Thomson Legal and Regulatory at NTCIR-3: Japanese, Chinese and English retrieval experiments

Isabelle Moulinier, Hugo Molina-Salgado, and Peter Jackson
Thomson Legal and Regulatory
Research and Development Group
610 Opperman Drive, Eagan, MN 55123, USA
{Isabelle.Moulinier,Hugo.Salgado,Peter.Jackson}@westgroup.com

Abstract

Thomson Legal and Regulatory participated in the CLIR task of the NTCIR-3 workshop. We submitted formal runs for monolingual retrieval in Japanese and Chinese, and for bilingual retrieval from English to Japanese. Our main focus was in Japanese retrieval. We compared word-based and character-based indexing, as well as query formulation using characters and character bigrams. Our results show that word-based and bigram-based retrieval show similar performance for most query formulation approaches, while they outperform character-based retrieval. For Chinese retrieval, we compared using single characters with using character bigrams. We also introduced a structured query to leverage both. Our results are consistent with previous work, where character bigrams were shown to have better performance than single characters. The structured query approach is promising, but requires more analysis. In our bilingual runs, queries were translated using a machine-readable dictionary. Translated terms were resegmented to match indexing units. Our results, so far, are inconclusive, as we experienced unexpected query formulation issues especially in our word-based approach.

Keywords: *word indexing, character and character bigram indexing, query formulation*

1 Introduction

For the NTCIR-3 workshop, Thomson Legal and Regulatory participated in the CLIR task and submitted runs for the following subtasks: monolingual Japanese retrieval, monolingual Chinese retrieval, and bilingual English to Japanese retrieval. For all these runs, we used the same retrieval engine, WIN which is an inference network engine similar to INQUERY.

Our main effort was focused on Japanese retrieval. Early work in Japanese text retrieval compared word-based and character-based indexing [4]. More recent approaches tend to prefer character bigrams (overlapping or not) over characters, but also consider words and phrases [7, 8]. Our runs compare word-based, character-based, and overlapping bigram-based indexing. When indexing is character or bigram based, we also vary query formulation, the process that identifies concepts in a natural language query and organizes these concepts into a structured query.

Since we had no prior experience with Chinese retrieval, we set out to compare retrieval using characters with retrieval using overlapping character bigrams. While indexing was the same in all cases, we used query formulation to restrict search to character or bigram units.

Our bilingual runs were from English queries to Japanese documents. We translated queries using a machine-readable dictionary, and kept multiple translations when they occurred. Translated terms were then re-segmented to match the segmentation performed during indexing, and we compared different query structures similar to the ones used in our monolingual Japanese runs.

We give some background to our experiments in Section 2. Sections 3, 4 and 5 respectively present our Japanese, Chinese and bilingual experiments.

2 Background

2.1 Previous research

Japanese, Chinese and multi-lingual retrieval have seen some interesting developments in recent years, thanks to workshops and conferences such as NTCIR, TREC and CLEF.

Because neither Japanese nor Chinese mark word boundaries in written text, one of the main issues with Japanese and Chinese retrieval is segmentation, i.e. the process of splitting text into words or more generally indexing units. Early work on Japanese retrieval by Fujii and Croft [4] compared characters and words as indexing units, and various query structures to group

characters or words into more meaningful concepts. Recent approaches for both Chinese and Japanese have introduced character bigrams as a good alternative to words [5, 2, 9] but with less focus on query structure.

Our approach to bilingual retrieval uses a machine-readable dictionary to translate query terms. By taking advantage of query structures available in INQUERY, Pirkola [10] has shown that, for European languages, grouping translations for a given term is a better technique than allowing all translations to contribute equally. Oard and Wang [9] build upon Pirkola's work by showing how the approach was also well suited for English/Chinese retrieval. One aspect of Oard and Wang's work that we revisit below is the effect of post-translation resegmentation.

2.2 The WIN system

The WIN system is a full-text natural language search engine, and corresponds to TLR/West Group's implementation of the inference network retrieval model. While based on the same retrieval model as the INQUERY system [3], WIN has evolved separately and focused on the retrieval of legal material in large collections in a commercial environment that supports both Boolean and natural language searches [11].

