Cross-Language Information Retrieval for NTCIR at Toshiba

Tetsuya Sakai [*] Yasuyo Shibazaki [†] Masaru Suzuki, Masahiro Kajiura,
Toshihiko Manabe and Kazuo Sumita
Toshiba R&D Center

Abstract

Toshiba submitted six runs in the NTCIR crosslingual task. Our basic approach was translation of Japanese search requests into English using a commercial machine translation system. Runs TSB1, TSB5 and TSB6 used the ASTRANSAC machine translation system, while, for comparison, TSB3 used queries manually constructed by a bilingual. TSB2 is TSB1 plus local feedback, and TSB4 is TSB3 plus local feedback. Therefore, only TSB1, TSB2, TSB5 and TSB6 can be considered as *automatic* runs. Runs TSB1 through TSB4 used the NEAT information filtering system based on the BM25 probabilistic retrieval model. Runs TSB5 and TSB6 used the vector-space search module of the KIDS information sharing system, the latter using its rule-based natural language analysis function. All runs used *short* queries, that is, they used as input the *description* fields only.

1 Introduction

Toshiba submitted six runs in the NTCIR crosslingual task. Our basic approach was translation of Japanese search requests into English using a commercial machine translation(MT) system. Although dictionary-based approaches and corpus-based approaches seem to be more popular than MT-based approaches in cross-language information retrieval (CLIR) [2] [4] , our previous CLIR experiments with English and Japanese showed that MT can be very effective [20] .

Runs TSB1 through TSB4 used the NEAT[1] information filtering system [8] [17] based on the BM25 probabilistic retrieval model [15] [21] . NEAT was originally developed for a commercial online news filtering service launched in 1996 [13] and is now also used for a Japanese WWW search/filtering service [3] . The work of the NEAT team, including our previous findings on MT-based CLIR, will be explained in Section 2.

Runs TSB5 and TSB6 used the search module of the KIDS[2] information sharing system [11] [12] . The goal of KIDS is to provide an office environment where knowledge of individual workers as well as information from various databases can easily be accessed and utilized. The KIDS search module is based on the vector-space model, but it can also rank documents using structured indexes constructed through natural language processing. The work of the KIDS team will be explained briefly in Section 3.

Table 1: Description of Toshiba Runs (short)

RunID	type	system	
TSB1	auto	NEAT	
TSB2	auto	NEAT	TSB1 + local feedback
TSB3	man	NEAT	
TSB4	man	NEAT	TSB3 + local feedback
TSB5	auto	KIDS	
TSB6	auto	KIDS	TSB5 + rule-based natural language analysis

Runs TSB1, TSB5 and TSB6 used the ASTRANSAC machine translation system [5] [6] for translating the search requests,[3] while, for comparison, TSB3 used queries manually constructed by a bilingual. TSB2 is TSB1 plus local feedback, and TSB4 is TSB3 plus local feedback. TSB6 is the same as TSB5 except that it used the rule-based natural language analysis function of KIDS. Therefore, only TSB1, TSB2, TSB5 and TSB6 can be considered as *automatic* runs. All six runs used *short* queries, that is, they used as input the *description* fields only. Table 1 provides a summary of our runs.

2 TSB1 - TSB4

Section 2 is organized as follows. Subsection 2.1 describes the NEAT information filtering system. Subsection 2.2 summarizes our previous work related to the NTCIR crosslingual task. Subsection 2.3 provides a detailed study of the NTCIR task using the training requests, and Subsection 2.4 provides the corresponding results with the official test requests. Finally, Subsection 2.5 concludes this section.

2.1 The NEAT Information Filtering System

2.1.1 The Original NEAT

The original NEAT system was developed for filtering online Japanese news articles. NEAT calculates the score of each document based on *document structure*, that is, it generates term frequency vectors for various document compo-

[*] Sections 1&2: tetsuya.sakai@toshiba.co.jp

[†] Section 3: yasuyo.shibazaki@toshiba.co.jp

[1] News Extractor with Accurately Tailored profiles

[2] Knowledge and Information on Demand System

[3] The NEAT team used the UNIX-based basic ASTRANSAC, while the KIDS team used a GUI-based version of ASTRANSAC. Therefore the results of query translation may be slightly different.

nents such as the *full text*, *heading*, *first sentence*, and *first paragraph*, and calculates a weighted average of the vector similarities [16] [17] .

For retrieval of Japanese texts, NEAT combines character-based matching and morpheme-based matching to avoid matching problems caused by nonexplicit word boundaries that are characteristic of Japanese texts. More details can be found in [17] and also in our IREX paper [19] . For retrieval of English texts, which is more relevant to our Japanese-to-English CLIR via translation of search requests, NEAT simply performs Porter stemming. NEAT also offers thesaurus-based term expansion options, but they were not used for NTCIR. The original NEAT is now being used for commercial services [13] and [3] .

