Description of NTU Approach to NTCIR3 Multilingual Information Retrieval

Wen-Cheng Lin and Hsin-Hsi Chen
Department of Computer Science and Information Engineering
National Taiwan University
Taipei, TAIWAN
E-mail: denislin@nlg.csie.ntu.edu.tw; hh_chen@csie.ntu.edu.tw

Abstract

This paper deals with Chinese, English and Japanese multilingual information retrieval. Several merging strategies, including raw-score merging, round-robin merging, normalized-score merging, and normalized-by-top-k merging, were investigated. Experimental results show that centralized approach is better than distributed approach. In distributed approach, normalized-by-top-k with consideration of translation penalty outperforms the other merging strategies.

Keywords: *Merging Strategy, Multilingual Information Retrieval, Query Translation*

1. Introduction

Multilingual Information Retrieval [7] uses a query in one language to retrieve documents in different languages. In addition to language translation issue, how to conduct a ranked list that contains documents in different languages from several text collections is also critical. There are two possible architectures in MLIR – say, centralized and distributed. In a centralized architecture, a huge collection that contains documents in different languages is used. In a distributed architecture, documents in different languages are indexed and retrieved separately, and all the results are merged into a multilingual ranked list. Several merging strategies have been proposed. Raw-score merging selects documents based on their original similarity scores. Normalized-score merging normalizes the similarity score of each document and sorts all the documents by their normalized scores. For each topic, the similarity score of each document is divided by the maximum score in this topic. Round-robin merging interleaves the results in the intermediate runs. In this paper, we adopted distributed architecture and proposed merging strategies to merge the result lists.

The rest of this paper is organized as follows. Section 2 describes the indexing method. Section 3 shows the query translation process. Section 4 describes our merging strategies. Section 5 shows the experiment results. Section 6 concludes the remark.

2. Indexing

The document set used in NTCIR3 MLIR task consists of Chinese, English and Japanese documents. The numbers of documents in Chinese, English, and Japanese document sets are 381,681, 22,927 and 236,664, respectively. The participants can use two or all of these three document collections as the target language sets. We used all of these three document collections to conduct X→ CJE experiments.

The IR model we used is the basic vector space model. Documents and queries are represented as term vectors, and cosine vector similarity formula is used to measure the similarity of a query and a document. Appropriate terms are extracted from each document in indexing stage. In the experiments, the <HEADLINE> and <TEXT> sections were used for indexing. For English, all words were retained, and all letters were transformed to lower case. The Japanese documents were first segmented by ChaSen [6]. All words in the above two sections were used as index terms. For Chinese, we used Chinese character bigrams to index Chinese documents. The term weighting function for all document sets is *tf*idf*.

3. Query Translation

In the experiment, the Japanese queries were used as source queries and translated into target languages, i.e., English and Chinese. We used CO model [1], which is a hybrid dictionary- and corpus-based method, to translate queries. Since we did not have a Japanese-Chinese dictionary, we used English as an intermediate language in the initial study. The Japanese queries were translated into English, and then the translated English queries were further translated into Chinese. The Japanese queries were translated into English in the way as follows:

(a) The Japanese query was segmented by ChaSen.

(b) For each Japanese query term, we found its English translation equivalents by looking up a Japanese-English dictionary.

(c) By using co-occurrence information trained from TREC6 text collection [4], we selected the best English translation equivalent for each source query term. We adopted mutual information (MI) [2] to measure the co-occurrence strength between words. For a query term, we compared the MI values of all the translation equivalent pairs (x, y), where x is the translation equivalent of this term, and y is the translation equivalent of another query term within a sentence. The word pair (x_i, y_j) with the highest MI value is extracted, and the translation equivalent x_i is regarded as the best translation equivalent of this query term. Selection is carried out based on the order of the query terms.

Translated English queries were translated into Chinese using the same method except that English queries did not need to be segmented. The MI values of Chinese words were trained from Academia Sinica Balanced Corpus (ASBC) [5].

