Experiments on Cross-language and Patent Retrieval at NTCIR-3 Workshop

Aitao Chen
School of Information Management and Systems
University of California at Berkeley, CA 94720-4600, USA
aitao@sims.berkeley.edu

Fredric C. Gey
UC Data Archive & Technical Assistance (UC DATA)
University of California at Berkeley, CA 94720-5100, USA
gey@ucdata.berkeley.edu

Abstract

The Berkeley group participated in the cross-language retrieval task and the patent retrieval task at the third NTCIR workshop. This paper describes our experiments on cross-language and patent retrieval. We present an automatic relevance feedback procedure for document ranking formula based on logistic regression, and a procedure for automatically extracting Chinese/Japanese translations of English words from search results returned from Internet search engines using English words as queries.

Keywords: *Chinese IR, Japanese IR, Korean IR, Cross-language IR, Relevance feedback, Translation extraction, and Patent retrieval.*

1 Introduction

At the NTCIR-3 workshop, the Berkeley group participated in the Cross-Language Retrieval Task (CLIR) and Patent Retrieval Task (Pat). For the CLIR task, we worked on all three tracks: *SLIR*, *BLIR*, and *MLIR*. This paper describes our experiments with monolingual and cross-language retrieval, and with patent retrieval. We will describe the relevance feedback procedure, and a procedure for automatically extracting Chinese or Japanese translations for English words from the search results returned by an Internet search engine when English words are submitted as queries. For the first time, we had the opportunity to perform cross-language retrieval from Chinese to Japanese, and Korean monolingual retrieval. Since Chinese and Japanese share some of the ideographs, directly mapping the Chinese characters into Japanese kanji may work well in Chinese-to-Japanese retrieval in the cases where many Chinese characters in the Chinese topics are the same as Japanese kanji characters. Readers are refered to [9] for an overview of the third NTCIR workshop, to [5] for an overview of the CLIR Task, and to [8] for an overview of the Patent Retrieval Task.

2 Document Ranking

A typical text retrieval system ranks documents according to their relevances to a given query. The documents that are more likely to be relevant are ranked higher than those that are less likely. In this section we briefly describe a logistic regression-based document ranking algorithm developed at Berkeley (Cooper et al. 1994). We used this document ranking algorithm for all the the retrieval runs reported in this paper. The log-odds (or the logit transformation) of the probability that document D is relevant with respect to query Q, denoted by $\log O(R|D,Q)$, is given by

$$\begin{aligned}\log O(R|D,Q) &= log\frac{P(R|D,Q)}{P(\overline{R}|D,Q)} \\ &= -3.51 + 37.4 * X_1 + 0.330 * X_2 + \\ &\quad -0.1937 * X_3 + 0.0929 * X_4\end{aligned}$$

where $P(R|D,Q)$ is the probability that document D is relevant to query Q, $P(\overline{R}|D,Q)$ the probability that document D is not relevant to query Q, which is 1.0 - $P(R|D,Q)$. The four explanatory variables X_1, X_2, X_3, and X_4 are defined as follows: $X_1 = \frac{1}{\sqrt{n}+1}\sum_{i=1}^{n}\frac{qtf_i}{ql+35}$, $X_2 = \frac{1}{\sqrt{n}+1}\sum_{i=1}^{n}\log\frac{dtf_i}{dl+80}$, $X_3 = \frac{1}{\sqrt{n}+1}\sum_{i=1}^{n}\log\frac{ctf_i}{cl}$, $X_4 = n$, where n is the number of matching terms between a document and a query, qtf_i is the within-query frequency of the ith matching term, dtf_i is the within-document frequency of the ith matching term, ctf_i is the occurrence frequency in a collection of the ith matching term, ql is query length, dl is document length, and cl is collection length. The relevance probability of document D with respect to query Q can be written as $P(R|D,Q) = \frac{1}{1+e^{-\log O(R|D,Q)}}$ in terms of log-odds of

the relevance probability. The documents are ranked in decreasing order by their relevance probabilities with respect to a query.

3 Relevance Feedback

The Berkeley document ranking formula has been in use for many years without blind relevance feedback. In this section we present a technique for incorporating blind relevance feedback into the logistic regression-based document ranking framework.

Two factors are import in relevance feedback. The first one is how to select the terms from top-ranked documents after the initial search, the second is how to assign weights to the selected terms with respect to the terms after the initial query. For term selection, we assume some top-ranked documents after the initial search are relevant, and the rest of the documents in the collection are irrelevant. For each term in the documents that are presumed relevant, after removing stopwords, we compute its relevance weight. The relevance weight proposed by Robertson and Sparck Jones in [11] is given by

$$w_t = log\frac{R_t(N - N_t - R + R_t)}{(R - R_t)(N_t - R_t)} \qquad (1)$$

The terms are shown in the following word contigency table.

	relevant	irrelevant	
indexed	R_t	$N_t - R_t$	N_t
not indexed	R - R_t	N - N_t - R + R_t	N - N_t
	R	N - R	N

where N is the number of documents in the collection, R the number of top-ranked documents after the initial search that are presumed relevant, R_t the number of documents among the R top-ranked documents that contain the term t, and N_t the number of documents in the collection that contain the term t.

The terms extracted from the R top-ranked documents are ranked by their relevance weights. A prespecified number of top-ranked terms are combined with the initial query to create a new query. Note that some of the selected terms maybe among the initial query terms. For the selected terms that are not in the initial query, the weight is set to 0.5. For those selected terms that are in the initial query, the weight is set to 0.5*t_i, where t_i is the occurrence frequency of term t in the initial query. The selected terms are merged with the initial query to formulate an expanded query. When a selected term is one of the query terms in the initial query, its weight in the expanded query is the sum of its weight in the initial query and its weight assigned in the term selection process. For a selected term that is not in the initial query, its weight in the final query is the same as the weight assigned in the term selection process, which is 0.5. The weights for the initial query terms that are not in the list of selected terms remain unchanged. Table 1 presents an example to illustrate how the expanded query is created from the initial query and the selected terms. The numbers in parentheses are term weights. The selected new terms are considered not as important as the initial query terms, so the weights assigned to them should fall in the range of 0 to 1, exclusive. In our implementation, we set the weights of the new terms to 0.5, expecting that the query length would be doubled after query expansion.

