrieval. In *Proceedings of the Sixth Text Retrieval Conference*.

Crouch, C., Crouch, D., Chen, Q., & Holtz, S. (2002). Improving the retrieval effectiveness of very short queries. *Information Processing and Management*, 38.

Kamps, J., (2004). Improving Retrieval Effectiveness by Reranking Documents Based on Controlled Vocabulary. In *Proceedings of the 21th European Conference on Information Retrieval*.

Lee, K., Park, Y., & Choi, K. S. (2001). Document re-ranking model using clusters. *Information Processing and Management, 37*(1), 1–14.

Luk, R.W.P. & Wong, K.F. (2004). Pseudo-Relevance Feedback and Title Re-Ranking for Chinese IR. In *Proceedings of NTCIR Workshop 4*.

Mitra, M., Singhal, A., & Buckley, C. (1998). Improving Automatic Query Expansion. In *Proceedings of ACM SIGIR'98*.

Nie, J.Y., Gao, J., Zhang, J., & Zhou, M. (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.

Qu, Y.L., Xu, G.W., & Wang, J., (2000). Rerank Method Based on Individual Thesaurus. In *Proceedings of NTCIR2 Workshop*.

Schutze, H., (1998). The Hypertext Concordance: A Better Back-of-the-Book Index. In *Proceedings of First Workshop on Computational Terminology*, pp. 101–104.

Xu, J., & Croft, W.B. (1996). Query Expansion Using Local and Global Document Analysis. In *Proceedings of ACM SIGIR'96*.

Xu, J., & Croft, W. B. (2000). Improving the effectiveness of information retrieval with local context analysis. *ACM Transactions on Information Systems, 18*(1), 79–112.

Yang, L.P., Ji, D.H., & Tang, L. (2004). Document Re-ranking Based on Automatically Acquired Key Terms in Chinese Information Retrieval. In *Proceedings of 20th International Conference on Computational Linguistics (COLING)*.

Yang, L.P., Ji, D.H., Zhou, G.D., & Nie, Y., (2005). Improving Retrieval Effectiveness by Using Key Terms in Top Retrieved Documents. In *Proceedings of 27th European Conference on Information Retrieval*.