Cross-Language IR at University of Tsukuba: Automatic Transliteration for Japanese, English, and Korean

Atsushi Fujii and Tetsuya Ishikawa
Graduate School of Library, Information and Media Studies
University of Tsukuba
1-2 Kasuga, Tsukuba, 305-8550, Japan
fujii@slis.tsukuba.ac.jp

Abstract

This paper describes our cross-language information retrieval system for the NTCIR-4 CLIR task. Our system, which follows the query translation approach, uses a compound word translation and transliteration. Transliteration is effective if a query includes foreign words, such as technical terms and proper nouns, spelled out by phonetic alphabets. We apply our method, which was originally proposed for Japanese Katakana words, to Korean Hangul words and realize JEK transliteration in a single framework. We produced a transliteration dictionary for Japanese and English letters via the Roman representation. To produce a new dictionary, we use the Unicode system to romanize Korean words. We also show the effectiveness of our method by means of experiments.

Keywords: *Cross-language information retrieval, Query Translation, Transliteration, Foreign words*

1 Introduction

In the NTCIR-4 cross-language information retrieval (CLIR) task [11], search topics and newspaper articles in Japanese, English, Korean, and Traditional Chinese were used to evaluate the performance of participating IR systems. The NTCIR-4 CLIR task consists of the following four subtasks:

- Multilingual CLIR (MLIR), in which a topic in one of the four languages is used to search a collection in more than one language for the documents relevant to user information needs,
- Bilingual CLIR (BLIR), which resembles the MLIR task, but a document collection is in a single language,
- Single Language IR (SLIR), which is a monolingual IR task for one of the four languages,
- Pivoted Bilingual CLIR (PLIR), in which a query is indirectly translated into a document language via a third language.

For the formal run, we submitted the retrieval results (i.e., the run files) for the first three subtasks. While in MLIR and BLIR we used only Japanese as the source language, in SLIR we used all the four languages as the source language independently.

Using the official evaluation results, we compared the performance of the case in which a Japanese user searches a document collection in a foreign language by CLIR system and the case in which a user speaking the document language as their native language retrieves the same collection by our SLIR system.

After the formal run, we further evaluated the performance of our CLIR system using the official relevance judgement files. A strength of our system is an automatic transliteration method, which associates out-of-dictionary query terms to phonetic equivalents in a target language. This method is effective to translate words imported from a foreign language and spelled out by phonetic alphabets, such as *Katakana* in Japanese. These words are usually technical terms and proper nouns.

In previous studies [4, 5, 7], we showed that our transliteration method improved the performance of CLIR for Japanese and English. This method has been used in a commercial J/E cross-language patent retrieval service[1]. In the NTCIR-4, we extended our method to Korean and evaluated its effectiveness for Japanese, English, and Korean.

Because the BLIR subtask is usually termed "cross-language/lingual information retrieval (CLIR)" in past literature, we shall use the terms "CLIR" and "BLIR" interchangeably. We focus mainly on CLIR, because CLIR and MLIR share the essential issues.

Sections 2 and 3 describe the outline of our system and the transliteration method, respectively. In Section 4, we elaborate on the experiments while/after the formal run.

[1] http://www.patolis.co.jp/products/e-index.html

2 System Description

2.1 Overview

Because by definition the queries and documents for CLIR are in different languages, they need to be standardized into a common representation so that monolingual retrieval techniques can be used. From this perspective, existing CLIR methods can be classified into the following fundamental approaches.

In the first approach queries are translated into a document language [1, 7]. In the second approach documents are translated into a query language [16]. In the third approach both queries and documents are projected into a language-independent space by means of thesauri [9, 18] and latent semantic indexing [2, 14]. Additionally, in the hybrid approach multiple fundamental approaches are used together [6, 15].

While we proposed a two-stage method, which combines query and document translation methods [6], in NTCIR-4 we evaluated only the performance of the query translation method.

Figure 1 depicts the overall design of our system. In CLIR, a query is first translated into a document language and the translation is used to search a monolingual collection for relevant documents, which are sorted according to the score.

In MLIR, a query in one language is first translated into the other languages independently. Second, each of the source and translated queries is used to search the corresponding monolingual collection. Finally, the retrieved document lists for the four languages are combined and the documents are sorted according to the score obtained in the retrieval process.

