Applying Multiple Characteristics and Techniques to Obtain High Levels of Performance in Information Retrieval

Masaki Murata, Qing Ma, and Hitoshi Isahara
Communications Research Laboratory
2-2-2 Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-0289, Japan
{murata,qma,isahara}@crl.go.jp

Abstract

Our information retrieval system which achieves its goals by taking advantage of numerous characteristics of the information and applying numerous sophisticated techniques is described. Robertson's 2-Poisson model and Rocchio's formula, both of which are known to be effective, have been applied in the system. Characteristics of newspapers such as locational information were applied. We give examples of this method's effectiveness in retrieval from collections of newspaper articles, such as the document set for NTCIR 3. We present our application of Fujita's method, where longer terms are used in retrieval by the system but de-emphasized relative to the emphasis on the shortest terms; this allows us to use both compound and single-word terms. The statistical test used in expanding queries through an automatic feedback process is described. The method gives us terms which have been statistically confirmed to be related to the top-ranked documents that were obtained in the first retrieval. We describe the success of the system in four tasks (Korean, Japanese, English, and Chinese) of monolingual information retrieval at NTCIR 3; i.e., the highest scores for precision on all tasks, except for average precision on the "rigid" CC task, where its score was second highest. In terms of the other evaluated measures and monolingual information retrieval in other languages, the system obtained both the best average precision and the best R-precision.

Keywords: *Monolingual IR, High Perfomance, Locational information, De-emphasis of longer terms, Statistical test*

1 Introduction

In NTCIR-3, we used our exsiting system (System-A of NTCIR-2) which achieves its goals by taking advantage of numerous characteristics of the information and applying numerous sophisticated techniques. Robertson's 2-Poisson model and Rocchio's formula, both of which are known to be very effective, have been applied in the system. We used such characteristics of newspapers as locational information. This method is very effective in retrieval from collections of newspaper articles, such as the document set for NTCIR 3. We applied Fujita's method, where longer terms are used in retrieval by the system but are assigned lower weights than the shortest terms; this allows us to use compound terms as well as single-word terms. We also used a statistical test in expanding queries through an automatic feedback process. This method gives us terms which have been statistically confirmed to be related to the top-ranked documents that were obtained in the first retrieval. We applied the system to the four tasks of monolingual information retrieval at NTCIR 3, referred to as JJ, CC, KK, and EE.[1] Our system obtained the highest scores for precision on all tasks, except for average precision on the "rigid" CC task, where its score was second highest. In terms of the other evaluated measures and monolingual information retrieval in other languages, our system obtained both the best average precision and the best R-precision. This paper gives a detailed description of our system, which showed a high level of performance at NTCIR 3.

2 Outline of our system

Our system uses Robertson's 2-Poisson model[7], which is a probabilistic approach. In Robertson's method, each document's score is calculated by using the following equation.[2] The documents that obtain high scores are then output as the results of retrieval. $Score(d, q)$ below is the score of a document d against

[1] CC means Chinese monolingual information retrieval, means Japanese monolingual information retrieval, KK means Korean monolingual information retrieval, and EE means English monolingual information retrieval. JJ, CC, and KK were official tasks in NTCIR-3 and EE was a unofficial task in NTCIR-3.

[2] This equation is BM11, which corresponds to BM25 in the case where $b = 1$ [8].

a query q.

$$Score(d,q) = \sum_{\substack{\text{term } t \\ \text{in } q}} \left(\frac{tf(d,t)}{tf(d,t) + k_t \frac{length(d)}{\Delta}} \times log\frac{N}{df(t)} \times \frac{tf_q(q,t)}{tf_q(q,t) + kq} \right) \quad (1)$$

where t indicates a term that appears in a query. $tf(d,t)$ is the frequency of t in a document d, $tf_q(q,t)$ is the frequency of t in a query q, $df(t)$ is the number of the documents in which t appears, and N is the total number of documents, $length(d)$ is the length of a document d, and Δ is the average length of the documents. k_t and k_q are constants which are set according to the results of experiments.