4. Merging Strategies

There are two possible architectures in MLIR, i.e., centralized and distributed. In a centralized architecture, document collections in different languages are viewed as a single document collection and are indexed in one huge index file. The advantage of centralized architecture is that it avoids the merging problem. It needs only one retrieving phase to produce a result list that contains relevant documents in different languages. One of problems of a centralized architecture is that index terms may be over weighted. In other words, the total number of documents increases, but the number of occurrences of a term does not. In *tf*idf* scheme, the *idf* of a term is increased and it is over-weighted. This phenomenon is clear in small text collection. For example, the *N* in *idf* formula is 22,927 when English document is used. However, this value is increased to 641,272, i.e., about 27.97 times larger, if the three document collections are merged together. Comparatively, the weights of Chinese index terms are increased only 1.68 times due to the size of *N*. The increments of weights are unbalance for document collections in different size. Thus, IR system may prefer documents in small document collection.

The second architecture is a distributed MLIR. Documents in different languages are indexed and retrieved separately. The ranked lists of all monolingual and cross-lingual runs are merged into one multilingual ranked list. How to merge result lists is a problem. Recent works have proposed various approaches to deal with the merging problem. A simple merging method is raw-score merging, which sorts all results by their original similarity scores and then selects the top ranked documents. Raw-score merging is based on the assumption that the similarity scores across collections are comparable. However, the collection-dependent statistics in document or query weights invalidates this assumption [3, 8]. Another approach, round-robin merging, interleaves the results based on the rank. This approach assumes that each collection has approximately the same number of relevant documents and the distribution of relevant documents is similar across the result lists. Actually, different collections do not contain equal numbers of relevant documents. Thus, the performance of round-robin merging may be poor. The third approach is normalized-score merging. For each topic, the similarity score of each document is divided by the maximum score in this topic. After adjusting scores, all results are put into a pool and sorted by the normalized score. This approach maps the similarity scores of different result lists into the same range, from 0 to 1, and makes the scores more comparable. But it has a problem. If the maximum score is much higher than the second one in the same result list, the normalized-score of the document at rank 2 would be made lower even if its original score is high. Thus, the final rank of this document would be lower than that of the top ranked documents with very low but similar original scores in another result list.

Similarity score reflects the degree of similarity between a document and a query. A document with high similarity score seems to be more relevant to the desired query. But, if the query is not formulated well, e.g., inappropriate translation of a query, a document with high score still does not meet the user’s information need. When merging results, such documents that have incorrect high scores should not be included in the final result list. Thus, we have to consider the effectiveness of each individual run in the merging stage. The basic idea of our merging strategy is that adjusting the similarity scores of documents in each result list to make them more comparable and to reflect their confidence. The similarity scores are adjusted by the following formula.

$$\hat{S}_{ij} = S_{ij} \times \frac{1}{\overline{S}_k} \times W_i \qquad (1)$$

where S_{ij} is the original similarity score of the document at rank j in the ranked list of topic i,

$\hat{S}_{ij}$ is the adjusted similarity score of the document at rank j in the ranked list of topic i,

$\bar{S}_k$ is the average similarity score of top k documents, and

W_i is the weight of query i in a cross-lingual run.

We divide the weight adjusting process into two steps. First, we use a modified score normalization method to normalize the similarity scores. The original score of each document is divided by the average score of top *k* documents instead of the maximum score. We call this normalized-by-top-*k*. Second, the normalized score multiplies a weight that reflects the retrieval effectiveness of the desired topic in each text collection. Because of not knowing the retrieval performance in advance, we have to guess the performance of each run. For each language pair, the queries are translated into target language and then system retrieves the target language documents. A good translation should have better performance. We can predict the retrieval performance based on the translation performance. There are two factors affecting the translation performance, i.e., the degree of translation ambiguity and the number of unknown words. For each query, we compute the average number of translation equivalents of query terms and the number of unknown words in each language pair, and use them to compute the weights of each cross-lingual run. The weight can be determined by the following three formulas:

$$W_i = c_1 + \left[c_2 \times \left(\frac{51 - T_i}{50}\right)^2\right] + \left[c_3 \times \left(1 - \frac{U_i}{n_i}\right)\right] \qquad (2)$$

$$W_i = c_1 + \left[c_2 \times \left(\frac{1}{\sqrt{T_i}}\right)\right] + \left[c_3 \times \left(1 - \frac{U_i}{n_i}\right)\right] \qquad (3)$$

$$W_i = c_1 + \left[c_2 \times \left(\frac{1}{T_i}\right)\right] + \left[c_3 \times \left(1 - \frac{U_i}{n_i}\right)\right] \qquad (4)$$

where W_i is the weight of query i in a cross-lingual run,

T_i is the average number of translation equivalents of query terms in query i,

U_i is the number of unknown words in query i,

n_i is the number of query terms in query i, and

c_1, c_2 and c_3 are tunable parameters, and $c_1+c_2+c_3=1$.