2.2 Query Translation

The query translation module is based on our previous method for compound words [5, 7].

First, we extract compound words from a target topic field. For `<TITLE>`, we use a delimiter (e.g., comma) to extract compound words.

For `<DESC>` and `<NARR>`, we perform morphological analysis and regard a sequence of content words (e.g., nouns) as a compound word. We use the following tools for morphological analysis purposes: ChaSen[2] (J), WordNet[3] (E), Kemorphor[4] (K), SuperMorph[5] (C).

For English, query terms in the stopword list of WordNet are discarded and the remaining words (i.e., nouns, verbs, adjectives, adverbs, and out-of-dictionary words) are used as query terms.

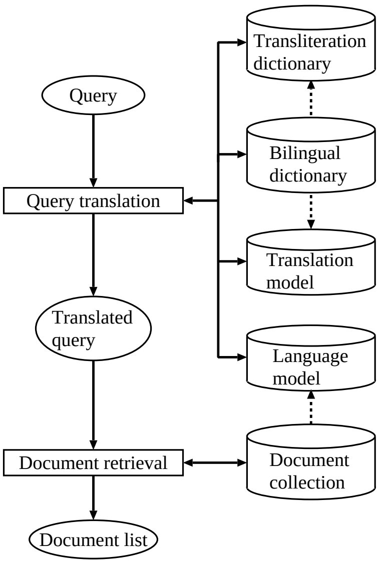


Figure 1. Design of our CLIR system (solid and dashed arrows denote on-line and off-line processes, respectively).

For Japanese, Korean, and Chinese, in which sentences lack lexical segmentation, we perform morphological analysis to identify words and their parts-of-speech, and extract content words. Although Korean sentences are segmented on a phrase-by-phrase basis, post-position suffixes (*Josa*) need to be discarded.

Second, for each of the compound words extracted from the topic, we derive possible word and phrase translations using a bilingual dictionary. We consider all possible segmentations of an input word by consulting the bilingual dictionary. We select such segmentations that consist of the minimal number of words.

During the segmentation process, transliteration is performed for out-of-dictionary words to identify phonetic equivalents in a document language. Transliteration is not performed for Chinese words and words including numerals (e.g., “Y2K”). For Japanese, transliteration is applied only to Katakana words.

Finally, we use a probabilistic method to resolve translation ambiguity. The formula for the source compound word S and a transliteration candidate T are represented as below.

$$\begin{aligned} S &= s_1, s_2, \ldots, s_n \\ T &= t_1, t_2, \ldots, t_n \end{aligned}$$

Here, s_i denotes an i-th word, and t_i denotes a translation candidate of s_i. From the viewpoint of probability theory, our task is to select T’s with greater probabilities, $P(T|S)$, which can be transformed as in Equation (1) through the Bayesian theorem.

$$P(T|S) = \frac{P(S|T) \cdot P(T)}{P(S)} \quad (1)$$

[2] http://chasen.aist-nara.ac.jp/

[3] http://www.cogsci.princeton.edu/ wn/

[4] http://www.crosslanguage.co.jp/english/

[5] http://www.omronsoft.com/

$P(S)$ can be omitted because this factor is a constant with respect to the given query and does not affect the relative probability for different translation candidates. If a user utilizes more than one translation as query terms, T's with greater probabilities are selected.

$P(S|T)$ and $P(T)$ are termed language and translation models, respectively. We approximate these factors using statistics associated with base words, as in Equation (2).

$$\begin{aligned} P(S|T) &\approx \prod_{i=1}^{n} P(s_i|t_i) \\ P(T) &\approx \prod_{i=1}^{n} P(t_i|t_{i-2}, t_{i-1}) \end{aligned} \tag{2}$$

We produced a translation model based on the word frequency in the bilingual dictionary. We produced a word tri-gram language model including the 100K high frequency words in the target document collection, for which we used Palmkit[6]. In practice, two dummy symbols were inserted at the beginning and end of T, respectively, to estimate $P(t_i|t_{i-2}, t_{i-1})$ $(i = 1, 2, n-1, n)$.