In this equation, we call $\frac{tf(d,t)}{tf(d,t) + k_t \frac{length(d)}{\Delta}}$ the TF term, (abbr. $TF(d,t)$), $log\frac{N}{df(t)}$ the IDF term, (abbr. $IDF(t)$), and $\frac{tf_q(q,t)}{tf_q(q,t)+kq}$ the TF_q term (abbr. $TF_q(q,t)$).

In our system, several terms are added to extend this equation, and the method for doing this is expressed by the following equation.

$$Score(d.q) = \left\{ \sum_{\substack{\text{term } t \\ \text{in } q}} (TF(d,t) \times IDF(t) \times TF_q(q,t) \times K_{location}(d,t) \times K_{detail} \times \left(log\frac{Nq}{qf(t)} \right)^{k_{Nq}} \right) + \frac{length(d)}{length(d) + \Delta} \right\} \quad (2)$$

The TF, IDF and TF_q terms in this equation are identical to those in Eq. (1). The value of the term $\frac{length}{length+\Delta}$ increases with the length of the document. This term is introduced because, in a case where all of the other information is exactly the same, a longer document is more likely to include content that is relevant as a response to the query. Nq is the total number of queries and $qf(t)$ is the number of queries in which t occurs. Those terms which occur more frequently in queries are more likely to be such as 文書 "document" and もの "thing". We use $log\frac{Nq}{qf(t)}$ to decrease the scores for stop words. $K_{location}$ and K_{detail} are extended numerical terms that are introduced to improve the precision of results. $K_{location}$ uses the location of the term within the document. If the term is in the title or at the beginning of the body of the document, it is given a higher weighting. K_{detail} uses the information such as whether the term is a proper noun and or a stop word. In the next section, we explain these extended numerical terms in detail.

3 Extended numerical terms

We use the two extended numerical terms $K_{location}$ and K_{detail} in Eq. (2). In this section, they are explained in detail.

1. Locational information ($K_{location}$)[3]

   The title or first sentence of the body of a document in a newspaper will generally indicates the subject. Precision in information retrieval can thus be improved by assigning greater weight to terms from these locations. This is achieved by $K_{location}$, which is used to adjust the weight of a term according to whether or not it appears at the beginning of a document. A term in the title or at the beginning of the body of a document, is assigned a higher weight. A term elsewhere is given a lower weight. $K_{location}$ is expressed as follows:

$$K_{location}(d,t) = \begin{cases} k_{location,1} & \text{(when a term } t \text{ occurs in the title of a document } d\text{)}, \\ 1 + k_{location,2} \frac{(length(d) - 2 * P(d,t))}{length(d)} & \text{(otherwise)} \end{cases} \quad (3)$$

   $P(d,t)$ is the location of a term t in the document d. When a term appears more than once in a document, the location in which it first appears is used to set this parameter. $k_{location,1}$ and $k_{location,2}$ are constants to which values are assigned according to the results of experiments.

2. Other information (K_{detail})

   K_{detail} is a more detailed numerical term that uses different information, such as whether or not a term is a proper noun and whether or not it is a stop word such as 文書 "document" and もの "thing". If a term is a proper noun, it is assigned a high weight. If a term is a stop word, it is assigned a low weight. K_{detail} is expressed in the following way for simplicity; the variables for the document and term, d and t, have been omitted:

$$K_{detail} = K_{descr} \times K_{proper} \times K_{num} \quad (4)$$

   The respective terms in this equation are explained below.

   - K_{descr}

     When a term is obtained from the title of a query, i.e. DESCRIPTION, K_{descr} =

[3] This method was developed by Murata et. al. [4].

$k_{descr}(\geq 1)$. Otherwise, K_{descr} = 1. This is because we can assume that terms obtained from the description of the query are important.

- K_{proper}
  When a term is a proper noun, K_{proper} = $k_{proper}(\geq 1)$. Otherwise K_{proper} = 1. This is because terms that are proper nouns are important.
- K_{num}
  When a term is numeric, K_{num} = $k_{num}(\leq 1)$. Otherwise, K_{num} = 1. A term which consists solely of numerals will not contain much relevant information, and thus lacks importance for the query.

4 How terms are extracted

We are only able to use Eq. (2) in information retrieval after we have extracted terms from the query. This section describes how this is achieved. We considered the several methods of term extraction listed below.