2.1.2 The Probabilistic NEAT

The latest version of NEAT [7] uses the BM25 probabilistic retrieval model [15] [21] , although currently it cannot fully utilize the document structure information. However, using standard Japansese test collections, we discovered that the probabilistic NEAT outperforms the original NEAT when *flat* queries are used, that is, when document structure information is not exploited. Therefore, for NTCIR as well, we used the probabilistic NEAT with flat queries.

The probabilistic NEAT calculates the term weight $tw(t,d)$ for a term t and a document d as follows:

$$tw(t,d) = \frac{\log(|C|/df(t)) * tf(t,d) * (K+1)}{K * ((1-b) + (b * L(d) * |C| / \sum_{d \in C} L(d))) + tf(t,d)} \quad (1)$$

where C is the document collection;
$L(d)$ = length of the document d in bytes;
$tf(t,d)$ = number of occurrences of the term t within d;
$df(t)$ = number of documents in C containing the term t;
K = empirically selected constant for controlling the effect of $tf(t,d)$;
b = empirically selected constant for controlling the effect of $L(d)$.

High values of K imply that frequent terms are important terms, while $K=0$ implies that it is only term *presence* that matters. High values of b imply that long documents are verbose, while low values imply that they are multitopic. The final document score is the sum of the term weights.

2.2 Our Previous Work Related to the Task

2.2.1 Monolingual IR

In [16] and [18] , we showed that the probabilistic NEAT achieves retrieval performance of the highest standard to date using the BMIR-J2 standard Japanese test collection [10] . We performed query expansion through relevance and local feedback by using the following term selection criterion, referred to as the offer weight [15] [21] :

$$ow(t) = rdf(t) * \log \frac{\frac{rdf(t)+0.5}{|R|-rdf(t)+0.5}}{\frac{df(t)-rdf(t)+0.5}{|C|-df(t)-|R|+rdf(t)+0.5}} \quad (2)$$

where R is the set of relevant documents;
$rdf(t)$ = number of documents in R containing the term t.

In our query expansion experiments with BMIR-J2, the offer weight outperformed other term selection criteria such as those based on within-document term frequency. The optimal *downweighting factor* w for the expansion terms was found to be 0.2. Thus the framework of the probabilistic retrieval model seemed to transfer well to Japananese, even though the gain by local feedback over the initial performance in terms of 11-point average precision was only 5%. The optimal number of pseudo-relevant documents (n) and the optimal number of expansion terms (m) were found to be around 5 and 10 respectively. We also showed that higher values of n and m hurt performance.

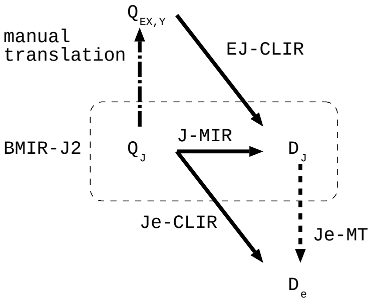

Figure 1: CLIR experiments using BMIR-J2.

In [20] , we repeated the above experiment to provide a monolingual baseline for our CLIR experiments prior to NTCIR, explained in 2.2.2. This time, we performed *term reweighting* in addition to query expansion by replacing the log component in Equation (1) (inverse document frequency) with that in Equation (2) (relevance weight), which resulted in a 6% gain in retrieval performance. Moreover, we improved the performance further by grouping the initial queries according to the number of search terms they contain and optimizing n and m for each group instead of the whole query set. This was based on a hypothesis that the amount of information that needs to be supplemented through local feedback depends on how much information the initial query contains. However, we decided not to adopt this strategy in NTCIR because the number of the training requests was relatively small.

2.2.2 Cross-Language IR

In [20] , we studied English-to-Japanese and Japanese-to-English CLIR using the probabilistic techniques described in 2.2.1 plus the ASTRANSAC machine translation system.

Figure 1 outlines the CLIR experiments we conducted. For our English-to-Japanese CLIR (EJ-CLIR) experiments, the Japanese search requests Q_J of BMIR-J2 were translated separately by two bilingual researchers X and Y into English requests Q_{EX} and Q_{EY} respectively. We then compared the following two approaches to EJ-CLIR:

(a) Translation of the documents: Translating the documents D_J of BMIR-J2 into English documents D_e using ASTRANSAC and then using Q_{EX} and Q_{EY} with D_e, thus reducing the problem to English retrieval. The small "e" in D_e implies that the documents are the output of Japanese-to-English MT and are *not* English documents, strictly speaking.

(b) Translation of the requests: Translating Q_{EX} and Q_{EY} into Japanese requests Q_{jX} and Q_{jY} using ASTRANSAC and then using them with D_J, thus reducing the problem to Japanese retrieval. The small "j" implies that the queries are the output of English-to-Japanese MT.