In addition, WIN has shifted from supporting mostly English content to supporting a large number of Western-European languages as well. This was performed by localizing tokenization rules and adopting morphological stemming. Moreover, WIN adopted Unicode as its internal character encoding. As a result of these improvements, we were able to integrate various segmentation methods for Japanese and Chinese that are not part of the production version of WIN.

2.2.1 Document Scoring

WIN supports various strategies for computing term beliefs and scoring documents. We used a standard tf-idf for computing term beliefs in all our runs. The document is scored by combining term beliefs using a different rule for each query operator [3]. The final document score is an average of the document score as a whole and the score of the best portion. The best portion is dynamically computed based on query term occurrences.

2.2.2 Query formulation

Query formulation identifies concepts in natural language text, and imposes a structure on these queries. In many cases, each term represents a concept, and a flat structure gives the same weight to all concepts. The processing of English queries eliminates stopwords and other noise phrases (such as "Find cases about", or "Relevant documents will include"), identifies (legal) phrases based on a phrase dictionary and detects common misspellings. When phrases or misspellings occur, the query structure is no longer flat, but include operators such as "natural phrase" (NPHR) and "synonym" (SYN).

In the experiments reported below, we used our standard English stopword and noise phrase lists, but did not identify phrases or misspellings. For Chinese and Japanese, we created a stopword list by identifying the most frequent indexing units in the collection, and by manually filtering these candidates. In addition, noise phrases were identified using the dry run topics.

Concept identification depends on text segmentation. In our experiments below, we follow two main definitions for a concept: a concept is an indexing unit (word, character, or character bigram), or a concept is a construct of indexing units. Constructs are expressed in terms of operators (average, proximity, synonym, etc.) and indexing units.

3 The Japanese retrieval subtask

During indexing, we used two different segmentation techniques. The first one is word-based and relies on ChaSen [6], a publicly available morphological analyzer for Japanese. All words identified and normalized by ChaSen were indexed. As a result, some Hiragana terms are indexed if they are identified by ChaSen. The second technique is character-based. Following Fujii and Croft [4], we use a change in alphabet to identify rough boundaries. Terms made of Hiragana characters were not indexed, since Hiragana is typically used for word inflections and functional words such as particles. Words consisting of Katakana or non-Kanji characters (such as English words) were indexed as a single unit. Sequences of Kanji characters were broken into single characters and overlapping character bigrams. Both character and character bigrams were indexed. Note that overlapping bigrams are bound by a change in alphabet. For instance, the following sequence $K_1K_2K_3hK_4K_5$, where K_i are Kanji characters and h an Hiragana sequence, generates K_1K_2, K_2K_3, and K_4K_5 but not K_3K_4.

Fujii and Croft [4] introduced four query types to group words or characters into more meaningful concepts. Their approach relied on part-of-speech tagging to determine compounds and noun phrases. Our experiments do not use part-of-speech tags, but rely on similar ideas, inasmuch as we group characters and bigrams into longer concepts.

We investigated the following query structures:

- flat_word: all words identified by ChaSen were grouped under a #SUM node. This corresponds to our formal run TLRRD-J-J-DC-02.
- flat_char: all indexing units (a single Kanji character or a sequence of Katakana or Latin charac-

ters) are grouped under the same #SUM node.

- flat_bi: same as flat_char, but with Kanji bigrams instead of single Kanji characters.
- phr_char: we keep each Kanji sequence as a single concept in the query by grouping each component character under a #NPHR (proximity of 3) node. Katakana sequences remain one concept under the top #SUM.
- phr_bi: same as phr_char, but with Kanji bigrams.
- sum_char: this is similar to phr_char, but we keep each Kanji sequence as a single concept in the query by grouping each component character under a #SUM node.
- sum_bi: same as sum_char, but with Kanji bigrams.
- phr_both: this introduces an additional level of structure in the query. We combine phr_bi and phr_char under a #SUM node. This is in the spirit of a back-off model, where single characters are used when bigrams do not appear. In our case, we use single characters even when bigrams are present. This corresponds to our formal run TLRRD-J-J-D-01.
- sum_both: same as phr_both, combining sum_bi and sum_char under a #SUM node. This corresponds to our formal run TLRRD-J-J-DC-03.