Initial Query	Selected Terms	Expanded Query
t_1 (1.0)		t_1 (1.0)
t_2 (2.0)	t_2 (2*0.5)	t_2 (3.0)
t_3 (1.0)	t_3 (1*0.5)	t_3 (1.5)
	t_4 (0.5)	t_4 (0.5)

Table 1. Query expansion.

Three minor changes are made to the blind relevance procedure described above. First, a constant of 0.5 was added to every item in formula 1 used to compute the weight. Second, the selected terms must occur in at least 3 of the top-ranked R documents. Third, the top-ranked two documents in the initial search remained as the top-ranked two documents in the final search. The rationale for not changing the top-ranked few documents is that when a query has only a few relevant documents in the entire collection and if they are not ranked in the top after the initial search, it is unlikely these few relevant documents would be risen to the top in the second search since most of the documents that are presumed relevant are actually irrelevant. On the other hand, if these few relevant documents are ranked in the top after the initial search, after expansion, they are likely to be ranked lower in the final search for the same reason. We believe a good strategy is to not change the ranking of the top few documents. In our implementation, we chose not to change the ranks of the top two documents in the final search. Note that in computing the relevance probability of a document with respect to a query in the initial search, the ql is the number of terms in the initial query, and qtf_t is the number of times that term t occurs in the initial query. After query expansion, qtf_t is no longer the raw term frequency in the initial query, instead it is now the weight of term t in the expanded query, and ql is the sum of the weight values of all the terms in the expanded query. For the example presented in table 1, qtf_{t_3} is 1.5, and ql is 6.0 (i.e., 1.0 + 3.0 + 1.5 + 0.5). The relevance clues related to documents and the collection are the same in computing relevance probability using the expanded query.

4 Cross-Language Retrieval Task

The cross-language retrieval task has three tracks: single language IR (SLIR), bilingual CLIR (BLIR), and multilingual CLIR (MLIR). The document collections consist of newspaper articles in Chinese, Japanese, Korean, and English, published during the period of between 1998 and 1999 except that the *Korean Economic Daily* in 1994. Readers are referred to [5] for an overview of the CLIR task and details on the the tracks, documents, topics, and evaluations of the CLIR task. We participated in all three tracks in the CLIR task and submitted Chinese, Japanese, Korean, and English monolingual runs for the SLIR track; English-to-Japanese, English-to-Chinese, and Chinese-to-Japanese runs for the BLIR track; and Chinese-to-Chinese/Japanese/English, and English-to-Chinese/Japanese/English runs for the MLIR track. For all the runs in the CLIR task, the average precisions and overall recalls were computed using the set of *rigid* relevant documents.

4.1 Single Language IR Track

4.1.1 Chinese Retrieval

The Chinese texts in documents were broken into single-character unigrams and overlapping two-character bigrams. Only the Big5 characters encoded in two-byte were retained. The topics were processed in the same way. A stoplist of 718 terms was used to remove stopwords. We submitted one official Chinese monolingual run, named Brkly-C-C-D-01, using only the *desc* field in the topics. The *desc* field is typically short, and almost all terms in the *desc* occur only once. However not all terms are equally important. To reflect the fact that some terms may be more useful than others in retrieval, we selectively doubled the term frequency for 10 terms in the original query. The terms in the original query were first ranked by their *average-tfidf* weight, a technique proposed by Kwok in [10]. Then the term frequencies for the top-ranked 10 terms (bigrams or unigrams) were doubled. The query after adjusting term weight was used for the initial search. After the initial search, the terms in the top-ranked 20 documents were ranked by their relevance weights computed using formula 1, and the top-ranked 50 terms were combined with the original query terms to formulate the expanded query, which was then used to retrieve 1000 documents from the collection for each topic. Without initial weight adjusting and query expansion, the average precision is 0.2048, and overall recall 1291/1928. With initial weight adjusting but no query expansion, the average precision is 0.2140, and overall recall 1288/1928. The average precision is 0.2738 and overall recall 1473/1928 with query expansion but no initial weight adjusting. With both initial weight adjusting and query expansion, the average precision of *Brkly-C-C-D-01* is 0.2847 and overall recall 1516/1928. While adjusting the term weights in the initial query made little difference in retrieval performance, relevance feedback improved the average precision by 33.69% without weight-adjusting, and 33.04% with weight-adjusting. Discarding letters encoded in one byte may have degraded the performance of topic 2 containing the term *WTO*, and topic 22 containing the term *Pol Pot*.

For the official run, the Chinese texts were split into unigrams and bigrams. We also indexed the Chinese texts in short words of one to three characters. The Chinese texts were split into words using the forward maximum matching technique with respect to a list of 194,000 short Chinese words. The average precision is 0.2089 for the initial search, and 0.2780 with relevance feedback. In both runs, the initial query term weights were not adjusted. For the latter run, 20 top-ranked terms from the top-ranked 20 documents were combined with the initial query to create the expanded query. The results suggest that short word indexing is as effective as unigram-and-bigram indexing.

4.1.2 Japanese Retrieval

The Japanese texts were split into single-character unigrams and overlapping two-character bigrams consisting of only *kanji* and *katakana* characters. All *hiragana* characters were discarded, so were the Roman letters that are encoded in single byte. One official run named Brkly-J-J-D-01 was submitted which used the *desc* field only. The average precision of Brkly-J-J-D-01 is 0.3255 and overall recall 1533/1654, with initial weight adjusting and query expansion. Without query expansion, the average precision is 0.2802 and overall recall 1416/1654. Not indexing the English words may have degraded the performances of topic 2 containing *WTO*, topic 5 containing *PRC*, topic 9 containing *ST1*, topic 41 containing *NGO*, and topic 42 containing *EU*.