1. Using only the shortest terms

   This is the simplest method. In this method, the query sentence is divided into short terms by using a morphological analyzer or similar tool. All of the short terms are used in the retrieval process. The method used to divide the query sentence into short terms is described in Section 5.

2. Using all term patterns

   The first method produces terms that are too short. For example, ”enterprise” and ”amalgamation” would be used separately while “enterprise amalgamation” would not be used. We felt that ”enterprise amalgamation” should be used with the two short terms. Therefore, we decided to use both short and long terms. We call this the ”all term-patterns method”. For example, when ”enterprise amalgamation realization”[4] was input, we used ”enterprise”, ”amalgamation”, ”realization”, ”enterprise amalgamation”, ”amalgamation realization”, and ”enterprise amalgamation realization” as terms in information retrieval. We felt that this method would be effective because it makes use of all term patterns. We also felt, however, that having only the three terms ”enterprise”, ”amalgamation”, and ”realization” derived from ”... enterprise ... amalgamation ... realization ...”, while six terms are derived from ”enterprise amalgamation realization” would lack balance. We examined several methods of normalization in preliminary experiments, then decided to divide the weight of each term by $\sqrt{\frac{n(n+1)}{2}}$, where n is the number of successive words. For example, in the case of ”enterprise amalgamation realization”, $n = 3$.

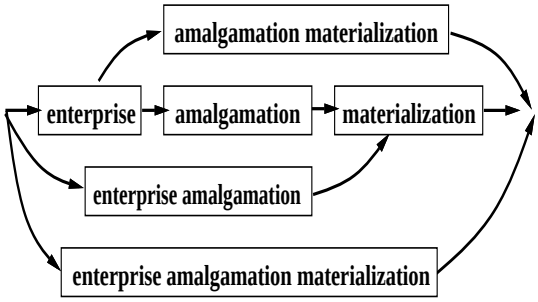

Figure 1. An example of a lattice structure

3. Using a lattice

   Although the above method effectively uses all patterns of terms, it needs to be normalized by using the ad hoc equation $\sqrt{\frac{n(n+1)}{2}}$. We thus considered a method in which all term patterns are stored in a lattice. We used the patterns in the path with the highest score on Eq. (2). The method is thus almost the same as Ozawa’s [6]. The differences are in the fundamental equation used for information retrieval, and the use or non-use of a morphological analyzer.

   In the case of ”enterprise amalgamation realization”, for example, we obtain the lattice shown in Fig. 1. The score for each of the four paths shown in this figure is calculated by using Eq. (2), and the terms along the highest-scoring path are used. This method does not require the ad hoc normalization which the method of using all term patterns requires.

4. Using de-emphasis of longer terms (”down-weighting”) [1]

   Fujita proposed this method at the IREX contest [10]. It is similar to the all-term-patterns method. All term patterns but the method of normalization is different from that used in the all-term patterns method. The weights of the shortest terms are kept constant while the weights of the longer terms are decreased. We decided to apply the weight ${k_{down}}^{x-1}$ to such terms, where x is the number of shortest terms and k_{down} was set according to the results of experiments.

[4] This example is not a term in English and is the English translation of a Japanese term “企業 (enterprise) 合併 (amalgamation) 成立 (realization)”. Its meaning is “realization of enterprise amalgamation”.

5 Dividing the query sentence into short terms

We used morphological analyzers to divide the queries into terms. We used ChaSen [3] for JJ and HAM5.0/KMA5.0 for KK. In EE, we used the OAK system for stemming terms in sentences. In the case of CC, we used the following three methods to divide the query sentences into short terms.[5]

1. Using a morphological analyzer

   In this method, the query sentence is segmented by using the CSeg&Tag 1.0 Chinese-language morphological analyzer [2].

2. Segmentation based on mutual information

   This is based on a method [11] which was proposed by Sproat et al. The mutual information of each two-character unit derived by morphological analysis is calculated; the two are divided up when the amount of mutual information is below some threshold. The details of our method are as follows. Almost all Chinese words consist of a single Chinese character or two Chinese characters.[6] We thus assumed that all terms consist of one or two Chinese characters. Thus, the application of this method starts with the division of Chinese sentences into fragments which consist of one or two Chinese characters by using mutual information. This is done by repeatedly applying the following procedure.