Table 1 shows the dictionaries used for our experiments. While the NTCIR-4 Chinese topics and documents were in Traditional Chinese represented by Big5, our dictionaries were in Simplified Chinese represented by GB2312. We converted the character code of our dictionaries into Big5 by the Linux `iconv` command.

EDICT[7] and an E-K dictionary were used only in the experiments after the formal run. To investigate the effect of the dictionary size, we compared the results obtained with the Cross Language dictionary and the results that obtained with EDICT, which was used for both J-E and E-J CLIR. We used the E-K dictionary for both E-K and K-E CLIR, because K-E dictionaries were not available.

Table 1. Dictionaries used for experiments (T: technical dictionary, G: general dictionary).

Languages	Name or Developer	#Entries	Type
J-E	Cross Language Inc.	1M	T
E-J	Cross Language Inc.	1M	T
J-E/E-J	EDICT	108K	G
J-K	UNISOFT Corp.	213K	G
K-J	UNISOFT Corp.	134K	G
J-C	Cross Language Inc.	1M	T
C-J	Cross Language Inc.	865K	T
E-K/K-E	Cross Language Inc.	548K	T

2.3 Document Retrieval

The document retrieval module is based on an existing probabilistic method [17], which computes the relevance score between a (translated) query and each document in a collection. The relevance score for document d is computed based on Equation (3).

$$\sum_t f_{t,q} \cdot \frac{(K+1) \cdot f_{t,d}}{K \cdot \left\{(1-b) + \frac{dl_d}{b \cdot avgdl}\right\} + f_{t,d}} \cdot \log \frac{N - n_t + 0.5}{n_t + 0.5} \tag{3}$$

Here, $f_{t,q}$ and $f_{t,d}$ denote the frequency that term t appears in query q and document d, respectively; N and n_t denote the total number of documents in the collection and the number of documents containing term t, respectively; dl_d denotes the length of document d, and $avgdl$ denotes the average length of documents in the collection. We empirically set $K = 2.0$ and $b = 0.8$, respectively.

For indexing purposes we used content words, which were extracted by the same method as in Section 2.2, as index terms. However, for the Japanese documents, we used both character bi-grams and words as index terms.

We also used a pseudo-relevance feedback method. In practice, we first retrieved the top ten documents and sorted the index terms in those documents according to the term weight. We added the top ten terms to the initial query and retrieved the final result.

3 Transliteration

3.1 Overview

The basis of our transliteration method [5, 7] is similar to that for compound word translation in Section 2.2. The formula for the source word S and a transliteration candidate T are represented as below.

$$\begin{aligned} S &= s_1, s_2, \ldots, s_n \\ T &= t_1, t_2, \ldots, t_n \end{aligned}$$

Unlike the case of compound word translation, s_i and t_i denote i-th “symbols”, which consist of one or more letters, respectively. To derive possible s_i's and t_i's, we consider all possible segmentations of S, by consulting a dictionary for symbols, namely the “transliteration dictionary”. We select such segmentations that consist of the minimal number of symbols.

We resolve the transliteration ambiguity based on the a probabilistic model similar to that for the compound word translation. We compute $P(T|S)$ for each T and select T's with greater probabilities. T must be a correct word that is indexed in a target document collection. However, because $P(T)$ is computed by combining $P(t_i)$'s for all substrings in T, incorrect words are potentially assigned to a positive value of $P(T)$.

[6] http://sourceforge.net/projects/palmkit/

[7] http://www.csse.monash.edu.au/˜jwb/edict.html

In view of this problem, we estimate $P(T)$ as the probability that T occurs in the document collection, and consequently $P(T)$ for unindexed words becomes zero. In practice, to enhance the computational efficiency, we perform a pruning during the segmentation process. While we progressively produce a transliteration candidate from the beginning of S, we also perform a forward partial-matching to discard the candidates in progress that do not match with any of the index terms.

We approximate $P(S|T)$ as in Equation (2), and estimate $P(s_i|t_i)$ based on the correspondence frequency for each combination of s_i and t_i in the transliteration dictionary.