To compare different indexing methods, we created a word index after segmenting the Japanese texts into words using the *Chasen* morphological analyzer. The average precision using *desc* field is 0.2758 without relevance feedback, and 0.3188 with relevance feedback. In both runs, no weight-adjusting based on *average-tfidf* was applied. The performance of word indexing and that of unigram-and-bigram indexing suggest that both indexing methods are equally effective.

4.1.3 Korean Retrieval

We removed the blank spaces between words and treated the Korean texts as a string of characters. The texts were then divided into single-character unigrams and overlapping two-character bigrams. Our Korean stoplist consists of the most frequent 97 bigrams and the most frequent 15 unigrams found in the document

collection. The average precision is 0.1549 and overall recall 1365/2081 with initial weight adjusting but no query expansion. With both initial weight adjusting and query expansion, the average precision of Brkly-K-K-D-01 is 0.2269, an increase of 46.48%, and overall recall 1617/2081.

4.1.4 English Retrieval

The English words were stemmed using Porter stemmer after stopwords were removed. We submitted three English monolingual runs, Brkly-E-E-C-01, Brkly-E-E-TDN-02, and Brkly-E-E-D-03. The average precision is 0.4054 for Brkly-E-E-C-01, 0.4156 for Brkly-E-E-TDN-02, and 0.4111 for Brkly-E-E-D-03. For these three runs, we did not adjust the term frequency but applied pseudo relevance feedback. The top-ranked 30 terms selected from top-ranked 20 documents after the initial retrieval were combined with the original query to create the expanded query.

4.2 Bilingual CLIR Track

4.2.1 English-Chinese Retrieval

The English-to-Chinese IR subtask is about searching English topics against the Chinese document collection for relevant documents. The English topics were translated into Chinese using the on-line *Babelfish* translation available at http://babelfish.altavista.com/. The untranslated English words were looked up in an English-Chinese bilingual dictionary created from a collection of Chinese-English parallel texts, the Hong Kong News downloaded from www.info.gov.hk. More details on the sentence alignment of the parallel texts and the creation of English-Chinese bilingual dictionaries are provided in our earlier work [2, 3]. The topmost-ranked Chinese term was selected as the translation of an English word. The translated Chinese texts were then split into single-character unigrams and two-character overlapping bigrams. We submitted one official run named *Brkly-E-C-D-01* that used *desc* field only. The average precision is 0.1282, and the overall recall 1176/1928. The untranslated English words or phrases include *anguish*, *Dae-Jung* in *Kim Dae-Jung*, *doomsday*, *El nino*, *famines*, *James Soong*, *Kazuhiro Sasaki*, *Macau*, *Medecins Sans Frontieres*, *Nissan Motor Company*, *Oscar*, *Pol* in *Pol Pot*, *Renault*, *Rong* in *Zhu Rong ji*, *Takeshi Kitano*, *Taoyan*, *Titanic*, and *Tomiich Murayama*. Most of the untranslated words are proper nouns. In our earlier work [4], we proposed a technique to automatically extract Chinese translations for English words from the search results of Internet search engines using English words as queries. Here we present a slightly different version of the procedure originally proposed in [4]. First we submit each of the untranslated English words or phrases as query to the search engine of Yahoo!Chinese in traditional Chinese (Big5 encoding) at http://chinese.yahoo.com/. If the search results have more than 200 entries, we keep the first 200 search result entries, otherwise we keep all the entries. The search result entries are then segmented into words using a dictionary-based longest matching method. For each line containing the English query word or phrase, we only consider the five Chinese words immediately to the left, and the five Chinese words immediately to the right of the English word or phrase. We assign a weight of $\frac{1}{n}$ to a Chinese word that is n words away from the English word or phrase, so the Chinese word immediately to the left or to the right of the English word or phrase receives a weight of 1.0. We accumulate the weight values assigned to the same word in all search result entries. At the end, all the Chinese words that are within five-word distance of the English query word or phrase are ranked by their accumulated weights. To translate English to Chinese, we keep the top-ranked m Chinese words as the translation of the English word or phrase that was used as query, where m is the same as the number of words in the English query. Figure 1 shows the Chinese translations automatically extracted from Yahoo!Chinese search results using the untranslated English words or phrases as search queries. The number of words in Chinese translations is the same as the number of words in the English query. We performed an English-to-Chinese run by replacing the untranslated English words or phrases with the Chinese translations automatically extracted from Yahoo!Chinese search results. This run is labeled E-C-D-02 as shown in Table 2. When the untranslated

run id	translation resources	average precision	overall recall
E-C-D-00	Babelfish	0.1226	1190/1928
Brkly-E-C-D-01	Babelfish + parallel texts	0.1282	1176/1928
E-C-D-02	Babelfish + Yahoo!Chinese	0.1668	1177/1928

Table 2. Performances of three English-to-Chinese CLIR runs using *desc* field only.

English words or phrases were replaced by the Chinese translations extracted from Chinese!Yahoo search results, the average precision increased from 0.1226 to 0.1668, an improvement of 36.05%. The average precision for E-J-C-02 run is 0.1000 without relevance feedback, and 0.1668 with relevance feedback, an increase of 66.8%.