In terms of overall performance, Approach (a) outperformed Approach (b), which is a finding in agreement with [14] where English-to-German CLIR was studied. It is generally acknowledged that, because documents are generally longer than requests, Approach (a) can provide more context to enhance translation quality. Moreover, because there are only a small number of search terms, the failure to translate these terms in Approach (b) can be fatal at the retrieval stage. However, (a) is applicable only if the size of the document collection is reasonably small or if the language familiar to the user is known in advance.

Our best EJ-CLIR performance was more than 90% of the monolingual baseline. This was impressive, given that the typical performance of CLIR between European languages in TREC-6 was 50-75% and that EJ-CLIR has to overcome the problems of different character sets and ambiguity in Japanese text segmentation. On the other hand, we showed that the overall results with human translators X and Y differ substantially due to differences in the choice of words, even though their translations seemed equally good. This meant that the traditional measure of CLIR based on a monolingual baseline can be greatly affected by how the requests are manually translated for simulating CLIR.

In our Japanese-to-English CLIR (JE-CLIR) experiments, which were regarded as preliminary experiments for NTCIR, we only considered Approach (b) by simply using Q_J with D_e aforementioned. That is, we conducted "Je-CLIR" experiments by using the MT output instead of real English documents, say D_E. Q_J was translated into English requests Q_e using ASTRANSAC. We expected this to provide an upperbound for our JE-CLIR approach because:

(i) JE-CLIR is reduced to matching Q_e with D_E;

(ii) Je-CLIR is reduced to matching Q_e with D_e.

And it is highly likely that (ii) is an easier task than (i), since Q_e and D_e are the output of the same MT system.

Our Je-CLIR results were as good as the monolingual baseline. Although we showed that *pre-translation expansion* [1] using a separate document collection was also effective, we only used *post-translation expansion* for NTCIR due to lack of linguistic resource.

For both EJ-CLIR and Je-CLIR, we tried to analyze the relationship between the quality of machine translation and the retrieval performance on a query-by-query basis. Although the translation quality was reflected in the retrieval performance in general, we also found that there were many outliers. Closer analyses suggested that CLIR involving Japanese is more problematic than CLIR between European languages, one problem being the choice of word segmentation strategy.

It was difficult to compare our EJ-CLIR results and Je-CLIR results, not only because the latter was somewhat artificial, but also because the current version of the Japanese-to-English ASTRANSAC was not as good as its English-to-Japanese counterpart in terms of dictionary coverage etc.[4] However, the overall standard of our CLIR experiments encouraged us to participate in the JE-CLIR task for NTCIR.

2.3 Experiments with the NTCIR Training Requests

Henceforth, both A-relevant documents and B-relevant documents will be treated as "relevant."

2.3.1 Query Formulation

Two sets of initial queries were generated as follows:

TSB1: The description fields of the NTCIR Japanese training requests were translated into English using ASTRANSAC. About 10 new phrases, such as "knowledge acquisition" and "unification grammar" were added to the Japanese-to-English MT dictionary in order to enhance the translation quality. Note that this dictionary tuning was allowed only for the training requests and not for the test requests in the case of an *automatic* run. Finally, the translated English requests were converted into flat queries using a simple stopword list.

TSB3: A bilingual IR researcher read the description fields of the NTCIR Japanese training requests and directly constructed queries with English terms, without using a dictionary. Note that he did not translate the whole sentences, and that the process of converting English sentences into queries was by-passed. Thus, provided that the manual translation was accurate, TSB3 can be considered as the best case of TSB1 with both successful MT and successful stopword filtering.

The probabilistic parameters K and b were optimized for TSB1, resulting in $K = 1.5$ and $b = 0$. These values were reused for all other runs. Since the NTCIR documents are *abstracts* of technical papers, they are unlikely to be verbose, so $b = 0$ (switching off document length normalization) seemed to make sense.

The TSB2 queries were generated through local feedback as in [20] , by treating TSB1 as the initial run. The optimal values of the local feedback parameters for TSB2 were found to be $n = 10$ and $m = 20$. These values were then reused to generate the TSB4 queries by treating TSB3 as the initial run. The downweighting factor $w = 0.2$ from [16] [18] was reused here.

2.3.2 Other Considerations

Prior to the preparation of TSB3, we tried constructing *long* manual queries from the training requests, by using information from the *narrative* fields as well as the description fields. However, they were discarded as their average performance turned out to be lower than that of TSB1. This was possibly because we did not make a distinction between terms from the description fields and those from the narrative fields. Downweighting the terms from the narrative fields might have been effective. Nevertheless, we feel that *short* queries are closer to reality where information needs are neither well-defined nor well-expressed.

