Applying Multiple Characteristics and Techniques in the NICT Information Retrieval System at NTCIR-6

Masaki Murata
National Institute of Information and Communications Technology
3-5 Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-0289, Japan
murata@nict.go.jp

Jong-Hoon Oh
National Institute of Information and Communications Technology
3-5 Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-0289, Japan
rovellia@nict.go.jp

Qing Ma
Ryukoku University
Otsu, Shiga, 520-2194, Japan
qma@math.ryukoku.ac.jp

Hitoshi Isahara
National Institute of Information and Communications Technology
3-5 Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-0289, Japan
isahara@nict.go.jp

Abstract

Our information retrieval system takes advantage of numerous characteristics of information and uses numerous sophisticated techniques. It uses Robertson's 2-Poisson model and Rocchio's formula, both of which are known to be effective. Characteristics of newspapers such as locational information are used. 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 that have been statistically shown to be related to the top-ranked documents obtained in the first retrieval. We also use a numerical term, QIDF, which is an IDF term for queries. QIDF decreases the scores for stop words that occur in many queries. It can be very useful for foreign languages for which we cannot determine stop words. We also use web-based unknown word translation for bilingual information retrieval. We participated in two monolingual information retrieval tasks (Korean and Japanese) and five bilingual information retrieval tasks (Chinese-Japanese, English-Japanese, Japanese-Korean, Korean-Japanese, and English-Korean) at NTCIR-6. We obtained good results in all the tasks.

Keywords: *Monolingual IR, Locational information, De-emphasis of longer terms, Statistical test, QIDF, Web-based unknown word translation*

1 Introduction

Our information retrieval system takes advantage of numerous characteristics of information and uses numerous sophisticated techniques. Robertson's 2-Poisson model and Rocchio's formula, both of which are known to be very effective, have been used in the system. We used characteristics of newspapers such as locational information. Our system is very effective in retrieval from collections of newspaper articles, such as the document set for NTCIR-6. 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 that have been statistically shown to be related to the top-ranked documents that were obtained in the first retrieval. We also used a numerical term, QIDF, which is an IDF term for queries. It decreases the scores for stop words that occur in many queries. Furthermore, we used web-based unknown word translation for bilingual information retrieval. In NTCIR-6, we applied the system to the two tasks of monolingual information retrieval, JJ and KK, and to the five tasks of bilingual information retrieval[1] , CJ, EJ, JK, KJ, and EK. JJ and, KK stand for Japanese and Korean monolingual information retrieval. EJ stands for English-Japanese bilingual information retrieval. The source language (used in queries) is English and the target language (used in documents) is Japanese. Our system obtained good results in all the tasks in which we participated.

2 Outline of our system

Our system uses Robertson's 2-Poisson model [8], 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 the retrieval.

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

where $Score(d,q)$ is the score of a document d against a query q, t indicates a term that appears in the query, $tf(d,t)$ is the frequency of t in document d, $tf_q(q,t)$ is the frequency of t in a query q, $df(t)$ is the number of documents in which t appears, N is the total number of documents, $length(d)$ is the length of document d, Δ is the average length of the documents, and k_t and k_q are experimentally determined constants.

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

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

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

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, when 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. The total number of queries is Nq, and $qf(t)$ is the number of queries in which t occurs. Terms that occur frequently in queries are words such as *bunsho* ("document") and *mono* ("thing"). We use $log \frac{Nq}{qf(t)}$ to decrease the scores for stop words. We refer to this numerical term as QIDF because it is an IDF term for queries. It decreases the scores for words that occur in many queries (i.e., stop words). It can be very useful for foreign languages for which we cannot determine stop words. When we use QIDF, we use 1 for k_{Nq}. When we do not use QIDF, we use 0 for k_{Nq}. We introduce the extended numerical terms $K_{location}$ and K_{detail} to improve the precision of results. The location of the term within the document determines $K_{location}$. If the term is in the title or at the beginning of the body of the document, it is given a higher weight. Information such as whether the term is a proper noun and/or a stop word determines K_{detail}. In the next section, we explain these extended numerical terms in detail.