4.2.2 English-Japanese Retrieval

We used the same online Babelfish translation to translate English topics into Japanese. The untranslated English words were not further looked up in any other

	English query	Chinese translations		
1	anguish	痛苦		
2	Dae-Jung	金大中		
3	doomsday	末日		
4	El nino	聖嬰	現象	
5	famines	提		
6	James Soong	宋楚瑜	前	
7	Kazuhiro Sasaki	水手	浩	
8	Macau	澳門		
9	Medecins Sans Frontieres	無	醫生	國界
10	Nissan Motor Company	通報		
11	Oscar	奧斯卡		
12	Pol	終將		
13	Renault	雷諾		
14	Rong	榮		
15	Takeshi Kitano	武	野	
16	Taoyan	說		
17	Titanic	鐵達尼號		

Figure 1. Chinese translations automatically extracted from Yahoo!Chinese search results. The number of words in the Chinese translations is the same as in the English query.

machine translation system or bilingual dictionary. Most of the untranslated words are proper nouns, including personal names such as *Zhu Rong ji*, *James Soong*, *Kazuhiro Sasaki*, *Takeshi Kitano*, *Tomiich Murayama*, *Kim Dae-Jung*, *Clinton*, and *Pol* in *Pol Pot*. Other untranslated proper nouns include *Han* in Han dynasty, *Taoyan*, *Kyoto*, *Oscar*, *Titanic*, *El Nino*, *Renault*, *Nissan Motor Company*, *Medecins Sans Frontieres*, *Macau*. The other untranslated words are *sightseeing*, *doomsday*, *Anti-personnel*, *famines*, *collaborations*, and *anguish*. We submitted only one official run named *Brkly-E-J-D-01* using the *desc* field. The average precision is 0.1899 with an overall recall of 1066/1654. We submitted each untranslated English word or phrase as a query to the search engine of Yahoo!Japan at http://www.yahoo.co.jp/. We downloaded up to 200 search result entries for each query. The result entries were then segmented into words using Chasen morphological analyzer. The Japanese words surrounding the English word were weighted and ranked as described in the previous section, and the top-ranked two translations were retained. Words consisting of only hiragana characters were ignored. The procedure of automatically extracting Japanese translations from Yahoo!Japan search results is the same as described in section 4.2.1. Figure 2 shows the first two Japanese translations automatically extracted from Yahoo!Japan search result entries. The column labeled *English query* shows the list of English words that were submitted as queries to the Yahoo!Japan search engine. After we translated the original English topics into Japanese using the online Babelfish, we replaced the untranslated English words or phrases by the top-ranked two Japanese translations automatically extracted from Yahoo!Japan search results. We performed another run using this version of Japanese translation. The average precision is 0.2625 with an overall recall of 1455/1654 as shown in table 3. The average precision was increased by 38.23%. For

run id	translation resources	average precision	overall recall
Brkly-E-J-D-01	Babelfish	0.1899	1066/1654
E-J-D-02	Babelfish+ Yahoo!Japan	0.2625	1455/1654

Table 3. Performances of two English-to-Japanese CLIR runs using *desc* field only.

the second run, the precision for topics 2, 4, and 5 are 0.0316, 0.0000, and 0.0002, respectively. An important word *WTO* in topic 2 was discarded in indexing because the English words were not indexed. Topic 4 contains *E-Business* where only *Business* was translated and *E* was discarded, which is probably why the precision was zero for topic 4. The proper name *Zhu*

	English query	Japanese translations	
1	anguish	苦痛	太
2	anti-personnel	地雷	対人
3	Clinton	大統領	クリントン
4	collaborations	マイス	コラボレーションズ
5	doomsday	審判	日
6	El nino	エルニーニョ	現象
7	famines	大学	嘆く
8	Han	アッ	韓
9	James Soong	愉	選
10	Kazuhiro Sasaki	佐々木	浩
11	Kim Dae-Jung	大中	金
12	Kyoto	京都	京都大学
13	Macau	マカオ	澳
14	Medecins Sans Frontieres	師団	医
15	Murayama	村山	氏名
16	Nissan Motor Company	業	種別
17	Oscar	オスカー	アライアンス
18	Pol	方程式	ファイル
19	Renault	ルノー	車
20	Tomiich	名前	
21	Zhu Rong ji	Not found	
22	sightseeing	観光	情報
23	Takeshi Kitano	武	北野
24	Taoyan	桃園	組織
25	Titanic	タイタニック	救助

Figure 2. Japanese translations automatically extracted from Yahoo!Japan search results.

Rong ji was not translated in topic 5. The Japanese names for *rice* and *U.S.* happen to be the same, which probably results in the zero precision for topic 34 on rice import policy in Asian countries. The low precision of 0.0654 for topic 22 may be attributed to the incorrect translation of *Pot* in *Pol Pot* into the Japanese word that means rounded earthen or metal container.

4.2.3 Chinese-Japanese Retrieval

The Chinese texts in the *desc* field were translated into Japanese in two steps using English as the intermediate language. First the Chinese texts were segmented into words, then each word was looked up in the Chinese-English bilingual dictionary created from the same collection of Hong Kong News articles as described in section 4.2.1. Only the topmost-ranked English word was retained as the translation of a Chinese word. Second, the translated English words were subsequently translated into Japanese using the online Babelfish translation. The untranslated words were not further processed. A single Chinese-to-Japanese run named *Brkly-C-J-D-01* using *desc* field was submitted. The average precision is 0.1189, and overall recall 810/1654.