We also considered an alternative to local feedback mentioned in [9] , which we called *keyphrase feedback*. *Keyphrases* are words or phrases assigned to a technical paper abstract by its author. Such information is explicitly tagged and available for all NTCIR documents. If the keyphrases roughly reflect the contents of the paper, it is reasonable to assume that they may be of use for enhancing retrieval performance. In keyphrase feedback, we extracted all keyphrases from the top n documents of the initial ranked output and used them as expansion terms as in local feedback. However, this

[4] The MT dictionaries were *not* tuned for the CLIR experiments with BMIR-J2.

method was also discarded because of its poor performance. There seemed to be at least three reasons for the failure:

1. *The author's choice of keyphrases is not necessarily reliable.* For example, we found that "natural language processing" had been assigned as a keyphrase in some NTCIR documents. While this phrase may be useful for categorizing documents according to general areas of research, it may be too vague to be of any use in retrieval. It seems that the purpose of manually assigning keyphrases had not been clarified when the authors filled out the NACSIS forms.

2. *Too many phrases.* Many keyphrases were indeed *phrases*, rather than single words. Because we treated each of them as a single term in our experiment, it might be the case that some relevant documents were missed. For example, the search term "document retrieval" may miss documents containing "retrieving documents" only, even though stemming is performed for each component word.

3. *Too many spelling mistakes!* Many of the keyphrases were misspelt. In fact, lots of misspellings were found among the expansion terms via local feedback as well. Obviously, misspelt terms are unlikely to contribute to retrieval performance unless many documents in the collection contain the same mistakes! Because most of the NTCIR English documents were written by Japanese researchers, they are *very* noisy. However, this does not discount the significance of NTCIR: it just reminds us that practical IR systems have to deal with problems such as this. Particularly in the context of CLIR or multilingual IR, we would be too naive to assume that every document in every language is a composition of a native speaker.

2.3.3 Results with the NTCIR Training Requests

Table 2 shows the 11-point average precision values for TSB1 through TSB4 averaged over the 21 training requests. For TSB2 and TSB4, it also shows the gain in retrieval performance over TSB1 and TSB3, respectively. Figure 2 shows the corresponding recall-precision curves. The following observations can be made:

1. Manual runs outperform automatic runs in terms of *average* performance.

Figure 3 shows the difference in 11-point average precision values between TSB3 and TSB1 on a query-by-query basis. This reveals that the overall difference between the *manual* run and the *automatic* run is affected by outliers. Since TSB1 actually outperforms TSB3 for 10 out of the 21 training requests, we cannot conclude that manual query formulation by a bilingual is more effective than machine translation plus stopword filtering.

Table 3 compares the terms used in TSB1 and TSB3 for some of the outliers in Figure 3. Although it is clear that the TSB1 queries contain automatically stemmed words while the TSB3 queries contain words that were manually entered without going through the stemming process, this is not a particularly relevant difference in this context because all the terms are stemmed *internally* in both cases.

In the manual queries, double quotation marks were used to treat phrases as single terms, and the plus signs were used to treat synonyms as single terms. The manual use of phrases and synonyms seems to have been effective for Request 0026, where the translation itself was successful for both TSB1 and TSB3. On the other hand, its effect is not clear for Request 0008, for which TSB1 actually outperformed TSB3.

For Request 0016, it is clear that the difference in translation quality caused the difference in retrieval performance.

Another difference between TSB1 and TSB3 is that the choice of terms is more *selective* in TSB3. That is, in the manual queries, only terms associated with the key concepts of the search requests were used. For example, Request 0008 in TSB1 contains "example,""newest" and "trend," all of which were discarded in TSB3. But since TSB1 outperformed TSB3 for 0008, it is not clear whether this selective strategy is a good idea in general or not.

2. Local feedback is not very effective.

In terms of the performance gain, local feedback in NTCIR seems less effective than it was in our small-scale Je-CLIR experiment with BMIR-J2 in [20], where we achieved a 12% gain from 0.409 to 0.457. The results are also disappointing when compared to our Japanese monolingual experiment for IREX, where local feedback achieved an 18% gain from 0.458 to 0.540 with a document collection as large as that of NTCIR [19] . It seems that there are at least three reasons that account for these differences:

(1) *NEAT is not yet fully adapted to retrieval of English texts.*

Figure 4 compares the 11-point average precision values before and after local feedback for the automatic runs, and Figure 5 provides the corresponding data for the manual runs. Tables 4 and 5 show the actual search terms for some of the extreme instances in the figures. These examples reveal that the current version of NEAT has a problem with retrieval of English texts. There are expansion terms such as "processflow," "thereason" and "expressionand," which are obviously results of concatenating two words. This was a bug in the morphological analysis module used at the indexing stage, which partly explains the difference between our NTCIR and IREX results.

(2) *Misspellings!*

We mentioned this problem in 2.3.2. In Tables 4 and 5, we see misspelt words such as "suitablil," "countermeashur," "capabilitii," "confition," "keywori," and "kovean." We suspect that this is the main cause of the difference between our NTCIR results and our preliminary Je-CLIR results, even though the performance gain is not directly comparable since the NTCIR collection is about 40 times larger than BMIR-J2.