Using the #SUM operator instead of the #NPHR operator alleviates the proximity constraint and allows any of the component units to contribute to the score of a document.

3.1 Experimental Results and Discussion

The average precision for our formal runs are summarized in Table 1. Our best run, the word-based approach, has an average precision of 0.4104 using relaxed relevance (0.3380 using rigid relevance), and a recall of 0.8873 ($\frac{2252}{2538}$) using relaxed relevance (0.8809 using rigid relevance).

The results in Table 1 may be misleading . One may conclude that word-based retrieval outperforms by far character and character bigram retrieval. This is not the case, as can be seen in Table 2 which reports results for all query formulation approaches described above. In this comparison, all runs use the description and concepts fields in the topics. These results now show that word-based retrieval performs only slightly better than most of bigram-based approaches. However, retrieval based on single characters definitely does not perform as well.

Our experiments show that query structure and indexing units are dependent. A given query structure may not work as effectively for both single characters and character bigrams. For instance, using a proximity constraint with bigrams (run phr_bi) seems detrimental, while using a #SUM node, which averages the contribution of all its children, does not benefit single characters (run sum_char).

Finally, grouping characters and bigrams to account for longer concepts does not improve retrieval over flat queries. More analysis is required to assess whether both approaches retrieve the same documents or a complementary list of documents.

4 The Chinese retrieval subtask

This participation marked our first attempt at Chinese retrieval. Since we did not have access to a word segmentation tool, we followed the character and bigram approaches reported in past research. As we did in the Japanese segmentation, we benefited from punctuation marks and non-Chinese characters. Sequences of non-Chinese characters, e.g. English names, were kept as one indexing units. We used punctuation marks to constrain bigrams not to overlap across sentences or groups of terms.

Our query formulation was straightforward. We used:

- flat_char: all single characters and non-Chinese tokens were grouped under a #SUM node.
- flat_bi: all overlapping bigrams, constrained by punctuation and change in alphabet, and non-Chinese tokens were grouped under a #SUM node.
- struct_both: we combined single characters, bigrams and non-Chinese tokens into a single structured query. The structured query groups all single characters under a #SUM node, all bigrams under another #SUM node, but leaves non-Chinese tokens as children of the top #SUM node. This is similar to averaging runs flat_char and flat_bi. Our two formal runs follow struct_both.

4.1 Experimental Results and Discussion

Table 3 reports our formal and unofficial runs. The only difference between our two formal runs is the topic fields used. We would have expected concepts in the topics, which are clearly identified, to have more influence on retrieval performance.

Our Chinese results were average, when we compare them with all the runs that were submitted for the workshop. However, we were disappointed by our struct_both runs, namely TLRRD-C-C-DC-01 and TLRRD-C-C-D-02. In designing the struct_both runs,

Run ID	Topic Fields	Indexing units	Query structure	Avg Prec. (relaxed)	Avg Prec. (rigid)
TLRRD-J-J-D-01	D	characters and bigrams	phr_both	0.3115	0.2569
TLRRD-J-J-DC-02	D,C	word	flat_word	0.4104	0.3380
TLRRD-J-J-DC-03	D,C	characters and bigrams	sum_both	0.2804	0.2443

Table 1. Summary of our formal runs for the Japanese subtask. The table shows which topic fields were used, the segmentation and query formulation methods.

Run ID/Query structure	Relaxed			Rigid		
	Avg Prec.	R-Prec.	Doc. retrieved	Avg Prec.	R-Prec.	Doc. retrieved
flat_word	0.4104	0.3988	2252	0.3380	0.3317	1457
flat_char	0.3738	0.3746	1829	0.3260	0.3268	1203
flat_bi	0.3986	0.3954	2232	0.3325	0.3250	1469
phr_char	0.3672	0.3687	1870	0.3070	0.3167	1256
phr_bi	0.3802	0.3800	2215	0.3072	0.3101	1454
sum_char	0.3345	0.3390	1740	0.2955	0.2949	1081
sum_bi	0.4059	0.4036	2180	0.3442	0.3432	1406
phr_both	0.3854	0.3793	2047	0.3221	0.3167	1306
sum_both	0.2804	0.3000	1597	0.2443	0.2466	1020

Table 2. Summary of our query formulation runs in the Japanese subtask. The topic fields D,C were used for all runs. flat_word uses words as indexing units. All other runs use characters and bigrams. The number of relevant documents is 2538 using relaxed judgments, and 1654 using rigid judgments.

