Document Retrieval in Consideration of the Amount of Term Frequencies

Hiroshi UMEMOTO, Tadanobu MIYAUCHI and Yoshihiro UEDA
Fuji Xerox Co., Ltd.
430 Sakai, Nakai-machi, Ashigarakami-gun, Kanagawa 259-0157, Japan
{Hiroshi.Umemoto,Tadanobu.Miyauchi, Ueda.Yoshihiro}@fujixerox.co.jp

Abstract

We propose a document retrieval that evaluates the degree of similarity between a query and a document in consideration of not only term-weights but also the amount of term frequencies. Different from tf-idf term-weighting schemes, the proposed scheme never reflects a term frequency in calculating the term-weight.

We carried out an experiment in retrieval performance evaluation using a subset of NTCIR-1. It turned out that appropriate parameters of calculating the similarity are depend on the number of query terms and that the proposed scheme is superior to well-known tf-idf schemes in retrieval performance.

Keywords: *term-weight, degree of similarity, tf-idf*

1 Introduction

Term frequency and inverse document frequency (tf-idf) term-weighting scheme is popular in document retrieval systems based on the vector space model. However, well-known tf-idf schemes tend to overestimate terms that appear in only few documents stored in a document collection, and this tendency is one of the serious factors that decrease the retrieval performance.

Thus, the values of term-weights based on tf-idf schemes are not always appropriate in document retrieval. While, in order to estimate the similarity between a query and a document, there are various indicators of the similarity except for term-weights. One of them is the number of terms that appear in both a query and a document mutually. We consider that the similarity based on the number of the mutual terms complements the retrieval performance.

In this paper, we propose a relevant document retrieval system that utilizes not a term frequency inside a document in calculating the weight of the term but the amount of term frequencies inside a query and a document in calculating a similarity between them. In section 2, we describe the details of the system.

The proposed similarity calculus is parameterized and the parameters can be tuned for each retrieval situation. In order to tune the parameters, we have carried out an experiment in retrieval performance evaluation using a subset of NTCIR-1. And we have compared the proposed scheme with well-known tf-idf schemes. In section 3, we describe the conditions and the results of the experiment.

2 Description of the proposed system

The proposed retrieval system is based on the vector model. Retrieval queries and documents of a collection are mapped into the vector space that is indexed by terms. When retrieving relevant documents by a query, similarities associated with a pair of the query and each document are calculated.

The major differences between the proposed system and general systems based on the vector model are:

1. similarity-calculation scheme between a query and a document
2. term-weighting scheme

The proposed system utilizes neither syntactic nor semantic information, nor query expansion by thesaurus or relevance feedback.

2.1 Index terms in the vector space

We represent documents and queries by the sets of index terms in the vector space. Index terms are generated from the result of a morphological analysis with a Japanese finite state transducer.

We use content words as index terms in the vector space, which are nouns, verbs, adjectives and adverbs. In addition to those words, compound nouns, which consist of two continuous nouns, are also used as index terms.

2.2 Term-weighting scheme

We define a term-weight as the ratio of the document frequency (df) of the term in the query and the df in the document repository. This definition is based on the concept of mutual information.

And if a term is either non-noun or compound noun or numeric nouns or prefixed or suffixed, then we decrease its term-weight. We define "linguistic parameter" as the decreasing rate of the term-weight according to the linguistic information of terms.

A term-weight formula is defined as follows:

$$w_i = L_i \cdot df_{qi}^{\ 2} / df_i$$

where L_i is the linguistic parameter of the i-th term, df_{qi} and df_i are the document frequencies of term T in the query and in the repository respectively.

2.3 Similarity-calculation scheme

We define the similarity between a query and a document as the sum of two functions, one is a function of term-weights, and the other is a function of the amount of term frequencies. The degree of similarity formula Sim between a query Q and a document D is defined as follows:

$$Sim(Q, D) = \alpha \sum_{i=1}^{N} w(T(Q, D, i)) + \beta \sum_{i=1}^{N} \{tf_q(T(Q, D, i)) + tf_d(T(Q, D, i))\}$$

where α and β are coefficients, N is the number of index terms that appear both in query Q and document D mutually, $w(T)$ is the term-weight of index term T, $T(Q, D, i)$ is the i-th term that appears both in query Q and document D, $tf_q(T)$ and $tf_d(T)$ are the term frequencies of index term T in the query and in the document respectively.