(3) *The effect of document cutoff.*

In addition to the document collection size, the document cutoff value can affect both the absolute retrieval performance and the relative performance gain by local feedback. While NTCIR evaluates the top 1000 documents of each ranked output, IREX evalutates the top 300 documents only. In our IREX paper [19] , we showed that using a small document cutoff value can underestimate the absolute retrieval performance and *overestimate* the effect of local feedback. However, since our IREX results show a 16% gain even at *cutoff* = 1000, probably the above two reasons had a greater impact.

Table 2: 11-point average precision averaged over the training requests (*cutoff* = 1000).

Run	11pt	Gain
TSB3(man)	0.339	-
TSB4(man + local feedback)	0.330	-3%
TSB2(auto + local feedback)	0.302	4%
TSB1(auto)	0.291	-

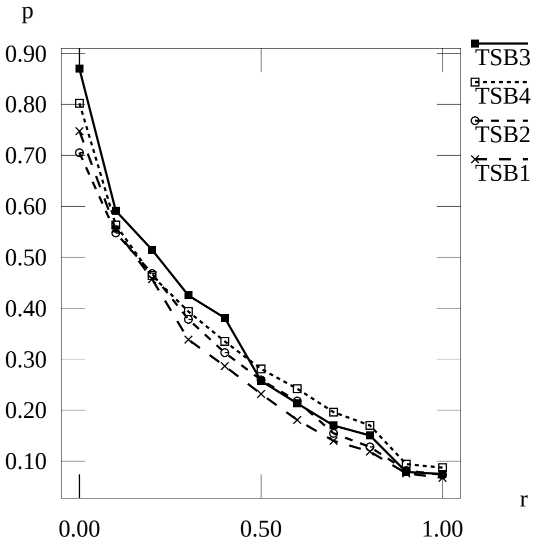

Figure 2: Recall-precision curves averaged over the training requests (*cutoff* = 1000)

Figures 4 and 5 indicate that our local feedback method is not very reliable for NTCIR. Nine queries in TSB2 and thirteen queries in TSB4 show performance degradation. Again, this is in contrast with our IREX results, in which the effect of local feedback was more consistent.

We did not conduct any monolingual baseline experiment for NTCIR. However, if we take the IREX Japanese monolingual results at *cutoff* = 1000 as the baseline, TSB1 over 1103a (comparison before local feedback) and TSB2 over 1103b (comparison after local feedback) are both 56%.

2.4 Results with the NTCIR Official Test Requests

Runs TSB1 through TSB4 for the test requests were generated as in 2.3, using the parameters optimized for the training requests. Table 6 shows the 11-point average precision values for TSB1 through TSB4 averaged over the 39 official test requests in the same way as in Table 2. Figure 6 shows the corresponding recall-precision curves.

The results are quite disappointing, with local feedback hurting performance in both TSB2 and TSB4. Moreover, the difference in retrieval performance between the manual runs and the automatic runs are greater. This is probably because, by definition of an automatic run, we did not allow ourselves to tune the MT dictionary for the test requests.

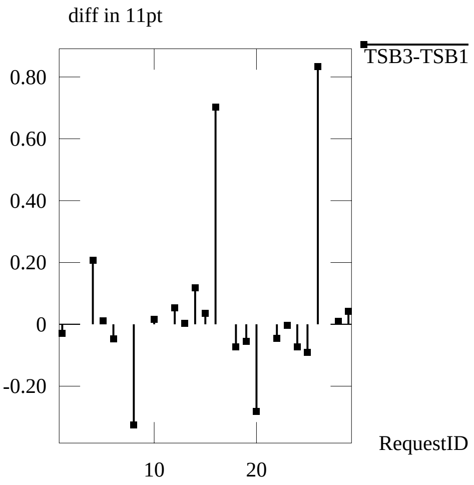

Figure 3: TSB3 (man) vs TSB1 (auto).

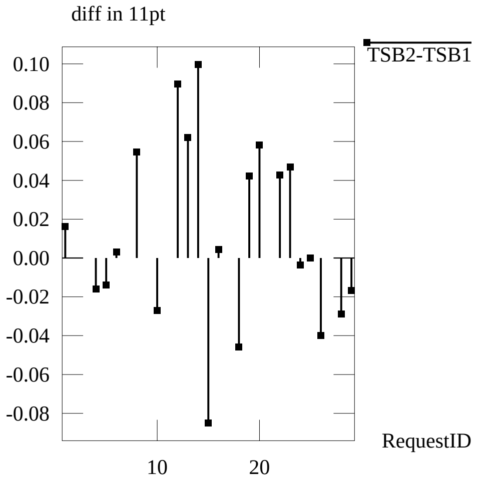

Figure 4: TSB2 (auto+LF) vs TSB1 (auto).