3 Extended numerical terms

We use 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 indicate the subject. Therefore, precision in information retrieval can be improved by assigning greater weight to terms from these locations. This is achieved by using $K_{location}$, which adjusts 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. We express $K_{location}$ as follows:

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

[1] Bilingual information retrieval is also called *cross-lingual information retrieval*.

[2] This equation is BM11, which corresponds to BM25 when $b = 1$ [9].

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

where $P(d,t)$ 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. The terms $k_{location,1}$ and $k_{location,2}$ are experimentally determined constants.

2. Other information (K_{detail})

   The more detailed numerical term, K_{detail}, 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 *bunsho* ("document") or *mono* ("thing"). If the term is a proper noun, it is assigned a high weight. If it is a stop word, it is assigned a low weight. For simplicity, K_{detail} is expressed in the following way; 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 terms in this equation are explained below.

   - K_{descr}
     When a term is obtained from the title of a query, i.e., a description, then K_{descr} = $k_{descr}(\geq 1)$. 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 that 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 several methods of term extraction as 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, if "enterprise amalgamation" was input, "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 for 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, for "enterprise amalgamation realization", $n = 3$.

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 [7]. The differences are in the fundamental equations used for information retrieval and the use or non-use of a morphological analyzer.

   For "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 that the method of using all term patterns requires.

[4] This example is the literal English translation of a Japanese term, "*kigyou gappei seiritsu*". It means "realization of enterprise amalgamation".

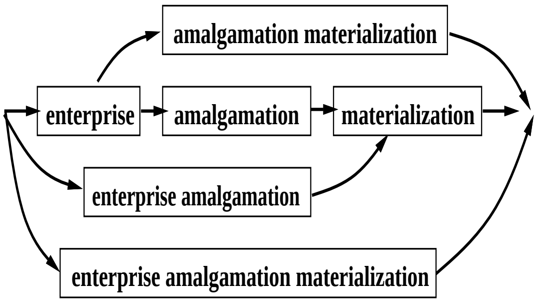

amalgamation materialization
enterprise
amalgamation
materialization
enterprise amalgamation
enterprise amalgamation materialization

Figure 1. Example of lattice structure.

4. Using de-emphasis of longer terms ("down-weighting") [2]

   Fujita proposed this method at the IREX contest [11]. It is similar to the all-term-patterns method, but the method of normalization is different. The weights of the shortest terms are kept constant, while the weights of the longer terms are decreased. We decided to apply a weight, ${k_{down}}^{x-1}$, to such terms, where x is the number of shortest terms, and k_{down} is experimentally determined.

5 Dividing a query into short terms

We used morphological analyzers to divide queries into terms. We used ChaSen [3] for JJ and HAM5.0/KMA5.0 for KK. For EE, we used the OAK system for stemming terms in sentences.

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 Eq. (2). To use automatic feedback, we substitute the following equation for the original IDF term.

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

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

where $Ratio\ C(t)$ is the proportion of the top k_r documents retrieved in the first round of retrieval that include the term t, $Ratio\ D(t)$ is the proportion of all documents in which the term t appears, and $IDF_{orig}(t)$ is the original IDF term. This formula is based on Rocchio's formula [10]. We experimentally determine the constants k_{af} and k_r.

Term expansion is also used 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 that satisfy the test condition are introduced as terms. That is, the terms, "Terms", as defined below, are added to the set of terms.

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

where $P(t)$ is calculated using the following equation[5] and k_p is an experimentally determined constant.

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

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

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

Here, freq(t) is the number of documents where the term t appears, and N is the total number of documents.[6]

7 Weighting the numbers counted in the automatic feedback process

We considered terms that occur in higher-ranked documents and are retrieved on the first retrieval to be 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.

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

where k_{afw} is an experimentally determined constant. The frequency calculated by the above equation is used in calculating $Ratio\ C(t)$ and r in Eqs. (5) and (7).

[5] In this study, we used the summation from 0 to k, but the summation from 0 to $k-1$ could also be used. When the summation from 0 to k is used, an expression having a lower value for $P(t)$ is judged to be an expression that occurs in the top documents less often than the average occurrence in the top documents, and it is eliminated. When the summation from 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.