   - Divide up those pairs of adjacent characters that have the lowest amounts of mutual information, considering each pair that is part of a fragment which consists of more than two Chinese characters.

   Next, we use the statistics on the Chinese corpus. In this case, we assume that the ratio of one-character words to two-characters words in a Chinese text is a:b.[7] We take this statistic then re-divide those fragments that consist of pairs of characters which have little mutual information into separate one-character words, in such a way that our process of division produces a text that has been broken up into one- and two-character words in the approximate proportion a:b. This is done by repeating the following procedure until the text has been divided up in the approximate proportion a:b.

   - Divide up those fragments which consist of pairs of characters that have the lowest amounts of mutual information

   The result of this procedure is equivalent to that of the following procedure.

   - Divide up those fragments which consist of pairs of characters that have levels of mutual information equal to or less than k_{cmi}, where k_{cmi} is the amount of mutual information that will divide up the text to produce the approximate proportion a:b.

3. Using both of the above two methods

   In this method the Chinese sentences are firstly divided up by the morphological analyzer and the fragments thus derived are further divided up by using the mutual information and statistics on the Chinese corpus.

[5] We used the three more complicated methods in CC, because the tagger used for CC does not work in Unix where our information retrieval system works. In CC, we used the tagger to segment query sentences only and did not use it to segment sentences of documents in CC. We used mutual information to segment sentences of documents in CC.

[6] According to one cited paper [11], the rate of occurrence of words that consist of three Chinese characters is less than 1%.

[7] Sproat, for example, has stated that this ratio is about 7:3 [11].

6 Automatic feedback

Automatic feedback is also used in our system. An element of automatic feedback is included in our system via the IDF term of equation (2). In applying automatic feedback, we substitute the following equation for the original IDF term.

$$IDF(t) = \{E(t) + k_{af} \times (Ratio\ C(t) - Ratio\ D(t))\} \times IDF_{orig}(t) \quad (5)$$

$$E(t) = \begin{cases} 1 & \text{(when a term t is in a query)} \\ 0 & \text{(otherwise)} \end{cases} \quad (6)$$

where $Ratio\ C(t)$ is the proportion of the top k_r documents retrieved in the first round of retrieval that include a term t. $Ratio\ D(t)$ is the proportion of all documents in which the term t appears. $IDF_{orig}(t)$ is the original IDF term. This formula is based on Rocchio's formula [9]. k_{af} and k_r are constants, which are set according to the results of experiments.

Term expansion is also applied in our system. All of the terms in the top k_r documents from the first round of retrieval are tested against a binominal distribution; those terms which satisfy the test condition are introduced as terms. That is, the terms 'Terms', as defined below, are added to the set of terms.

$$Terms = \{t | P(t) \geq k_p\} \quad (7)$$

where $P(t)$ is calculated by the following equation[8] and k_p is a constant that is set based on experimental results.

[8] In this study, we used the summation of 0 to k, but the summation of 0 to $k-1$ could also be used. When the summation of 0 to k is used, an expression having a lower value for $P(t)$ is judged

$$P(t) = \sum_{r=0}^{k} C(n,r) p(u)^r (1-p(u))^{n-r} \quad (8)$$

where $C(x,y)$ is the number of combinations when we select y items from x items, n is equals to is k_r, k is the number of times the term t occurs in the top k_r documents, and $p(t)$ is calculated by

$$p(t) = \frac{\text{freq}(t)}{N} \quad (9)$$

where freq(t) is the number of the documents where the term t appears and N is the number of all documents.[9]

7 Weighting of the numbers counted in the automatic feedback process

We considered that terms which occur in higher-ranked documents and are retrieved on the first retrieval are more important than those in documents of lower rank and those retrieved later on. Thus, when counting the frequency with which a term t occurs in a document d that has a rank of $Rank(d)$, the system applies the following factor $AFW(t,d)$ to the frequency.

$$AFW(t,d) = (k_{afw}+1) - 2 \times k_{afw} \frac{Rank(d)-1}{k_r - 1} \quad (10)$$

where k_{afw} is a constant that is set according to the results of experiments. The frequency calculated by the above equation is used in calculating Equations (5) and (7).