In the experiment, the Japanese queries were translated into English, and then the translated English queries were further translated into Chinese. Some Japanese query terms have no English translation, and therefore they cannot be translated into English and also Chinese. The unknown words in Japanese-English translation are also unknown in English-Chinese translation. Thus, the number of unknown words in Japanese-English-Chinese translation is the sum of those in Japanese-English translation and English-Chinese translation.

5. Results

We submitted three J→CJE multilingual runs and one E→E monolingual run. All runs use description field only. The English monolingual run, NTU-E-E-D-01, uses official English topics to retrieve English documents. The three multilingual runs use Japanese topics as source queries. The Japanese topics were translated into English and Chinese by CO Model described in Section 3. The source Japanese topics and the translated English and Chinese topics were used to retrieve Japanese, English and Chinese documents, respectively. Then, we merged these three result lists. We used different merging strategies for the three multilingual runs.

1. NTU-J-CJE-D-01
   First, we used formula (1) to adjust the similarity score of each document. We used the average similarity score of top 10 documents for normalization. The weight W_i was determined by formula (2). The values of c_1, c_2 and c_3 were set to 0.1, 0.4 and 0.5, respectively. Then all results were put in a pool and sorted by the adjusted score. The top 1000 documents were selected as the final results.
2. NTU-J-CJE-D-02
   The merging strategy is the same as run NTU-J-CJE-D-01 except that the weight W_i was determined by formula (3).
3. NTU-J-CJE-D-03
   The similarity scores were adjusted by multiplying a constant weight. The similarity scores in Japanese-English run multiplied 1.5; the similarity scores in Japanese-Chinese run multiplied 0.5; the similarity scores in monolingual Japanese run were not changed. These values were trained from the experiments using dry-run data.

The results of our official runs are shown in Table 1. Table 2 shows the unofficial evaluation of intermediate monolingual (i.e., Japanese to Japanese) and cross-lingual runs (i.e., Japanese to English and Japanese to English). The relevant assessment of each language is extracted from multilingual assessment file. The performance of run NTU-J-CJE-D-02 is slightly better than that of run NTU-J-CJE-01. The weight W_i of run NTU-J-CJE-D-02 is smaller than that of run NTU-J-CJE-D-01 for most queries. But it seems not small enough for Japanese-Chinese cross-lingual run. Since the Chinese translations of Japanese queries are not translated well, the performance of Japanese-Chinese cross-lingual run is worse. When merging results, the

Table 1. The results of official runs

Run	# Topic	Scoring Mode	Average Precision	Recall
NTU-E-E-D-01	32	Rigid	0.2072	391 / 444
		Relax	0.2519	641 / 741
NTU-J-CJE-D-01	50	Rigid	0.0884	1211 / 4053
		Relax	0.0839	1769 / 6648
NTU-J-CJE-D-02	50	Rigid	0.0907	1172 / 4053
		Relax	0.0865	1719 / 6648
NTU-J-CJE-D-03	50	Rigid	0.0934	1194 / 4053
		Relax	0.0893	1766 / 6648

Table 2. The results of intermediate runs

Run	# Topic	Scoring Mode	Average Precision	Recall
ntu-fr-j-j-d	45	Rigid	0.1506	1064/ 1659
ntu-fr-j-e-d	40	Rigid	0.1269	225/456
ntu-fr-j-c-d	48	Rigid	0.0146	517/1938

Japanese-Chinese cross-lingual run should have lower weight. The performances of the official multilingual runs do not differ too much. The best run is NTU-J-CJE-D-03 whose average precision is 0.0934. The weights trained from dry-run experiments still perform well in the formal-run.

In order to compare the effectiveness of different merging strategies, we also conducted several unofficial runs shown as follows.