3.2 Dictionary Production

The method to produce the transliteration dictionary is crucial, because such dictionaries have rarely been published. To illustrate our dictionary production method, we consider Figure 2, in which we insert hyphens between each Katakana character for enhanced readability. In this figure, the first letter in each Katakana character tends to be contained in its corresponding English word. However, there are a few exceptions. A typical case is that because Japanese has no distinction between "L" and "R" sounds, the two English sounds collapse into the same Japanese sound. In addition, a single English letter may correspond to multiple Katakana characters, such as "x" to "*ki-su*" in "<text, *te-ki-su-to*>". In summary, English and romanized Katakana words are not exactly identical, but similar to each other.

English	Japanese
system	*si-su-te-mu*
mining	*ma-i-ni-n-gu*
text	*te-ki-su-to*
collocation	*ko-ro-ke-i-syo-n*

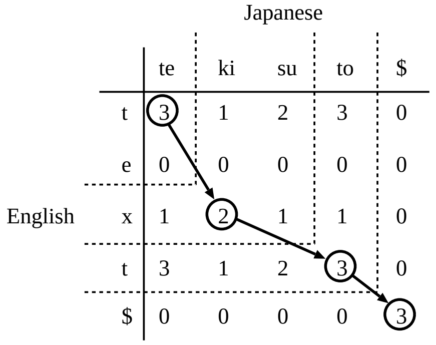
Figure 2. Example correspondences between English and romanized Japanese Katakana words.

This phenomena can also be observed in Korean. For example, the English word "system" is romanized as "*si-seu-te-m*". This motivated us to extend our transliteration method to Korean.

We first manually defined the similarity between the English letter e and the first romanized letter for each Katakana character j, as shown in Table 2. In this table, "phonetically similar" letters refer to a certain pair of letters, such as "L" and "R," for which we identified approximately twenty pairs of letters. We then consider the similarity for any possible combination of letters in English and romanized Katakana words, which can be represented as a matrix, as shown in Figure 3. This figure shows the similarity between letters in "<text, *te-ki-su-to*>". We put a dummy letter "$," which has a positive similarity only to itself, at the end of both English and Katakana words.

Table 2. Similarity between English letter e and Japanese letter j.

Condition	Similarity
e and j are identical	3
e and j are phonetically similar	2
both e and j are vowels or consonants	1
otherwise	0



		te	ki	su	to	$
	t	3	1	2	3	0
	e	0	0	0	0	0
English	x	1	2	1	1	0
	t	3	1	2	3	0
	$	0	0	0	0	3

Figure 3. Example matrix for English-Japanese symbol matching (arrows denote the best path).

Here, matching plausible symbols can be recast as identifying the path which maximizes the total similarity from the first to last letters. The best path can efficiently be found by, for example, Dijkstra's algorithm [3]. From Figure 3, we can derive the following correspondences: "<te, *te*>", "<x, *ki-su*>", and "<t, *to*>". We used bilingual dictionaries in Table 1 to produce the transliteration dictionaries for Japanese, English, and Korean.

Japanese words consist of different types of characters, such as "*Kanji*", "*Katakana*", "*Hiragana*", alphabets, and numerals. Here, *Kanji* (Chinese character) is the Japanese idiogram, and *Katakana* and *Hiragana* are the phonograms.

However, foreign words are always spelled out by Katakana, which is seldom used to describe the conventional Japanese words excepting proper nouns. Thus, in cases where Japanese is a source or target language, we extracted only Katakana words and their translations from the dictionaries. This process can be performed systematically on the basis of a Japanese character code, such as EUC-JP and SJIS, in which all Katakana characters are coded in a specific region.

At the same time, there are noisy correspondences in which a Katakana word is not the precise transliteration of a foreign word. For example, "*waapuro*", which is a short form of "*waadopurosessaa*", is listed as a translation of the English word "word processor". To exclude these noisy correspondences, we used only the translations whose total similarity from the first to last letters is above a predefined threshold. The method to exclude noisy correspondences is more effective in Korean, because both conventional and foreign words are written with *Hangul* characters.

In summary, for any language pair of Japanese, English, and Korean, our method produces a transliteration dictionary as follows:

1. romanizes Japanese and/or Korean words in a bilingual dictionary,
2. computes the phone-based similarity between a source word and its translation, and aligns them on a symbol-by-symbol basis,
3. selects the translation pairs whose similarity is above a threshold,
4. derives a transliteration dictionary from the selected translation pairs.

While the basis is the same as in our previous method, a new challenge in the NTCIR-4 was to romanize Korean words, which is described in Section 3.3.