3 Retrieval performance evaluation using a subset of NTCIR-1

In the proposed system, the parameters α and β in the degree of similarity formula affect the retrieval performance. In order to tune the parameters, we carried out an experiment in retrieval performance evaluation using a subset of NTCIR-1. And we compared the proposed scheme with tf-idf term-weighting scheme in retrieval performance.

3.1 A set of documents

A part of the NTCIR-1 was used as the document set. We selected all relevant documents to the all-83 topics plus randomly sampled non-relevant documents in NTCIR-1. The total number of documents is about 30,000.

3.2 Queries

For each topic, we used following 4 queries:

1. description of the topic in NTCIR-1 that is shown in the <DESCIRPTION> field
2. Top ranked relevant document retrieved by the query 1.
3. Top 3 ranked relevant documents retrieved by the query 1.
4. Top 5 ranked relevant documents retrieved by the query 1.

3.3 Parameters in the degree of similarity formula

In order to tune the parameters α and β, we evaluated retrieval performances under the condition that the parameter α is fixed and the parameter β is assigned to 16 different values ranged from 0 to 10000. Then we classified the retrieval results into 7 groups according to the numbers of query index terms. Average numbers of query index terms in each group are shown in Table 1. Relations between the parameter β and the retrieval performance of each group are shown in Table 2, Table 3, Figure 1 and Figure 2.

We can see that β should be small when the number of query index terms q is small and it should be larger as q increases. From this analysis, we define the relation between β and q as the following the formula:

if q is in the rage $(0, q_0)$ *then* β_0
if q is in the rage (q_0, q_1) *then* $\gamma(q - q_0) + q_0$
if q is in the rage (q_1, ∞) *then* β_1

where α is 100,000, β_0 is 20, γ is 3.5, β_1 is 2,000, q_0 is 20, and $q_1$is 500. This approximated relation is shown in Figure 3.

group 1	group 2	group 3	group 4	group 5	group 6	group 7
10.9	72.0	146.5	247.4	340.3	432.0	737.0

Table 1. Average number of query index terms in each group

β	0	5	10	15	20	30	40	50
gourp 1	0.3990	0.4263	0.4337	0.4357	0.4370	0.4345	0.4312	0.4269

β	100	200	500	1,000
gourp 1	0.4100	0.3865	0.3545	0.3222

Table 2. Parameter β and 11-points average precision in group 1

β	group 2	group 3	group 4	group 5	group 6	group 7
0	0.1718	0.2439	0.3706	0.4503	0.3246	0.4103
10	0.1801	0.2501	0.3744	0.4545	0.3304	0.4185
50	0.1837	0.2664	0.3872	0.4680	0.3535	0.4579
100	0.1870	0.2747	0.3978	0.4788	0.3845	0.4908
200	0.1898	0.2853	0.4124	0.4912	0.4125	0.5265
300	0.1880	0.2947	0.4209	0.5020	0.4296	0.5476
500	0.1872	0.2962	0.4270	0.5085	0.4529	0.5654
600	0.1862	0.2977	0.4285	0.5110	0.4632	0.5744
800	0.1871	0.2964	0.4295	0.5120	0.4745	0.5884
1,000	0.1852	0.2951	0.4273	0.5118	0.4839	0.5947
1,200	0.1840	0.2925	0.4250	0.5096	0.4888	0.5985
1,500	0.1816	0.2913	0.4233	0.5069	0.4931	0.6027
2,000	0.1810	0.2893	0.4182	0.5003	0.4999	0.6088
5,000	0.1767	0.2821	0.4060	0.4852	0.5002	0.6208
8,000	0.1760	0.2813	0.4007	0.4780	0.4956	0.6229
10,000	0.1752	0.2806	0.3985	0.4739	0.4914	0.6236