[6] This method of term expansion using a statistical test was developed by Murata, Utiyama, et al. in NTCIR-2 [5].

Table 1. Experimental results at NTCIR-6 (NICT)

				Parameters						R-precision		Ave. precision	
	Task	Query	ID	dw	af	L	QIDF	k_r	k_{af}	Rigid	Relaxed	Rigid	Relaxed
S1	JJ	T	1	n	y	y	n	5	0.7	0.2859	0.3727	0.2821	0.3659
S2	JJ	D	2	n	y	y	n	5	0.7	0.2852	0.3618	0.2680	0.3532
S3	JJ	TDNC	3	n	y	y	y	5	0.7	0.3155	0.4085	0.2969	0.3898
S4	KK	T	1	n	y	y	n	5	0.7	0.4026	0.4711	0.4122	0.4775
S5	KK	TDNC	2	n	y	y	n	5	0.7	0.4565	0.5229	0.4710	0.5326
S6	KK	D	3	n	y	y	y	5	0.7	0.3997	0.4824	0.4139	0.4870
S7	CJ	T	1	n	y	y	n	5	0.7	0.2796	0.3270	0.2660	0.3148
S8	CJ	D	2	n	y	y	n	5	0.7	0.2384	0.2962	0.2206	0.2730
S9	CJ	DN	3	n	y	y	y	5	0.7	0.2529	0.3241	0.2394	0.3026
S10	CJ	TDNC	4	n	y	y	y	5	0.7	0.2728	0.3315	0.2492	0.3084
S11	CJ	D	5	n	y	y	y	5	0.7	0.2536	0.3186	0.2408	0.2971
S12	EJ	T	1	n	y	y	n	5	0.7	0.2195	0.2935	0.2052	0.2763
S13	EJ	D	2	n	y	y	n	5	0.7	0.2627	0.3379	0.2450	0.3260
S14	EJ	TDNC	3	n	y	y	y	5	0.7	0.2943	0.3749	0.2669	0.3522
S15	JK	T	1	n	y	y	n	5	0.7	0.2849	0.3677	0.2803	0.3592
S16	JK	D	2	n	y	y	n	5	0.7	0.2908	0.3629	0.2804	0.3485
S17	JK	DN	3	n	y	y	y	5	0.7	0.3489	0.4183	0.3343	0.4088
S18	JK	TDNC	4	n	y	y	y	5	0.7	0.3593	0.4416	0.3401	0.4213
S19	JK	D	5	n	y	y	y	5	0.7	0.2950	0.3663	0.2866	0.3559
S20	KJ	T	1	n	y	y	n	5	0.7	0.2687	0.3219	0.2452	0.3081
S21	KJ	D	2	n	y	y	n	5	0.7	0.2894	0.3604	0.2671	0.3459
S22	KJ	DN	3	n	y	y	y	5	0.7	0.2791	0.3551	0.2451	0.3266
S23	KJ	TDNC	4	n	y	y	y	5	0.7	0.2759	0.3637	0.2596	0.3397
S24	KJ	D	5	n	y	y	y	5	0.7	0.2580	0.3295	0.2412	0.3223
S25	EK	T	1	n	y	y	n	5	0.7	0.2622	0.3221	0.2496	0.3121
S26	EK	TDNC	2	n	y	y	n	5	0.7	0.3446	0.4095	0.3191	0.4073
S27	EK	D	3	n	y	y	y	5	0.7	0.2643	0.3387	0.2531	0.3257

8 How to handle bilingual information retrieval

We used high-level, commercially available software to translate a query into the target language[7], and then used the translated query for information retrieval in the target language. We did not translate the documents.

9 Experiments

The name of our team is NICT. Our experimental results at NTCIR-6 (first stage) are given in Table 1. "Query" indicates the parts of the query definition that provided input to our system. "T" indicates the title, "D" the description, "N" the narrative, and "C" the concept field of the query. The "ID" column indicates the system identifiers in the NTCIR-6 contest.[8] The values of k_r and k_{af} are as given in the table. 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 QIDF and locational information. Use of a given method is indicated by a "y", and non-use by an "n". When we do not apply de-emphasis, we extract terms according to the shortest-terms method.[9] The other parameters were 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.