3.3 Romanizing Korean Words

In Japanese, one-to-one mapping between each phone and its Roman representation can be performed with a negligible cost, because the numbers of Katakana characters and combined phones are small.

However, the number of Korean Hangul characters is much greater than that of Japanese Katakana characters. Each Hangul character is a combination of more than one consonant. The pronunciation of each character is determined by its component consonants.

In Korean, there are three types of consonants, i.e., the first consonant, vowel, and last consonant. The numbers of these consonants are 19, 21, and 27, respectively. The last consonant is optional. Thus, the total number of combined characters is 11,172. However, to transliterate imported words, the official guideline suggests that only seven consonants be used as the last consonant.

In EUC-KR, which is a common coding system for Korean text, 2350 common characters are coded independent of the pronunciation. Therefore, if we target text in EUC-KR, each of the 2350 characters has to be corresponded to its Roman representation.

In view of this problem, we used the Unicode system, in which Hangul characters are sorted according to the pronunciation. Figure 4 depicts a fragment of the Unicode table for Korean, in which each line corresponds to a combination of the first consonant and vowel and each column corresponds to the last consonant. The number of columns is 28, i.e., the number of the last consonants and the case in which the last consonant is not used. From this figure, the following rules can be found:

- the first consonant changes every 21 lines, which corresponds to the number of vowels,
- the vowel changes every line (i.e., 28 characters) and repeats every 21 lines,
- the last consonant changes every column.

Based on these rules, a specific character and its pronunciation can be identified systematically by means of the three consonant types. Thus, we manually corresponded only the 68 consonants to Roman alphabets. Because the entries in our E-K dictionary were represented by EUC-KR, we used the `native2ascii` command in the Java 2 SDK, to convert these entries into Unicode.

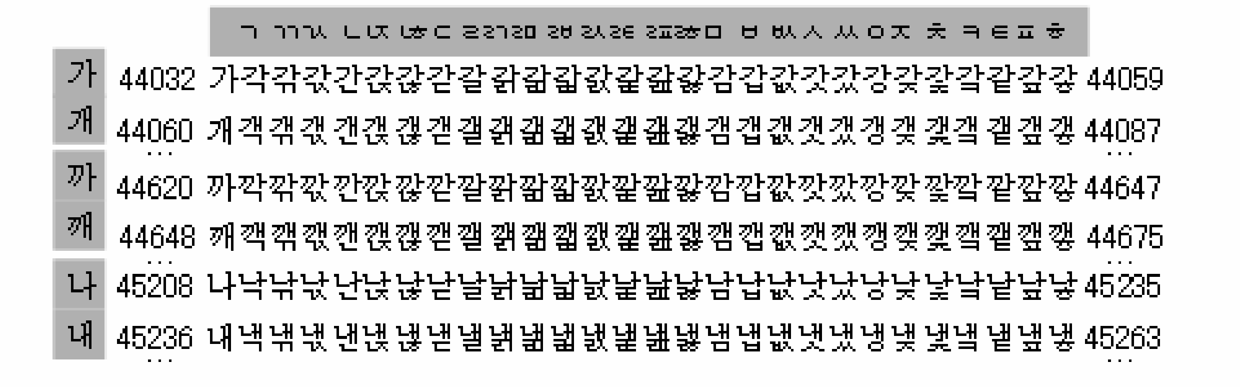
3.4 Related Work

Lee and Choi [13] and Jeong et al. [10] independently explored English-Korean transliteration. These method correspond English letters to Hangul letters directly, while we first romanized Hangul characters and then corresponded them to English letters. In other words, we used pronunciation information to improve the correspondence accuracy.

Because in Korean both foreign and conventional words are written with Hangul, it is crucial to select foreign words prior to the dictionary production. Jeong et al. [10] proposed a statistical method to detect foreign words in Korean. However, their method requires a training corpus in which conventional and foreign words are annotated. Our method, which can select plausible foreign words by means of the phonetic similarity with English or Japanese words, does not require (annotated) corpora. We applied this method to extracting foreign words from Korean text [8].

Knight and Graehl [12] proposed a Japanese-English transliteration method based on the correspondence probability between English and Japanese Katakana sounds. However, their method requires a phoneme inventory and cannot generate English words not listed in the inventory. Our method is not constrained by phoneme inventories.