Since some of the Japanese kanji and traditional Chinese characters share the same ideographs, direct mapping from Japanese kanji into Chinese may work well in the cases where Japanese topics consist of mainly kanji characters. Of course, when the same concept or proper noun like *Asia* is expressed in katakana in a Japanese topic, direct mapping from Japanese into Chinese is of no use. Another case where direct mapping does not work is when a concept is expressed in kanji characters that is different from the Chinese characters for the same concept. For example, the Japanese kanji characters for *film* are different from the Chinese word for *film*. We converted the Chinese topics in Big5 into Japanese in EUC-JP in two steps, first converting the Chinese topics in Big5 into Unicode (UTF-8), then converting the Unicode into Japanese in EUC-JP. We used the Japanese topics converted from the Chinese topics for retrieval. This run is labeled *C-J-D-02*. The average precision of C-J-D-02 is 0.1109, which is as good as the official run

which used Babelfish and parallel corpus as translation resources. When the Japanese topics translated using Babelfish and parallel corpus were concatenated with the Japanese topics converted from the Chinese topics, the average precision increased to 0.1927 as shown in table 4. Topic 31 is about viewing Japanese maple trees in *Kyoto*, but the *desc* field in the Chinese version means viewing Japanese maple trees in *Tokyo*.

run id	translation resources	average precision	overall recall
Brkly-C-J-D-01	Parallel + Babelfish	0.1189	810/1654
C-J-D-02	BIG5-EUC	0.1109	835/1654
C-J-D-03	Parallel + Babelfish BIG5-EUC	0.1927	1276/1654

Table 4. Performances of three Chinese-to-Japanese CLIR runs using *desc* field only.

4.3 Multilingual retrieval track

4.3.1 English-Chinese/Japanese/English Retrieval

We submitted one multilingual run named *Brkly-E-CJE-D-01* using *desc* field in the English topics. This run was produced by combining three retrieval runs: one English monolingual run, one English-Chinese bilingual run, and one English-Japanese bilingual run. The English-Chinese bilingual run is *Brkly-E-C-D-01*, and the English-Japanese bilingual run is *Brkly-E-J-D-01*. We did not use *Brkly-E-E-D-01* as the English monolingual run, instead performed another English monolingual run using *desc* field. We will call this run *E-E-D-01*. The average precision values for *E-E-D-01*, *Brkly-E-C-D-01*, and *Brkly-E-J-D-01* are 0.3660, 0.1282, and 0.1899, respectively. The results of these three runs were combined and re-ranked by the probability of relevance. The final result consists of the top-ranked 1000 documents per topic. The average precision for the *Brkly-E-CJE-D-01* run is 0.1287, and overall recall 2067/4053.

4.3.2 Chinese-Chinese/Japanese/English Retrieval

The *Brkly-C-CJE-D-01* run was produced by combining *Brkly-C-C-D-01*, *Brkly-C-J-D-01*, and *C-E-D-01*. The C-C and C-J runs were discussed in previous sections. In performing Chinese-to-English retrieval, the Chinese texts in the *desc* field were segmented into words, then the Chinese words were looked up in an Chinese-English bilingual dictionary created from the Hong Kong News parallel texts. For each Chinese word, only the topmost-ranked English translation was retained. The English translation was used to produce the *C-E-D-01* run. For the initial run, the weight for the top-five terms ranked by their average tfidf value was doubled. For relevance feedback, top-ranked 30 terms from top-ranked 20 documents were combined with the initial query. The average precision for *Brkly-C-C-D-01*, *Brkly-C-J-D-01*, and *C-E-D-01* are 0.2847, 0.1189, and 0.2522, respectively. The average precision for *Brkly-C-CJE-D-01* is 0.1462, and overall recall 2111/4053.

5 Patent Retrieval Task

We took a conventional approach to patent retrieval and treated the patent test collection (both topics and documents) as another test collection. We applied the same set of techniques to patent retrieval as those for Japanese text retrieval and English-Japanese bilingual retrieval using Japanese newspaper articles or abstracts. The same retrieval system described in section 2 was also used for all the retrieval runs reported below. We submitted four official runs for the Patent Retrieval Task, two using the mandatory topic fields, *ARTICLE* and *SUPPLEMENT*, and two using optional fields, *DESCRIPTION* and *NARRATIVE*. The four official runs are labeled as *brklypat1*, *brklypat2*, *brklypat3*, and *brklypat4*. All other runs are unofficial runs. The average precisions and overall recalls reported for all the runs for the patent retrieval tasks were computed with respect to the strict relevance. On the average, after removing stopwords, the full-text patent documents in the *kkh98* and *kkh99* collections are about 21 times as long as the newspaper articles in the *Mainichi* collection used for the CLIR task. Another feature in patent retrieval task that is missing in Ad Hoc retrieval with newspaper document collections is that the topic field *ARTICLE* is the clipping of a newspaper article. The texts in the *ARTICLE* field are long and contain many words that are not important. Among the main questions we investigated in patent retrieval are:

1. is word indexing as effective as bigram indexing for long documents?
2. is retrieval from much shorter patent abstracts as effective as that from the full-text patent documents?
3. is stemming and splitting long katakana words helpful in retrieval?
4. is retrieval using long queries as effective as using short queries? and
5. is query expansion effective with long patent documents?

To save some space in presenting the results, we will use the initial letter of a topic field to represent that field, so *A* stands for *ARTICLE*, *C* for *CONCEPT*, *D* for *DESCRIPTION*, *H* for *HEADLINE*, *N* for *NARRATIVE*, *S* for *SUPPLEMENT*, *T* for *TITLE*. For all four official runs, the collections used are *kkh98* and *kkh99*, consisting of 697,262 full-text Japanese patent applications published in 1998 and 1999. Readers are referred to [8] for details on the task, the collections, the topics, and the evaluation of patent retrieval.

Both documents and topics were indexed using overlapping bigrams consisting of only katakana and kanji characters. The four official runs were produced using the bigram index.

5.1 Monolingual patent retrieval

run id	topic fields	overall recall	average precision
brklypat1	A,S	849	0.1547
brklypat2	D,N	1029	0.2236
brklypat5	C,D	1110	0.2404
brklypat6	C,D,N,T	1131	0.2505

Table 5. Summary of monolingual patent retrieval runs using overlapping bigram indexing.