The experimental results indicate the following:

- Using QIDF was effective in CJ and JK (compare "S8" and "S11", and "S16" and "S19") and not effective in KJ (compare "S21" and "S24").
- Using "TDNC" obtained good results.

Although we did not check the effectiveness of the other methods (automatic feedback method, etc.) applied in our system, they would be effective. Each

[7] The target language is the language used in the documents.

[8] We could submit up to five systems for each task of NTCIR-6.

[9] In previous work [4], we found that using all term patterns is not a good approach and that even the simple method of using only the shortest terms leads to better results.

Table 2. Experimental results at NTCIR-3 data (NICT)

			R-precision		Ave. precision	
Task	Query	ID	Rigid	Relaxed	Rigid	Relaxed
JJ	T	1	0.3221	0.3962	0.3385	0.3972
JJ	D	2	0.3055	0.3911	0.3316	0.4004
JJ	TDNC	3	0.3671	0.4690	0.3929	0.4762
KK	T	1	0.2948	0.3842	0.2858	0.3725
KK	TDNC	2	0.4073	0.5050	0.3983	0.5037
KK	D	3	0.3110	0.4065	0.3003	0.3940
CJ	T	1	0.2678	0.3322	0.2715	0.3299
CJ	D	2	0.2975	0.3738	0.3017	0.3665
CJ	DN	3	0.2932	0.3703	0.3095	0.3677
CJ	TDNC	4	0.2998	0.3712	0.3075	0.3690
CJ	D	5	0.2793	0.3729	0.2904	0.3649
EJ	T	1	0.2524	0.3158	0.2547	0.3060
EJ	D	2	0.2508	0.3145	0.2672	0.3189
EJ	TDNC	3	0.3096	0.3740	0.3237	0.3791
JK	T	1	0.2948	0.3842	0.2858	0.3725
JK	D	2	0.2864	0.3781	0.2745	0.3635
JK	DN	3	0.4049	0.4678	0.3906	0.4681
JK	TDNC	4	0.4073	0.5050	0.3983	0.5037
JK	D	5	0.3110	0.4065	0.3003	0.3940
KJ	T	1	0.2392	0.3034	0.2341	0.2850
KJ	D	2	0.2654	0.3283	0.2534	0.3114
KJ	DN	3	0.3250	0.3822	0.3201	0.3727
KJ	TDNC	4	0.3248	0.3876	0.3281	0.3776
KJ	D	5	0.2790	0.3408	0.2651	0.3292
EK	T	1	0.2948	0.3842	0.2858	0.3725
EK	TDNC	2	0.4073	0.5050	0.3983	0.5037
EK	D	3	0.3110	0.4065	0.3003	0.3940

Table 3. Experimental results at NTCIR-4 data (NICT)

			R-precision		Ave. precision	
Task	Query	ID	Rigid	Relaxed	Rigid	Relaxed
JJ	T	1	0.3730	0.4764	0.3524	0.4638
JJ	D	2	0.3799	0.4759	0.3604	0.4624
JJ	TDNC	3	0.4025	0.5106	0.3803	0.4955
CJ	T	1	0.3009	0.3918	0.2815	0.3710
CJ	D	2	0.2680	0.3463	0.2405	0.3232
CJ	DN	3	0.3252	0.4191	0.2925	0.3896
CJ	TDNC	4	0.3246	0.4154	0.2961	0.3907
CJ	D	5	0.2692	0.3458	0.2347	0.3143
EJ	T	1	0.2933	0.3885	0.2681	0.3627
EJ	D	2	0.3318	0.4068	0.2993	0.3848
EJ	TDNC	3	0.3682	0.4743	0.3377	0.4481
KJ	T	1	0.2831	0.3498	0.2552	0.3301
KJ	D	2	0.2721	0.3434	0.2443	0.3183
KJ	DN	3	0.3313	0.4155	0.2996	0.3921
KJ	TDNC	4	0.3378	0.4172	0.3033	0.3885
KJ	D	5	0.2907	0.3636	0.2577	0.3357

Table 4. Experimental results at NTCIR-5 data (NICT)