Table 3. Parameter β and 11-points average precision in group 2, .., 7

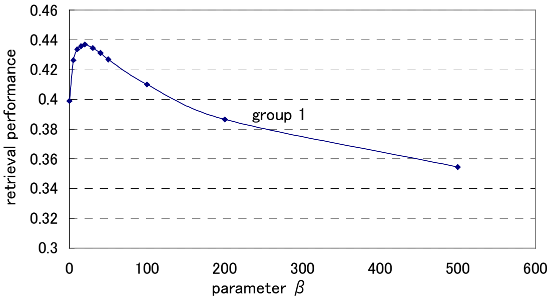

Figure 1. Parameter β and 11-points average precision in group 1

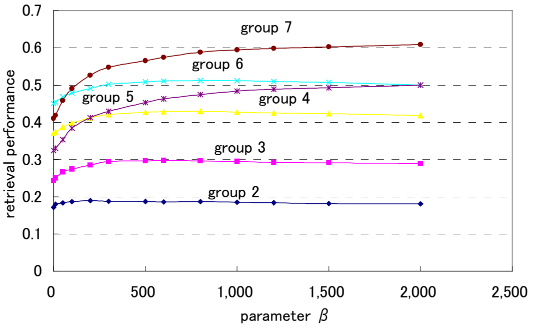

Figure 2. Parameter β and 11-points average precision in group 2, .., 7

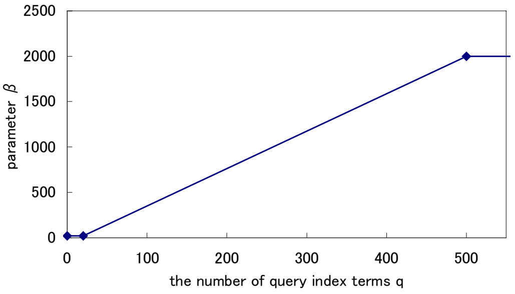

Figure 3. Approximated relation between parameter β and the number of query terms *q*

3.4 Results of retrieval performance evaluation

The results of retrieval performance evaluation are shown in Table 4, which was done with the set of documents and the queries that are shown in the <DESCRIPTION> fields. Retrieval performance is represented by the average precision over all queries. We have obtained the results by 4 different schemes. And we introduced linguistic parameters to every scheme in order to compare each scheme with the others in the same condition.

(1) tf-idf scheme

The similarity *Sim(Q, D)* between query *Q* and document *D* is defined as follows:

$$Sim(Q, D) = \mathbf{w}_q \cdot \mathbf{w}_d$$

where $\mathbf{w}_q$ is the term-weight vector of query *Q*, and $\mathbf{w}_d$ is that of document *D*. Each element of the term-weight vector is defined as follows:

$$w_i = L_i \cdot tf(T_i) \cdot log(Nd / df(T_i))$$

where L_i is the linguistic parameter of the i-th term, *tf(T)* is the term frequency of term *T*, T_i is the i-th term, *Nd* is the number of documents in the repository, and *df(T)* is the document frequency of term *T* in the repository.

(2) square root of tf and idf

Each element of the term-weight vector is defined as follows:

$$w_i = L_i \cdot tf^{1/2}(T_i)\ log(Nd / df(T_i))$$

(3) idf only

Each element of the term-weight vector is defined as follows:

$$w_i = L_i \cdot log(Nd/df(T_i))$$

(4) proposed method

1. tf–idf	2. root tf idf	3. idf	4. proposed
0.3023	0.4276	0.4281	0.4524

Table 4. Retrieval performance of each scheme

4 Conclusion

We proposed a relevant document retrieval system that utilizes not only term-weight but also the amount of term frequencies.

We carried out an experiment in order to evaluate the retrieval performance of the proposed system using a subset of NTCIR-1. It turned out that appropriate parameters of the similarity calculation are depend on the number of query terms and the proposed scheme is superior to well-known tf-idf schemes in the retrieval performance.

References

[1] R. Baeza-Yates, B. Ribeiro-Neto, *Modern information retrieval*, ACM press, 1999.

[2] T. Hisamitsu, Y. Niwa, J. Tsujii, "A method of measuring term representativeness -baseline method using co-occurrence distribution-", Proc. COLLING 2000.

[3] M. Tateno, H. Masuichi, H. Umemoto, "The Japanese lexical transducer based on stem-suffix style forms", *Extended finite state models of language*, Cambridge university press, pp.48-55, 1999.

[4] H. Umemoto, T. Kuramochi, Y. Ishitobi, M. Tateno, "Development of a related document retrieval system and evaluation of the system using NTCIR-1", Proc. NTCIR workshop 1, pp.183-185, 1999.