We submitted two official monolingual patent retrieval runs, labeled as *brklypat1* and *brklypat2*. The texts were split into overlapping bigrams consisting of only kanji and katakana characters. A small stoplist of 159 words was used to remove stopwords in both documents and topics indexing. Table 5 presents the results of four monolingual runs without query expansion. The average precision of the required run which used the *ARTICLE* and *SUPPLEMENT* fields was substantially lower than that of using other topic fields, such as the *DESCRIPTION* and *NARRATIVE* fields. Table 6 shows the performances of monolingual patent retrieval runs using word indexing without query expansion. The texts in the patent documents and topics were segmented into words using the Chasen morphological analyzer. The words, after removing stopwords, were not stemmed. Among the runs presented in table 6, the run using *CONCEPT* and *DESCRIPTION* fields achieved the highest average precision of 0.3129. As with bigram indexing, the performance of the run using the *ARTICLE* and *SUPPLEMENT* fields was substantially inferior than any of the runs without using the *ARTICLE* field. In our experiments, we simply treated the texts in the *ARTICLE* field as a very long query without making any effort to identify and then remove the topic words that are not important. In our experiments with Japanese monolingual retrieval

run id	topic fields	overall recall	average precision
brklypat6	A	648	0.1230
brklypat7	C	1115	0.2374
brklypat8	S	730	0.1693
brklypat9	A,S	742	0.1482
brklypat10	C,D	1168	**0.3129**
brklypat11	D,N	1044	0.2480
brklypat12	D,S	987	0.2577
brklypat13	H,S	780	0.1888
brklypat14	C,D,T	1170	0.2980
brklypat15	C,D,N,T	1173	0.2849

Table 6. Summary of monolingual patent retrieval runs using word indexing. Words were not stemmed.

from the document collection consisting of newspaper articles, indexing by overlapping bigrams and unigrams together was as effective as indexing by words. The results presented in tables 5 and 6 show that word indexing was substantially better than bigram indexing when the *ARTICLE* field was not used. For example, the run bkypat5 using the *CONCEPT* and *DESCRIPTION* fields with bigram indexing has an average precision of 0.2404, while the run bkypat10 using the same topic fields with word indexing has an average precision of 0.3129, an increase of 30.16%.

Our stemmer removes any hiragana characters from a word, including the ones appearing in the middle of a word. So the stem of a word consisting of hiragana and kanji characters will contain only the kanji characters. A word consisting of only hiragana characters will be deleted. The full-text Japanese patent document collection has about 1.7 million unique words (not stemmed) after segmentation using the Chasen analyzer. About 921,000 of the unique words are katakana words having 8 or more characters. Most of the long katakana words are formed by joining two or more short katakana words. The long katakana words in Japanese are like the compound words in German. We have used the German decompounding procedure described in our earlier work [1] to break up long katakana words into short katakana words. The base dictionary has all the katakana words found in the full-text patent documents that are 3 to 7 characters long. The katakana words having 8 or more characters were split, if possible, into short katakana words in the base dictionary. Figure 3 presents an example of segmenting the long katakana word for the English phrase “computer network system.” It lists all the ways in which this katakana word can be segmented into short katakana words with respect to the base dictionary. The last column shows the probability of a segmentation which is computed as the product of the relative frequencies of the component words in

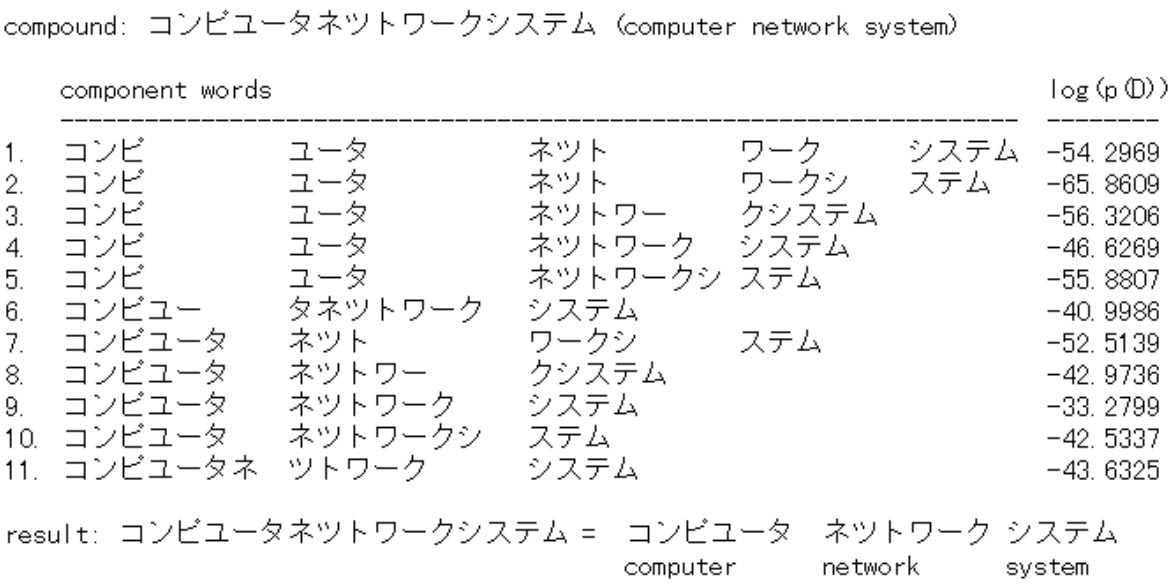

Figure 3. Segmentation of a long katakana word.

the full-text patent document collections after segmentation using the Chasen analyzer. The segmentation having the highest probability is chosen to segment a long katakana word. Table 7 presents the results of five monolingual runs using the index created after removing hiragana characters and segmenting long katakana words into short ones. The average precision values are close to those using the index without stemming and katakana words segmentation.

run id	topic fields	overall recall	average precision
brklypat16	C	1120	0.2409
brklypat17	A,S	748	0.1528
brklypat18	C,D	1187	0.3041
brklypat19	D,N	1059	0.2436
brklypat20	C,D,N,T	1174	0.2912

Table 7. Summary of monolingual patent retrieval runs using word indexing. Words were stemmed and long katakana words segmented.