In addition, we realized transliteration for more than two languages in a single framework and evaluated its performance by means of CLIR experiments. To the best of our knowledge, no method has been used and evaluated in CLIR for more than two languages.



			ㄱ	ㄲ	ㄳ	ㄴ	ㄵ	ㄶ	ㄷ	ㄹ	ㄺ	ㄻ	ㄼ	ㄽ	ㄾ	ㄿ	ㅀ	ㅁ	ㅂ	ㅄ	ㅅ	ㅆ	ㅇ	ㅈ	ㅊ	ㅋ	ㅌ	ㅍ	ㅎ	
가	44032	가	각	갂	갃	간	갅	갆	갇	갈	갉	갊	갋	갌	갍	갎	갏	감	갑	값	갓	갔	강	갖	갗	갘	같	갚	갛	44059
개	44060	개	객	갞	갟	갠	갡	갢	갣	갤	갥	갦	갧	갨	갩	갪	갫	갬	갭	갮	갯	갰	갱	갲	갳	갴	갵	갶	갷	44087
	...																													...
까	44620	까	깍	깎	깏	깐	깑	깒	깓	깔	깕	깖	깗	깘	깙	깚	깛	깜	깝	깞	깟	깠	깡	깢	깣	깤	깥	깦	깧	44647
깨	44648	깨	깩	깪	깫	깬	깭	깮	깯	깰	깱	깲	깳	깴	깵	깶	깷	깸	깹	깺	깻	깼	깽	깾	깿	꺀	꺁	꺂	꺃	44675
	...																													...
나	45208	나	낙	낚	낛	난	낝	낞	낟	날	낡	낢	낣	낤	낥	낦	낧	남	납	낪	낫	났	낭	낮	낯	낰	낱	낲	낳	45235
내	45236	내	낵	낶	낷	낸	낹	낺	낻	낼	낽	낾	낿	냀	냁	냂	냃	냄	냅	냆	냇	냈	냉	냊	냋	냌	냍	냎	냏	45263
	...																													...

Figure 4. Fragment of the Unicode table for Korean Hangul characters.

4 Evaluation

4.1 Formal run results

Table 3 shows the mean average precision (MAP) values for our results submitted to the formal run.

Relevance assessment was performed based on four ranks of relevance, that is, highly relevant (S), relevant (A), partially relevant (B) and irrelevant (C). In Table 3 the columns for “Rigid” denote the cases in which the documents judged S and A were regarded as the correct answers and the columns for “Relaxed” denote the cases in which the documents judged B were also regarded as the correct answers.

In the BLIR and MLIR subtasks, for each (compound) word in a source language, the top three translation candidates were used for retrieval purposes. We used transliteration only to convert Japanese query terms into English in J-E and J-CJKE.

Our primary research interest in the formal run was to compare the performance of BLIR and SLIR. According to Table 3, the MAP values for J-E and J-K were roughly 70% of those for E-E and K-K, respectively. This ratio has also been observed in past CLIR literature and is fairly reasonable.

However, the performance of J-C was significantly low. The MAP value for J-C was roughly 25% of that for C-C. The MAP value of J-CJKE was also decreased accordingly. A possible reason is that our dictionaries in Simplified Chinese were automatically converted into Traditional Chinese and thus the translation accuracy was decreased. This issue needs to be further explored.

4.2 Evaluating Transliteration

Table 4 shows the MAP values for our results obtained after the formal run. However, J-E combined with transliteration used the same method for J-E in Table 3. As in the formal run, for each (compound) word in a source language, the top three translation candidates were used for retrieval purposes.

While the MAP values of J-E and E-J were obtained with the dictionaries developed by the Cross Language Inc., the MAP values of J-E* and E-J* were obtained with EDICT.

In Table 4, the effects of transliteration can be investigated for different methods. For each MAP pair to be compared, we boldfaced the greater values. Suggestions which can be derived from Table 4 are as follows.

First, in most cases the MAP values were improved by means of our transliteration method. In the cases associated with Japanese and English, the improvement was more salient when EDICT was used. As in Table 1, the numbers of entries in the Cross Language J/E dictionaries is ten times as large as that of EDICT. In other words, the number of out-of-dictionary terms increased when EDICT was used.