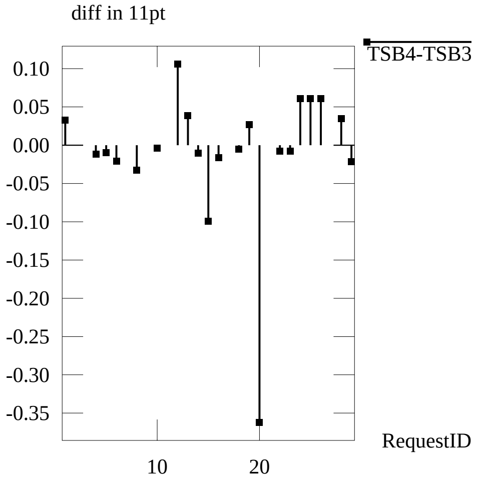

Figure 5: TSB4 (man+LF) vs TSB3 (man).

Table 3: Search terms used in TSB1 (auto) and TSB3 (man) for the training requests.

Request ID	terms(TSB1)	terms(TSB3)	11pt(TSB3-TSB1)
0008	algorithm, apriori, associ, data, example, improv, mine, newest, rule, trend	“data mining,” associative, rule, apriori+”a priori,” algorithm	-0.326
0016	graph, intersect, maximum, problem	largest, common, subgraph	0.702
0026	function, grammar, lexic	LFG+“lexical functional grammar”	0.833

Table 4: Search terms used in TSB1 (auto) and TSB2 (auto+LF) for the training requests.

Request ID	initial terms	expansion terms	11pt(TSB2-TSB1)
0014	diagnost, failur, system	oper, knowledg, diagnosis, diagnosi, fault, base, **suitablil**, recognizi, **processflow**, inabsorpt, dualimg, **countermeashur**, **capabilitii**, causal, process, sympotom, ntr, **confition**, **thereason**, plant	0.10
0015	automat, colloc, extract, text	inform, paper, new, gram, nonsens, keyword, string, japanes, corpus, exclusion, corpu, uniterrupt, morpholoiz, **keywori**, **expressionand**, statist, word, research, describ, ambigu	-0.08

Table 5: Search terms used in TSB3 (man) and TSB4 (man+LF) for the training requests.

Request ID	initial terms	expansion terms	11pt(TSB4-TSB3)
0020	Japanese, sentence, katakana, foreign, word	languag, verb, phrase, trigram, semant, text, translat, kana, use, newspap, testset, syllabiccharact, signsent, shinmeikai, sanseido, **kovean**, kokugojiten, keystork, jyoshi, dicitonari	-0.36

Table 6: 11-point average precision averaged over the official test requests (*cutoff* = 1000).

Run	11pt	Gain
TSB3(man)	0.249	-
TSB4(man + local feedback)	0.249	0%
TSB1(auto)	0.182	-
TSB2(auto + local feedback)	0.173	-5%

Thus there is a difference in the quality of MT between the training set and the test set[5] .

Figure 7 shows the 11-point average precision values of TSB3 and TSB1 on a query-by-query basis. Table 7 shows the terms used in TSB3 and TSB1 for Requests 0050 and 0058, as in Table 3. Again, the selective strategy in TSB3 seems to have been effective for Request 0058, but not for 0050. Thus the term “applic”(application) was important for Request 0050 but not for 0058.

2.5 Conclusions for TSB1 - TSB4

On the whole, we found the NTCIR crosslingual task quite problematic. Surely, we need to work more on NEAT to ensure retrieval quality of a high standard in a multilingual environment. Moreover, we need a robust mechanism for handling noises in the texts such as misspellings.

In the context of CLIR, the issue of how to present the retrieved information to the user becomes particularly important. This would involve machine translation, summarization and human-computer interaction. At the same time, a standard methodology for quantitative evaluation of such features would be in order. We plan to address these issues in the near future.

3 TSB5 and TSB6

3.1 Retrieval by KIDS using a Structured Index

KIDS is being developed for facilitating knowledge sharing at an office environment. Its search module retrieves documents and knowledge fragments by interpreting natural language search requests. The KIDS search module was originally developed for retrieval of Japanese texts, but recently it has been extended to handle English texts as well.

Most retrieval systems use only term occurrence information for ranking documents. In contrast, KIDS aims at a deeper understanding of documents and search requests to enhance retrieval performance. In addition to the traditional term occurrence information, KIDS utilizes a *structured index*, which consists of the following:

- A set of morphemes extracted from each document, excluding stopwords such as prepositions;
- Frequency and location information for each morpheme in each document;

[5] While many of the training requests were NLP-related, the test requests seemed more varied, including completely new topics such as molecular biology. Tuning the MT dictionary for the training requests is unlikely to be of any use in a case like this.

Table 7: Search terms used in TSB1 (auto) and TSB3 (man) for the official test requests.