We carried out two monolingual retrieval runs with query expansion, one using the *CONCEPT* field only, the other using both *CONCEPT* and *DESCRIPTION* fields. The first run is labeled brklypat25, and the second run brklypat26. For query expansion, 10 terms were selected from the top-ranked 5 documents after the initial search. The words were stemmed and long katakana words were split into short katakana words. The average precision of brklypat25 is 0.2233, which is slightly lower than 0.2409 of bkylypat16 without query expansion. The average precision of brklypat26 is 0.3043, which is almost the same as 0.3041 of brklypat18 without query expansion. The results of these two experiments show that query expansion did not improve retrieval performance. A plausible explanation is that it is more difficult to select the appropriate terms for query expansion in the term selection process since the average patent document length is about 21 times as long as that for the newspaper documents used in the CLIR task.

A word index was created for the Japanese abstracts for 1998 and 1999. The average document length is about 138 words for the Japanese newspaper collection used in the Cross-language task, about 100 words for the Japanese patent abstracts for 1998 and 1999, and about 2868 words for the full-text Japanese patent documents for 1998 and 1999. All the Japanese texts were segmented using the Chasen analyzer, and average document length was computed after removing stopwords. The average full-text patent document is about 29 times as long as the average patent abstract after removing stopwords. Table 8 presents the performances of the four runs using the word index created from only abstracts. The monolingual performances

run id	topic fields	overall recall	average precision
brklypat21	A,S	853	0.1370
brklypat22	C,D	950	0.1799
brklypat23	D,N	839	0.1407
brklypat24	C,D,N,T	927	0.1623

Table 8. Summary of monolingual patent retrieval runs using word index created from abstracts. Words were stemmed and long katakana words segmented.

of using abstracts only were substantially poorer than that of using full-text patent documents.

5.2 Cross-language Patent Retrieval

We created an English-Japanese dictionary from the English and Japanese abstracts published from 1995 to 1997. The abstracts were first split into sentences, then sentences were aligned using a modified version [2] of the length-based algorithm proposed in [7]. A small but important modification to the length-based algorithm is that the lengths of the Japanese sentences are scaled before sentence alignment so that the length ratio of the Japanese texts over the translated English texts is close to one. The Japanese sentences were segmented into words using Chasen morphological analyzer. About 3.7 million English/Japanese sentence pairs and 1 million English/Japanese titles were produced from the parallel abstracts for 1995 to 1997. An associative English-Japanese dictionary was created from the aligned English-Japanese sentence/title pairs based on word co-occurrence. We used the association measure described in [6] to compute the association strength between an English word and a Japanese word. We refer readers to [3] for more details on the construction of bilingual associative dictionaries from parallel texts. To translate English topics to Japanese, we looked up each English word in the English-Japanese dictionary, and kept only the topmost-ranked Japanese translation. The translated Japanese topics were used to search against the full-text Japanese patents to produce the final runs, *brklypat3* and *brklypat4*. These two runs used the overlapping bigram index. The performances of our official English-Japanese patent retrieval runs are presented in Table 9. The results for additional English-Japanese bilingual runs using the word index with stemming are presented in table 10. The performance of English-Japanese bilingual retrieval is substantially inferior to that of Japanese monolingual retrieval. As table 10 shows, the best English-Japanese bilingual performance is around only half of the monolingual performance. Figure 4 presents some of the problems with translating English topics into Japanese. The *English word* column shows English words or phrases found in the English topics, the *Japanese translation* column shows the Japanese translation from the bilingual dictionary for an English word or phrase, and the *Japanese in original topics* column shows the Japanese word or phrase found in the original Japanese topics for an English word or phrase. (The patent English topics are manually translated from the original Japanese topics).

run id	topic fields	topic language	overall recall	average precision
brklypat3	A,S	English	565	**0.0607**
brklypat4	D,N	English	815	**0.0827**

Table 9. Summary of official English-Japanese bilingual runs for Patent Retrieval Task. The bigram index was used in these two runs.

run id	topic fields	topic language	overall recall	average precision	% of mono
brklypat27	A,S	English	573	0.0547	35.79%
brklypat28	C,D	English	880	0.1493	50.66%
brklypat29	D,N	English	799	0.1234	49.09%

Table 10. Additional English-Japanese bilingual runs. The word index with stemming was used for the runs presented in this table.

We translated the English topics into Japanese by looking up the English topic words, after removing stopwords, individually in the bilingual associative English/Japanese dictionary and retained one Japanese word for each English topic word. So the translation model is essentially a word-for-word one. The failure cases shown in Figure 4 demonstrate that the word-for-word model is not adequate for translation from English to Japanese. There are cases where an English phrase should be collectively translated into a single Japanese word like cases 3 to 6 shown in Figure 4, and cases where a single English word is translated into a Japanese phrase like the case 11 shown in Figure 4. There maybe even cases where an English phrase should be collectively translated into a Japanese phrase. As an example, there is a single Japanese word for the English phrase *body temperature*. But in the word-for-word model, the words in the phrase *body temperature* are individually translated into Japanese, resulting in two Japanese words which are not the same as the single Japanese word for the whole phrase. Although it is possible sometimes to derive the single Japanese word for the English phrase from the Japanese translations of the individual English words, it is by no means easy. The English phrase *waste oil* in case 4, like the English phrase *body temperature* in case 3, should be collectively translated into Japanese. The phrase *electric motor* in case 6 should be collectively translated into Japanese. It is possible to derive the correct Japanese translation (a single word) from the two Japanese translations of the individual words in cases 3 to 5, but not in case 6 since the Japanese translation for the word *electric* is a kanji word, and the Japanese translation for the word *motor* is a katakana word while the correct translation is a kanji word. The case 7 *soft drink* illustrates that the phrase has to be translated collectively into Japanese, since the individual word *soft* cannot be properly translated into Japanese in this case. The word *soft* in the context of the phrase *soft drink* is like the word *real* in