Second, in K-J and K-E, the MAP values of D and DN decreased when combined with transliteration, while the MAP values of T increased by means of transliteration.

For D and DN, we performed morphological analysis for Korean sentences before the query translation process. Although we used different dictionaries for morphological analysis and query translation, foreign words were often listed in neither of the dictionaries. Consequently, specific foreign words were mistakenly interpreted as combinations of conventional Korean words.

Third, in E-K the MAP values of D decreased marginally by transliteration. This was due to the topic 041, in which the English word “received” was not in our dictionary and the transliteration was performed, although this word should not be transliterated. A possible solution is to use a probability score as a confidence measure and discard the transliteration candidates whose score is below a threshold.

Table 3. MAP values for different methods in the formal run (T: TITLE, D: DESC, N: NARR).

		Rigid			Relaxed		
Subtask	Languages	T	D	D+N	T	D	D+N
SLIR	J-J	0.2886	0.2957	0.3373	0.4033	0.4024	0.4535
SLIR	E-E	0.3090	0.2953	0.3358	0.3954	0.3732	0.4117
SLIR	C-C	0.2206	0.1920	0.2473	0.2649	0.2539	0.2981
SLIR	K-K	0.3794	0.3675	0.3675	0.4081	0.3934	0.3934
BLIR	J-E	0.2182	0.2225	0.2261	0.3008	0.2943	0.2929
BLIR	J-C	0.0483	0.0548	0.0673	0.0633	0.0682	0.0819
BLIR	J-K	0.2457	0.2363	0.2696	0.2681	0.2613	0.2998
MLIR	J-CJKE	0.1316	0.1296	0.1335	0.1888	0.1858	0.1877

Finally, the MAP values of E-J* were relatively low. This was partially due to the fact that Japanese person names romanized in the English topics, such as “Keizo Obuchi”, “Akira Kurosawa”, and “Masako”, were not correctly translated. Because these words are not Katakana words, our transliteration method can not be applied. In addition, a single Roman representation usually corresponds to multiple Japanese words, specifically, *Kanji* characters, we need a method to resolve back-romanization ambiguity.

5 Conclusion

We described our system for the NTCIR-4 CLIR task, in which Japanese, English, Korean, and Chinese were used as target languages. Our CLIR method, which followed the query translation approach, used a compound word translation and transliteration.

Through the experiments in the formal run, we identified that the performance of J-C CLIR needs to be further improved, while for the other language pairs the performance was comparable with those reported in past literature.

After the formal run, we applied our transliteration method, which was originally proposed for Japanese and English, to Korean and realized the transliteration for the three languages in a single framework. Because our method produces a transliteration dictionary via the Roman representation, a challenge was to romanize Korean words. To resolve this problem, we used the Unicode system, in which Korean Hangul characters were coded according to the pronunciation information. We also showed the effectiveness of our transliteration method by means of experiments.

Future work will include applying the automatic transliteration method to other languages, such as Chinese, in which the processing of out-of-dictionary foreign words is problematic.

References

[1] L. Ballesteros and W. B. Croft. Resolving ambiguity for cross-language retrieval. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 64–71, 1998.

[2] J. G. Carbonell, Y. Yang, R. E. Frederking, R. D. Brown, Y. Geng, and D. Lee. Translingual information retrieval: A comparative evaluation. In *Proceedings of the 15th International Joint Conference on Artificial Intelligence*, pages 708–714, 1997.

[3] E. W. Dijkstra. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1:269–271, 1959.

[4] A. Fujii and T. Ishikawa. Cross-language information retrieval at ULIS. In *Proceedings of the 1st NTCIR Workshop on Research in Japanese Text Retrieval and Term Recognition*, pages 163–169, 1999.

[5] A. Fujii and T. Ishikawa. Cross-language information retrieval for technical documents. In *Proceedings of the Joint ACL SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, pages 29–37, 1999.

[6] A. Fujii and T. Ishikawa. Applying machine translation to two-stage cross-language information retrieval. In *Proceedings of the 4th Conference of the Association for Machine Translation in the Americas*, pages 13–24, 2000.