Request ID	terms(TSB1)	11pt	terms(TSB3)	11pt
0050	applic,artifici,intellig,shogi	0.650	artificial,intelligence,shogi	0.280
0058	applic,current,law,relev,zipf	0.530	Zipf,law+rule	1.000

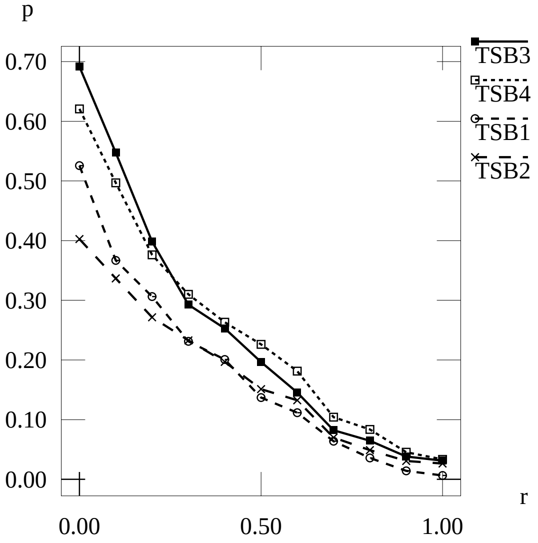

Figure 6: Recall-precision curves averaged over the official test requests (*cutoff* = 1000)

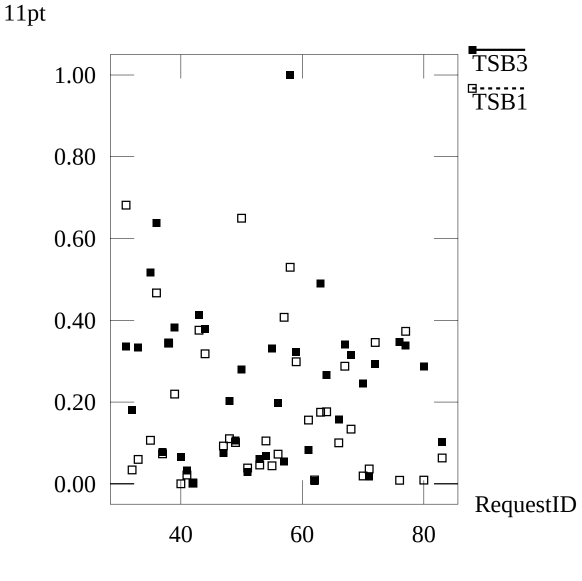

Figure 7: TSB3 (man) and TSB1 (auto).

- A set of tags automatically assigned to *simple sentences* extracted from each document. Simple sentences are sentence fragments obtained by decomposing a sentence by using punctuation marks and words such as "from" and "about" (mainly prepositions) as anchors.

The tags are usually used for indicating the *semantic roles* of the simple sentences, such as "situation", "requirement" and "reason." However, for NTCIR, we only used three kinds of tags: positive, negative and neutral. Twenty-four rules for tag assignment were formulated using the training requests. *Positive tags* were assigned to simple sentences containing expressions such as "this paper", "deal with" and "based on," which are likely to be used with important concepts in English technical papers. On the other hand, *negative tags* were assigned to simple sentences containing negative expressions such as "not", "no" and "without." All other simple sentences were tagged as *neutral.*

Our retrieval algorithm is as follows:

(1) Extract morphemes from the request;

(2) Extract and tag simple sentences as in the case of the documents;

(3) Add synonyms via dictionary lookup;

(4) Retrieve documents containing the morphemes extracted in (1) and (3);

(5) Calculate the similarity between the request and each document retrieved in (4), and rank the documents.

KIDS calculates the similarity $S_{q,d}$ for a request q and a document d as follows (t is a term in document d).

$$S_{q,d} = \sum_{t \in d} \delta_{t,q} W_{t,d} C_{t,q,d} \tag{3}$$

where $\delta_{t,q}$ is 1 if t occurs in q, otherwise 0;

$$W_{t,d} = ((1-p_1) + p_1 \frac{TF_{t,d}}{TF_d}) \log_2 \frac{D}{D_t} \tag{4}$$

p_1 = Constant($\geq 0, \leq 1$);
$TF_{t,d}$ = number of occurrences of t within d;
TF_d = maximum number of occurrences of any term within d;
D = total number of documents;
D_t = number of documents containing t;

$$C_{t,q,d} = C^1_{t,q} C^2_{t,q,d} \tag{5}$$

$$C^1_{t,q} = (1-p_2) + p_2 \frac{TF_{t,q}}{TF_q} \tag{6}$$

p_2 = Constant($\geq 0, \leq 1$);
$TF_{q,d}$ = number of occurrences of t within q;
TF_q = maximum number of occurrences of any term within q;

Table 8: 11-point average precision averaged over the training requests ($cutoff = 1000$).

Run	11pt	structured index
TSB6(auto)	0.251	YES
TSB5(auto)	0.238	NO

$$C^2_{t,q,d} = \max_{u \in d} \delta_{t,u} TC_{q,u}(1 + p_3(K_{q,u} - 1)) \qquad (7)$$

where $\delta_{t,u}$ is 1 if t occurs in a simple sentence u, otherwise 0;
$TC_{q,u}$ = *tag correlation* between q and u (≥ 0);
p_3 = Constant(≥ 0);
$K_{q,u}$ = number of terms from q within u.