			R-precision		Ave. precision	
Task	Query	ID	Rigid	Relaxed	Rigid	Relaxed
JJ	T	1	0.3622	0.4612	0.3613	0.4615
JJ	D	2	0.3180	0.4240	0.3162	0.4154
JJ	TDNC	3	0.3828	0.4786	0.3896	0.4894
KK	T	1	0.4764	0.5320	0.4912	0.5441
KK	TDNC	2	0.4874	0.5491	0.5159	0.5799
KK	D	3	0.4718	0.5372	0.4936	0.5571
CJ	T	1	0.2494	0.3143	0.2319	0.3037
CJ	D	2	0.2256	0.3183	0.2245	0.3137
CJ	DN	3	0.2773	0.3812	0.2700	0.3686
CJ	TDNC	4	0.2673	0.3636	0.2641	0.3573
CJ	D	5	0.2248	0.3127	0.2252	0.3128
EJ	T	1	0.2537	0.3271	0.2458	0.3210
EJ	D	2	0.2752	0.3625	0.2663	0.3590
EJ	TDNC	3	0.3056	0.4119	0.3006	0.4000
JK	T	1	0.4764	0.5320	0.4912	0.5441
JK	D	2	0.4771	0.5268	0.4897	0.5449
JK	DN	3	0.4845	0.5560	0.5089	0.5771
JK	TDNC	4	0.4874	0.5491	0.5159	0.5799
JK	D	5	0.4718	0.5372	0.4936	0.5571
KJ	T	1	0.2678	0.3482	0.2642	0.3584
KJ	D	2	0.2568	0.3384	0.2472	0.3360
KJ	DN	3	0.3161	0.4158	0.3051	0.4073
KJ	TDNC	4	0.2986	0.4026	0.2930	0.3991
KJ	D	5	0.2558	0.3411	0.2486	0.3432
EK	T	1	0.4764	0.5320	0.4912	0.5441
EK	TDNC	2	0.4874	0.5491	0.5159	0.5799
EK	D	3	0.4718	0.5372	0.4936	0.5571

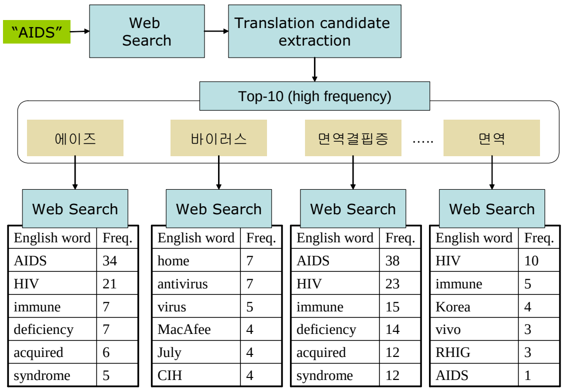
method and technique may only make a small contribution to the overall effectiveness. However, using all of them makes for a better system.

Our experimental results in NTCIR-3, NTCIR-4, and NTCIR-5 data (second stage of NTCIR-6) are given in Tables 2, 3, and 4.

10 CLIR using Web-based Unknown Word Translation

We submitted two runs based on CLIR using Web-based unknown word translation to the EJ and EK CLIR tracks in stage 1, respectively. In the runs, we employed a dictionary-based query translation technique using a bilingual dictionary. Note that the EJ and EK runs in Table 1 used a machine translation software for query translation. However, query translation based on a bilingual dictionary in CLIR frequently suffers from out-of-vocabulary or unknown word problem caused by proper nouns or technical terms [1]. To address the problem, we extracted translations of unknown words (unregistered in a bilingual dictionary) from the Web. Figure 2 shows examples of our unknown word translations for the English word *AIDS*.