8 Experiments

The experimental results are given in Table 1. "Query" indicates the parts of the query definition that provided inputs to our system. "T" indicates the title, "D" indicates the description, "N" indicates the narrative, and "C" indicates the concept field of the query. The column "ID" indicates the system identifiers in the NTCIR 3 contest.[10] "–" in "ID" indicates a system which was not submitted for the formal run of the NTCIR 3 contest. The values of k_r and k_{af} are as given in Table 1. Entries in the columns marked "dw", "af" and "L" indicate the application of the longer-term de-emphasis method, automatic feedback method, and the use of locational information, respectively. Use of the given method is indicated by a "y", with non-use indicated by "n". When we do not apply de-emphasis, we extract terms according to the shortest-terms method.[11] The other parameters are set as follows: $k_{location,1} = 1.2$, $k_{location,2} = 0.1$, $k_{category} = 0.1$, $k_t = 1$, $k_q = \infty$, $k_p = 0.9$, $k_{afw} = 0.5$, k_{descr} = 1, k_{proper} = 1, and k_{num} = 1. In CC, we used both the morphological analyzer and mutual information in term extraction from a query and we used only mutual information in term extraction of automatic feedback. k_{cmi} = 4.

The following findings are indicated by the experimental results.

- The automatic feedback method was always effective.
- The use of locational information as a characteristic of newspapers articles was often effective (compare "S3" and "S4", "S9" and "S10" under "R-precision; rigid", and "S20" and "S21" under "rigid").
- The longer-term de-emphasis method was sometimes effective. (compare "S3" and "S5")

Although we did not check the effectiveness of the other methods applied in our system, they would be effective. Each method and technique may only make a small contribution to the overall effectiveness. However, using all of them makes for a better system.

9 Conclusion

Multiple characteristics of information and many sophisticated techniques are applied in our information retrieval system. The techniques included Robertson's 2-Poisson model and Rocchio's formula, both of which are known to be very effective. We used such characteristics of newspapers as locational information. We used Fujita's de-emphasis ("down-weighting") method, which provides a reasonable way of including compound terms as terms used in retrieval. We also used a statistical test in expanding the queries through automatic feedback. We participated in four tasks of monolingual information retrieval (CC, JJ, KK, and EE). Our system obtained the highest values for precision on almost all of the tasks. This is because we had applied almost all of the more effective methods.

to be an expression that occurs in the top documents less often than the average occurrence in the top ducuments and it is eliminated. When the summation of 0 to $k-1$ is used, an expression having a higher value for $P(t)$ is judged to be an expression that occurs in the top documents more often than the average occurrence and the expressions other than such an expression are eliminated.

[9] This method of term expansion using a statisctical test was developed by Murata, Utiyama, and et. al. in NTCIR 2 [5].

[10] We could submit up to three systems to each task of NTCIR 3.

[11] In previous work [4], we had confirmed that using all term patterns is not a good approach, while even the simple method of using only the shortest terms leads to good results.