Thus, in addition to the traditional $tf-idf$ based weighting, KIDS examines the tag correlation $TC_{q,u}$ between q and each simple sentence u in d. $TC_{q,u}$ is heuristically determined for each pair of tags. There were six values for $TC_{q,u}$ in the case of NTCIR, because both the request and the simple sentence could be either positive, negative or neutral. These values were optimized using the training requests and were used for TSB6. For TSB5, we let $TC_{q,u} = 1$ for all pairs of tags, thus switching off the tag correlation component to provide a baseline.

3.2 Results and Conclusions for TSB5 and TSB6

Table 8 shows the 11-point average precision values for TSB5 and TSB6 with the training requests. In terms of overall performance, the use of structured index seems effective. However, it hurt performance for six requests. One of the reasons was that, because simple sentences were extracted through pattern matching rather than deep natural language analysis, negative tags were assigned to some important simple sentences. Another reason was that term weighting was performed even for those terms that were unimportant from the viewpoint of retrieval. Discarding these terms would probably improve performance. We plan to resolve these problems and improve KIDS further.

References

[1] Ballesteros, L. et al.: Resolving Ambiguity for Cross-language Retrieval, *ACM SIGIR '98 Proceedings*, pp. 64–71 (1998).

[2] Braschler et al. : Cross-Language Information Retrieval(CLIR) Track Overview, TREC7, `http://trec.nist.gov/pubs/trec7/t7_proceedings.html` (1998).

[3] FreshEye : `http://www.fresheye.com`

[4] Grefenstette, G. (ed.) : *Cross-Language Information Retrieval*, Kluwer Academic Publishers, Norwell, Massachusetts (1998).

[5] Hirakawa, H. et al: EJ/JE Machine Translation System ASTRANSAC - Extensions towards Personalization, MT Summit III Proceedings, pp. 73–80 (1991).

[6] Kinoshita, S. et al.: ASTRANSAC - Toshiba Machine Translation System, MT Summit VI Proceedings, p. 284 (1997).

[7] Jones, G. J. F. et al. : Experiments in Japanese Text Retrieval and Routing using the NEAT System, *ACM SIGIR '98 Proceedings*, pp. 197–205 (1998).

[8] Kajiura, M. et al. : Development of the NEAT Information Filtering System (in Japanese), *IPSJ 54th National Conference Proceedings*, pp. 3–(299–300) (1997).

[9] Kando, N. et al.: Cross-Lingual Information Retrieval using Automatically Generated Multilingual Keyword Clusters, *Proc. IRAL '98*, pp. 86–94 (1998).

[10] Kitani, T. et al. : Lessons from BMIR-J2: a Test Collection for Japanese IR Systems, *ACM SIGIR '98 Proceedings*, pp. 345–346 (1998).

[11] Nakayama, Y. et al. : Development of a Knowledge / Information Sharing System "Advice / Help on Demand" – Integration of Organized Office Knowledge and Personal Knowledge (in Japanese), *IPSJ Interaction '97 Proceedings*, pp. 103–110 (1997).

[12] Nakayama, Y. et al. : Knowledge Management – Development of a knowledge and information sharing system "KIDS" – (in Japanese), *JSAI 13th National Conference Proceedings*, pp. 124–127 (1999).

[13] NewsWatch : `http://www.newswatch.co.jp`

[14] Oard, D.W. et al.: Document Translation for Cross-Language Text Retrieval at the University of Maryland, TREC6, `http://trec.nist.gov/pubs/trec6/t6_proceedings.html` (1997).

[15] Robertson, S. E. et al.: Simple, Proven Approaches to Text Retrieval, Computer Laboratory, University of Cambridge (1994).

[16] Sakai, T. et al. : Application of Query Expansion Techniques in Probabilistic Japanese News Filtering, *Proc. IRAL '98*, pp. 46–55 (1998).

[17] Sakai, T. et al. : Generation and Evaluation of Search Queries Using Boolean Expressions and Document Structure for Information Filtering (in Japanese), *IPSJ Transactions*, Vol. 39, No. 11, pp. 3076–3083 (1998).

[18] Sakai, T. et al. : Query Expansion through Feedback in Japanese Information Filtering based on the Probabilistic Model (in Japanese), *IPSJ Transactions*, Vol. 40, No. 5, pp. 2429–2438 (1999).

[19] Sakai et al.: Probabilistic Retrieval of Japanese News Articles for IREX at Toshiba, *IREX-1 Proceedings*, to appear (1999).

[20] Sakai, T. et al. : A Study on English-to-Japanese / Japanese-to-English Cross-Language Information Retrieval using Machine Translation (in Japanese), *IPSJ Transactions*, submitted (1999).

[21] Sparck Jones, K. et al.: A Probabilistic Model of Information Retrieval: Development and Status, Computer Laboratory, University of Cambridge (1998).