Run ID/Query structure	Relaxed		Rigid	
	Avg Prec.	Doc. retrieved	Avg Prec.	Doc. retrieved
TLRRD-C-C-DC-01	0.2413	2300	0.2077	1436
TLRRD-C-C-D-02	0.2185	2090	0.1686	1286
flat_char	0.1878	1920	0.1637	1234
flat_bi	0.2685	2215	0.2334	1381

Table 3. Summary of our Chinese runs. TLRRD-C-C-DC-01, flat_char and flat_bi used the D,C fields in the topics. TLRRD-C-C-D-02 used the D field only. The number of relevant documents with the relaxed judgment is 3284; it is 1928 with the rigid judgments

we intended to create a boosting effect, i.e. we expected that struct_both would rank documents high if they ranked high in either flat_char or flat_bi runs. However, our choice of query structure relying on the SUM operator exhibits little boosting effect. On the the contrary, its main effect is averaging. A per query analysis shows that struct_both is detrimental for 27 queries, and helpful for 15 queries. We still need to conduct a document level analysis. We are currently looking into alternative query structures and operators that would combine unigrams and bigrams without averaging their contributions.

The boosting effect may exist, but may influence recall rather than precision. Table 3 also reports the number of relevant documents, and we noticed that struct_both retrieves more relevant documents than the other approaches, i.e. it has a higher recall.

Our choice for the structured query gives more importance to non-Chinese terms than to Chinese characters and bigrams, as the non-Chinese terms are left as children of the top #SUM node. We have not yet determined which impact this has on the boosting effect. While only 20% of the queries have non-Chinese terms, these terms have high idf. An alternative to our current structure is to fold the non-Chinese tokens under both the character and bigrams subqueries, thus truely averaging runs flat_char and flat_bi at the document level.

5 The bilingual retrieval subtask

The Japanese collection was indexed using the same approach as in Section 3. Our main effort here was in query formulation. We used the JMDICT Japanese-English machine-readable dictionary (MRD) [1] and massaged dictionary entries to generate English-Japanese translations. Most entries contain Kanji or Katakana translations, as well as their transliteration in Hiragana. Dictionary entries contain both single (English) words, and multiple words/phrases.

After the usual stopword and noise phrase removal, we extract English phrases and words. If we find a translation for a phrase, we do not translate the phrase's individual components. This is an attempt to capture longer concepts in Japanese if they appear in the MRD.

Once we have translated each concept, resegmentation is required so that query terms and indexed terms will match. Our word-based resegmentation relies on ChaSen, as indexing did. We use both the Kanji and Hiragana fields from the MRD. During training, we noticed that, without proper context, ChaSen broke some sequences into shorter units, which was not the case during indexing. As a result, we investigated grouping resegmented translations with the translations themselves. Resegmented translations were grouped under a #SUM node in our formal run. We are currently experimenting with #NPHR nodes, but do not have results at this time.

When we segmented translated terms into both characters and character bigrams, we used the sum_both and phr_both approaches from the Japanese retrieval subtask. At this point we did not attempt to resegment using the other approaches based on bigrams only.

All runs group multiple translations under a #SUM node.

5.1 Experimental results and discussion

The bilingual subtask includes our weakest runs, summarized in Table 4. Our word-based runs retrieved little to no relevant documents. The second run based on word indexing did not including Hiragana fields. While the average precision of that run is very low, we do notice an improvment in the number of relevant document retrieved.

We have identified several problems with our word-based runs. First, we find that using ChaSen for tokenization and normalization is context-dependent, and we observe this behavior to be very pronounced for Hiragana sequences. As a result, search units may not match indexing units. Next, we identify query structure to also be a problem. We relied on a structure that proved detrimental in our Japanese experiments. The use of the SUM node is especially harmful when there is a mismatch between indexing and search units: the contribution of the units that indeed match has little effect on the final score because the score averaging performed by the SUM node.