For a given unknown word, *AIDS*, we first retrieve Web pages using the unknown word as a query term for Web search engine. The Web search results usu-



English word	Freq.
AIDS	34
HIV	21
immune	7
deficiency	7
acquired	6
syndrome	5

English word	Freq.
home	7
antivirus	7
virus	5
MacAfee	4
July	4
CIH	4

English word	Freq.
AIDS	38
HIV	23
immune	15
deficiency	14
acquired	12
syndrome	12

English word	Freq.
HIV	10
immune	5
Korea	4
vivo	3
RHIG	3
AIDS	1

Figure 2. Examples of Web-based unknown word translations

ally consist of a series of snippets composed of title and summary of each retrieved documents. Through morphological analysis[10] , we extract a list of translation candidates that appear with high frequency from the Web search results. Then the translation candidates are validated using a joint validation model [6]. Let s be the unknown word, T be a set of translation candidates extracted from Web-search results retrieved by s, and t_i be the i^{th} translation candidate in T. Also, let S_i be a set of English words extracted from Web-search results retrieved by query t_i, s_{ij} be the j^{th} English word in S_i, and $Freq_s(t_i)$ be frequency of t_i in Web search results retrieved by s. The assumption underlying joint validation is that s will be the most relevant counterpart of t_i and vice versa if they are the correct translation pair. Then, we can select the most relevant translation candidate using the equation

$$S_{Joint}(s, t_i) = S_{forward}(s, t_i) \times S_{backward}(t_i, s) \quad (11)$$

$$S_{forward}(s, t_i) = \frac{Freq_s(t_i)}{\sum_{t_j \in T} Freq_s(t_j)}$$

$$S_{backward}(t_i, s) = \frac{Freq_{t_i}(s)}{\sum_{s_{il} \in S_i} Freq_{t_i}(s_{il})}$$

10.1 Results of CLIR using Web-based unknown word translation

Table 5 shows monolingual-retrieval and our CLIR results (the average precision of each run). Here, we used monolingual-retrieval, JJ and KK, as baseline systems for EJ and EK, respectively. JJ and KK in Table 5 were submitted for the formal run of the NTCIR-6 contest. Compared to the baseline systems, our CLIR achieved 78–97% in relaxed and rigid evaluation.

Table 5. Ave. precision at NTCIR-6 (NICT)

Task	Query	ID	Relaxed	Rigid
JJ	T	4	0.3499	0.2762
JJ	D	5	0.3389	0.2584
EJ	T	4	0.2819	0.2195
EJ	D	5	0.3206	0.2508
KK	T	4	0.4439	0.3566
KK	D	5	0.4497	0.3704
EK	T	4	0.3755	0.2813
EK	D	5	0.3736	0.2915

11 Conclusion

Multiple characteristics of information and many sophisticated techniques are used in our information retrieval system. The techniques include Robertson's 2-Poisson model and Rocchio's formula, both of which are known to be very effective. We used characteristics of newspapers such as locational information. We used Fujita's de-emphasis (down-weighting) method, which provides a reasonable way of using compound terms in retrieval. We also used a statistical test in expanding queries through automatic feedback. We used a numerical term, QIDF, which is an IDF term for queries. It decreases the scores for stop words that occur in many queries. It can be very useful for foreign languages for which we cannot determine stop words. Furthermore, we used web-based unknown word translation for bilingual information retrieval. We participated in two monolingual information retrieval tasks (Korean and Japanese) and five of bilingual information retrieval tasks at NTCIR-6. We obtained good results in all the tasks in which we participated.

Acknowledgements

We thank Prof. Dosam Hwang for information on the Korean morphological analyzer. We are grateful to all of the organizers of NTCIR-6, who gave us a chance to participate in the NTCIR-6 contest to improve and examine our information retrieval techniques. We greatly appreciate the kindness of all those who helped us.

References

[1] A. Fujii and I. Tetsuya. Japanese/English cross-language information retrieval: Exploration of query

[10] JUMAN (http://nlp.kuee.kyoto-u.ac.jp/nl-resource/juman-e.html) for Japanese and KLT (http://nlp.kookmin.ac.kr/HAM/kor/index.html) for Korean

translation and transliteration. *Computers and the Humanities*, 35(4):389–420, 2001.

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

[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] J.-H. Oh and H. Isahara. Mining the web for transliteration lexicons: Joint-validation approach. In *2006 IEEE/WIC/ACM International Conference on Web Intelligence (WI'06)*, pages 254–261, 2006.

[7] 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).

[8] 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.

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

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

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