Table 1. Experimental results

				Parameters				R-precision		Ave. precision	
	Task	Query	ID	dw	af	L	k_r	k_{af}	rigid	relaxed	rigid	relaxed
S1	CC	TC	3	y	y	y	5	0.7	0.3084	0.3872	0.3001	0.3780
S2	CC	D	2	y	y	y	5	0.7	0.2874	0.3678	0.2672	0.3448
S3	CC	TDNC	1	y	y	y	5	0.7	0.3463	**0.4330**	**0.3379**	**0.4165**
S4	CC	TDNC	–	y	y	n	5	0.7	**0.3468**	0.4258	0.3243	0.3996
S5	CC	TDNC	–	n	y	y	5	0.7	0.3253	0.4247	0.3232	0.4062
S6	CC	TDNC	–	n	y	n	5	0.7	0.3316	0.4207	0.3148	0.3901
S7	CC	TDNC	–	y	n	y	5	0.7	0.3359	0.4014	0.3115	0.3797
S8	JJ	D	3	y	y	y	5	0.7	0.3090	0.3861	0.3312	0.3965
S9	JJ	TDNC	1	y	y	y	5	0.7	**0.3768**	0.4758	0.3990	0.4896
S10	JJ	TDNC	2	y	y	n	5	0.7	0.3679	0.4736	0.4016	0.4898
S11	JJ	TDNC	–	n	y	y	5	0.7	0.3822	0.4775	0.4027	0.4896
S12	JJ	TDNC	–	n	y	n	5	0.7	0.3687	**0.4792**	**0.4042**	**0.4902**
S13	JJ	TDNC	–	n	y	y	5	0.7	0.3659	0.4550	0.3682	0.4558
S14	KK	D	3	y	y	y	5	0.7	0.2762	0.3708	0.2691	0.3602
S15	KK	TDNC	1	y	y	y	5	0.7	0.4056	0.4989	0.3954	0.5022
S16	KK	TDNC	2	y	y	n	5	0.7	0.4037	0.4977	0.4099	0.5005
S17	KK	TDNC	–	n	y	y	5	0.7	0.4055	**0.5029**	0.3958	**0.5023**
S18	KK	TDNC	–	n	y	n	5	0.7	**0.4068**	0.4974	**0.4117**	0.5012
S19	KK	TDNC	–	y	n	y	5	0.7	0.3658	0.4448	0.3540	0.4304
S20	EE	TDNC	1	n	y	y	5	0.7	**0.4595**	0.4731	**0.4883**	0.5000
S21	EE	TDNC	–	n	y	n	5	0.7	0.4436	**0.4777**	0.4860	**0.5036**
S22	EE	TDNC	–	n	n	y	5	0.7	0.4183	0.4519	0.4549	0.4580

Acknowledgement

Our thanks go to Prof. Satoshi Sekine for developing the OAK system which we used to obtain the stems of words in English sentences. We are also grateful to Prof. Maosong Sun, who developed CSeg&Tag 1.0. We thank Prof. Dosam Hwang for the information on the Korean morphological analyzer. We greatly appreciated the kindness in all three cases.

References

[1] S. Fujita. Notes on phrasal indexing JSCB evaluation experiments at IREX-IR. *Proceedings of the IREX Workshop*, pages 45–51, 1999.

[2] S. Maosong, S. Dayang, and H. Changning. CSeg&Tag1.0: A Practical Word Segmenter and POS Tagger for Chinese Texts. In *fifth Conference on Applied Natural Language Processing, Associtation for Computational Linguistics*, pages 119–126, 1997.

[3] Y. Matsumoto, A. Kitauchi, T. Yamashita, Y. Hirano, H. Matsuda, and M. Asahara. Japanese morphological analysis system ChaSen version 2.0 manual 2nd edition. 1999.

[4] M. Murata, K. Uchimoto, H. Ozaku, Q. Ma, M. Utiyama, and H. Isahara. Japanese probabilistic information retrieval using location and category information. *the Fifth International Workshop on Information Retrieval with Asian Languages*, pages 81–88, 2000.

[5] M. Murata, M. Utiyama, Q. Ma, H. Ozaku, and H. Isahara. CRL at NTCIR2. *Proceedings of the Second NTCIR Workshop Meeting on Evaluation of Chinese & Japanese Text Retrieval and Text Summarization*, pages 5–21–5–31, 2001.

[6] T. Ozawa, M. Yamamoto, H. Yamamoto, and K. Umemuru. Word detection using the similarity measurement in information retrieval. *Proc. of the 5th Conference on Applied Natural Language Processing*, pages 305–308, 1999. (in Japanese).

[7] S. E. Robertson and S. Walker. Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In *Proceedings of the Seventeenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1994.

[8] S. E. Robertson, S. Walker, S. Jones, M. M. Hancock-Beaulieu, and M. Gatford. Okapi at trec-3. In *TREC-3*, 1994.

[9] J. J. Rocchio. *Relevance feedback in information retrieval*, pages 313–323. Prentice Hall, Inc., 1971.

[10] S. Sekine and H. Isahara. IREX project overview. *Proceedings of the IREX Workshop*, pages 7–12, 1999.

[11] R. Sproat and S. Shih. A statistical method for finding word boundaries in Chinese text. In *Computer processing of chinese & oriental languages*, pages 336–351, 1990.