Our other runs, based on characters and bigrams were significantly better, although more work is required to achieve an acceptable performance. These runs are also negatively influenced by our choice of query structure. While we observe a large difference between runs phr_both and sum_both in the Japanese retrieval subtask, the difference in average precision for the bilingual task is not significant. We are not able to explain at this point the differences in the number of relevant documents retrieved.

Finally, we noted that English terms were not always translated using the JMDICT dictionary, although its coverage is large. This, too, may have impacted retrieval.

6 Conclusion

For our participation at the NTCIR worshop, we explored alternative query structures to group characters and character bigrams into longer concepts. Our results on the Japanese retrieval subtask show that some of these structures lead to good performance, similar to word-based retrieval. However, we also find that

RunID	Fields	Indexing	Query Structure	Relax		Rigid	
				Avg Prec.	# of Docs	Avg Prec.	# of Docs
TLRRD-E-J-D-01	D	bigrams	phr_both	0.0617	618	0.639	382
TLRRD-E-J-DC-02	D,C	words	Kanji + Hira.	0.0001	13	0.0000	9
TLRRD-E-J-DC-03	D,C	bigrams	sum_both	0.0957	1041	0.0922	705
Word, no Hira.	D,C	words	Kanji	0.0232	405	0.0222	405
TLRRD-E-J-DC-04	D,C	bigrams	phr_both	0.1043	890	0.1054	596

Table 4. Effect of resegmentation and indexing units on average precision and the number of relevant documents retrieved in the bilingual English/Japanese subtask. Runs labels with Kanji include Katakana as well. Bigram indexing include characters and character bigrams. The number of relevant documents is 2538 using relaxed judgments, and 1654 using rigid judgments.

the advantage of structured queries over flat queries is limited. Unlike previous work [2], we did not find that bigram indexing outperforms word indexing. Our Chinese runs did not support our assumption that combining characters and bigrams would improve retrieval. Instead of a boosting effect, we mostly observed an averaging effect. There are still too many unanswered issues with our bilingual runs to draw any conclusion.

References

[1] J. Breen. http://www.csse.monash.edu.au/ jwb/j_jmdict.html.

[2] A. Chen, F. C. Gey, and H. Jiang. Berkeley at ntcir-2: Chinese, japanese, and english ir experiments. In NTCIR2 [8].

[3] W. B. Croft, J. Callan, and J. Broglio. The inquery retrieval system. In *Proceedings of the 3rd International Conference on Database and Expert Systems Applications*, Spain, 1992.

[4] H. Fujii and W. B. Croft. A comparison of indexing techniques for japanese text retrieval. In *Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 237–246, Pittsburg, PA, 1993.

[5] K. L. Kwok. Comparing representations in chinese information retrieval. In *Proceedings of the 20th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 34–41, Philadelphia, PA, 1997.

[6] Y. Matsumoto, A. Kitauchi, T. Yamashita, Y. Hirano, H. Matsuda, K. Takaoka, and M. Asahara. *Morphological Analysis System ChaSen version 2.2.1 Manual*. Nara Institute of Science and Technology, December 2000.

[7] *Proceedings of the First NTCIR Workshop on Research in Chinese and Japanese Text Retrieval and Text Summarization*, Tokyo, Japan, 2000.

[8] *Proceedings of the Second NTCIR Workshop on Research in Chinese and Japanese Text Retrieval and Text Summarization*, Tokyo, Japan, 2001.

[9] D. W. Oard and J. Wang. Ntcir-2 ecir experiments at maryland: Comparing pirkola's structured queries and balanced translation. In NTCIR2 [8].

[10] A. Pirkola. The effects of query structure and dictionary setups in dictionary-based cross-language information retrieval. In *Proceedings of the 21th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 55–63, Melbourne, Australia, 1998.

[11] H. Turtle. Natural language vs. boolean query evaluation: a comparison of retrieval performance. In *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 212–220, Dublin, Ireland, 1994.