	English word	Japanese translation	Japanese in original topics
1	motor	モータ	モータ
2	agitation	撹拌	攪拌
3	body temperature		体温
	body	本体	
	temperature	温度	
4	waste oil		廃油
	waste	廃棄物	
	oil	油	
5	electromagnetic wave		電磁波
	electromagnetic	電磁	
	wave	波	
6	electric motor		電動機
	electric	電気	
	motor	モータ	
7	soft drink		清涼飲料水
	soft	軟質	
	drink	飲料	
8	business	業務	ビジネス
9	diesel	ディーゼルエンジン	ディーゼル
10	liquid	液晶	液体
11	photocatalyst	触媒	光 触媒
12	cosmetic	料	化粧品
13	totalnitrogen		全窒素

Figure 4. Some of the problems in English to Japanese translation.

the phrase *real estate*. Sometimes the Japanese translation is a katakana word, but the original Japanese is a kanji word. The opposite cases also occur, such as the case 8 in Figure 4 where the original Japanese word is a katakana word, but the Japanese translation from the English word *business* is a kanji word. The case 9 illustrates that the compound katakana words should be split into short ones. The case 10 shows the original Japanese word and its Japanese translation from English are semantically close. The case 11 shows that the English word *photocatalyst* should be translated into two Japanese words. The Japanese translation in case 12 is simply wrong. And case 13 is a misspelling. A space should be inserted between the two words, *total* and *nitrogen*. The case 1 shows that different characters is used to denote a long vowel in katakana words. The case 2 show that different spellings of the same kanji character are in use. It is like the case where both traditional and simplied forms of the same character are in use in Chinese texts. The list of problems in translating English into Japanese is by no means exhaustive.

6 Conclusions

We have presented a pseudo relevance feedback procedure for the logistic regression-based document ranking algorithm. The performance improvement brought by relevance feedback ranges from a few points to 66.80%. For all of our official runs in the cross-language retrieval track, the Chinese, Japanese, and Korean texts are indexed using single-character unigrams and overlapping two-character bigrams. The retrieval performance on Chinese, Japanese, and Korean monolingual retrieval shows the simple unigram-and-bigram indexing is effective for all three languages. The performance on short Chinese words index is as good as that on unigram-and-bigram index. The word-based indexing for Japanese works equally well as unigram-and-bigram indexing. We have described a revised procedure for automatically extracting Chinese or Japanese translations for English words from the results returned from a search engine when the English words are submitted as queries. When this procedure is combined with the online Babelfish translation, the performances for both English-to-Chinese

and English-to-Japanese are substantially improved. For Chinese-to-Japanese retrieval, we found combining Chinese-to-Japanese character conversion with machine translation in translating Chinese topics into Japanese significantly improved the performance over using either technique alone.

Our experimental results for patent retrieval show that using the *DESCRIPTION* and *NARRATIVE* fields was more effective than using the long *ARTICLE* and the *SUPPLEMENT* fields; not using the *ARTICLE* field worked better than using it; full-text patents worked better than patent abstracts; word indexing worked better than bigram indexing; stemming and splitting long katakana words did not help; query expansion did not help; and English-Japanese bilingual patent retrieval was substantially worse than Japanese monolingual patent retrieval.

7 Acknowledgements

We would like to thank Vivien Petras for carrying out the English monolingual retrieval experiments. This research was supported by DARPA under research grant N66001-00-1-8911 as part of the DARPA Translingual Information Detection, Extraction, and Summarization Program (TIDES).

References

[1] A. Chen. Multilingual information retrieval using english and chinese queries. In C. Peters, M. Braschler, J. Gonzalo, and M. Kluck, editors, *Evaluation of Cross-Language Information Retrieval Systems*, volume Lecture Notes in Computer Science, Vol. 2406, pages 44–58, Berlin, 2002. Springer-Verlag.

[2] A. Chen, F. Gey, and H. Jiang. Alignment of english-chinese parallel corpora and its use in cross-language information retrieval. In *9th International Conference on Computer Processing of Oriental Languages*, Seoul, Korea, May 14-16 2001.

[3] A. Chen, F. Gey, and H. Jiang. Berkeley at ntcir-2: Chinese, japanese, and english ir experiments. In *Proceedings of the Second NTCIR Workshop on Research in Chinese & Japanese Text Retrieval and Text Summarization.*, pages 137–145, Tokyo, Japan, March 2001.

[4] A. Chen, H. Jiang, and F. Gey. Combining multiple sources for short query translation in chinese-english cross-language information retrieval. In *Proceedings of the Fifth International Workshop on Information Retrieval with Asian Languages, Sept. 30-Oct 1, 2000, Hong Kong*, pages 17–23, 2000.

[5] K. Chen, H. Chen, N. Kando, K. Kuriyama, S. Lee, S. Myaeng, K. Kishida, K. Eguchi, and H. Kim. Overview of clir task at the third ntcir workshop. In *this volume*.

[6] T. Dunning. Accurate methods for the statistics of surprise and coincidence. *Computational linguistics*, 19:61–74, March 1993.

[7] W. A. Gale and K. W. Church. A program for aligning sentences in bilingual corpora. *Computational linguistics*, 19:75–102, March 1993.

[8] M. Iwayama, A. Fujii, N. Kando, and A. Takano. Overview of patent retrieval task at ntcir-3. In *this volume*.

[9] N. Kando. Overview of the third ntcir workshop. In *this volume*.

[10] K. L. Kwok. A new method of weighting query terms for ad-hoc retrieval. In *Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Zurich, Switzerland, August 18-22, 1996*, pages 187–195, 1996.

[11] S. E. Robertson and K. S. Jones. Relevance weighting of search terms. *Journal of the American Society for Information Science*, pages 129–146, May–June 1976.