[7] A. Fujii and T. Ishikawa. Japanese/English cross-language information retrieval: Exploration of query translation and transliteration. *Computers and the Humanities*, 35(4):389–420, 2001.

[8] A. Fujii, T. Ishikawa, and J.-H. Lee. Term extraction from Korean corpora via Japanese. In *Proceedings of the 3rd International Workshop on Computational Terminology*, 2004.

[9] J. Gonzalo, F. Verdejo, C. Peters, and N. Calzolari. Applying EuroWordNet to cross-language text retrieval. *Computers and the Humanities*, 32:185–207, 1998.

[10] K. S. Jeong, S. H. Myaeng, J. S. Lee, and K.-S. Choi. Automatic identification and back-transliteration of foreign words for information retrieval. *Information Processing & Management*, 35:523–540, 1999.

[11] K. Kishida, K. hua Chen, S. Lee, K. Kuriyama, N. Kando, H.-H. Chen, S. H. Myaeng, and K. Eguchi. Overview of CLIR task at the forth NTCIR workshop. In *Proceedings of the Fourth NTCIR Workshop*, 2004.

[12] K. Knight and J. Graehl. Machine transliteration. *Computational Linguistics*, 24(4):599–612, 1998.

Table 4. MAP values for different methods after the formal run (J-E and E-J were obtained with the Cross Language dictionaries and J-E* and E-J* were obtained with EDICT).

		Rigid			Relaxed		
Languages	Transliteration	T	D	D+N	T	D	D+N
J-E	No	0.2174	0.2171	0.2250	0.2952	0.2881	0.2901
J-E	Yes	**0.2182**	**0.2225**	**0.2261**	**0.3008**	**0.2943**	**0.2929**
J-E*	Yes	0.1147	0.0954	0.1151	0.1666	0.1334	0.1599
J-E*	No	**0.1383**	**0.1230**	**0.1410**	**0.1941**	**0.1670**	**0.1955**
J-K	No	0.2177	0.2151	0.2495	0.2376	0.2377	0.2764
J-K	Yes	**0.2457**	**0.2363**	**0.2696**	**0.2681**	**0.2613**	**0.2998**
E-J	No	0.1250	0.1135	0.1447	0.1673	0.1519	0.1965
E-J	Yes	0.1250	**0.1247**	**0.1474**	0.1673	**0.1640**	**0.2004**
E-J*	Yes	0.0612	0.0402	0.0141	0.0847	0.0537	0.0212
E-J*	No	**0.0857**	**0.0557**	**0.0316**	**0.1189**	**0.0786**	**0.0474**
E-K	No	0.2026	**0.1712**	0.2116	0.2131	**0.1770**	0.2267
E-K	Yes	**0.2153**	0.1711	**0.2235**	**0.2265**	0.1769	**0.2393**
K-J	No	0.1486	**0.1376**	**0.1328**	0.2035	**0.1916**	**0.1824**
K-J	Yes	**0.1746**	0.0739	0.0756	**0.2397**	0.1226	0.1136
K-E	No	0.1017	**0.0865**	0.0836	0.1411	**0.1191**	0.1170
K-E	Yes	**0.1231**	0.0507	**0.0927**	**0.1705**	0.0817	**0.1235**

[13] J. S. Lee and K.-S. Choi. A statistical method to generate various foreign word transliterations in multilingual information retrieval system. In *Proceedings of the 2nd International Workshop on Information Retrieval with Asian Languages*, pages 123–128, 1997.

[14] M. L. Littman, S. T. Dumais, and T. K. Landauer. Automatic cross-language information retrieval using latent semantic indexing. In G. Grefenstette, editor, *Cross-Language Information Retrieval*, chapter 5, pages 51–62. Kluwer Academic Publishers, 1998.

[15] J. S. McCarley. Should we translate the documents or the queries in cross-language information retrieval? In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*, pages 208–214, 1999.

[16] D. W. Oard. A comparative study of query and document translation for cross-language information retrieval. In *Proceedings of the 3rd Conference of the Association for Machine Translation in the Americas*, pages 472–483, 1998.

[17] S. Robertson and S. Walker. Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 232–241, 1994.

[18] G. Salton. Automatic processing of foreign language documents. *Journal of the American Society for Information Science*, 21(3):187–194, 1970.