1. ntu-fr-j-cje-d-01
   The merging strategy is the same as run NTU-J-CJE-D-01, but the values of parameters c_1, c_2 and c_3 were set to 0, 0.6 and 0.4, respectively.
2. ntu-fr-j-cje-d-02
   The merging strategy is the same as run NTU-J-CJE-D-02, but the values of parameters c_1, c_2 and c_3 were set to 0, 0.6 and 0.4, respectively.
3. ntu-fr-j-cje-d-04
   The merging strategy is the same as run ntu-fr-j-cje-d-01 except that the weight W_i was determined by formula (4).
4. ntu-fr-j-cje-d-raw-score
   We used raw-score merging to merge result lists.
5. ntu-fr-j-cje-d-normalized-score
   The result lists were merged by normalized-score merging strategy. The maximum similarity score was used for normalization.
6. ntu-fr-j-cje-d-normalized-top10
   In this run, we used the modified normalized-score merging method. We did not consider the performance drop caused by query translation. That is, the weight W_i in formula (1) was 1 for every sub run.
7. ntu-fr-j-cje-d-round-robin
   We used round-robin merging to merge result lists.
8. ntu-fr-j-cje-d-centralized
   This run adopted centralized architecture. All document collections were indexed in one index file. The topics contained original Japanese query terms, translated English query terms and translated Chinese query terms.

The results of unofficial runs are shown in Table 3. We used the rigid relevant set to evaluate the unofficial runs. In the official evaluation, the Japanese documents without text were removed from the ranked list. We did not remove those Japanese documents when evaluating our unofficial runs in Table 1. Therefore, the results of unofficial runs cannot be compared to the official runs. We re-evaluated the official runs without removing the Japanese documents without text. The new results of runs NTU-J-CJE-D-01, NTU-J-CJE-D-02 and NTU-J-CJE-D-03 are shown in the last three rows in Table 3.

Table 3 shows that the performances of ntu-fr-j-cje-d-01, ntu-fr-j-cje-d-02, and ntu-fr-j-cje-d-04 are similar to those of official runs even the values of parameters c_1, c_2 and c_3 are changed. The performance of raw-score merging is good. This is probably because we use the same IR model and term weighting scheme for all text collections. Comparatively, the performances of normalized-score, round-robin and normalized-by-top-*k* merging are poor, especially round-robin merging strategy. Normalizd-by-top-*k* is better than normalized–score merging. When considering the translation penalty, the performance of normalized-by-top-*k* is increased and is better than the other merging strategies. It shows that translation penalty is helpful. The best run is ntu-fr-j-cje-d-centralized, which indexes all documents in different languages together. In this run, most of the top ranked documents are in Japanese or in English in most topics. Table 2 shows that the performances of Japanese monolingual

Table 3. The results of unofficial runs

Run	Average Precision	Recall
ntu-fr-j-cje-d-01	0.0833	1194 / 4053
ntu-fr-j-cje-d-02	0.0872	1152 / 4053
ntu-fr-j-cje-d-04	0.0868	1124 / 4053
ntu-fr-j-cje-d-raw-score	0.0867	1310 / 4053
ntu-fr-j-cje-d-normalized-score	0.0492	1245 / 4053
ntu-fr-j-cje-d-normalized-top10	0.0514	1257 / 4053
ntu-fr-j-cje-d-round-robin	0.0447	1233 / 4053
ntu-fr-j-cje-d-centralized	0.0973	1149 / 4053
NTU-J-CJE-D-01	0.0842	1211 / 4053
NTU-J-CJE-D-02	0.0863	1172 / 4053
NTU-J-CJE-D-03	0.0891	1194 / 4053

Table 4. The results of unofficial runs using new Japanese-Chinese run

Run	Average Precision	Recall
ntu-fr-j-c-d-2	0.0289	340/1938
ntu-fr-j-cje-d-01-2	0.0841	1242 / 4053
ntu-fr-j-cje-d-02-2	0.0869	1233 / 4053
ntu-fr-j-cje-d-04-2	0.0863	1229 / 4053
ntu-fr-j-cje-d-raw-score-2	0.0850	1315 / 4053
ntu-fr-j-cje-d-normalized-score-2	0.0685	1273 / 4053
ntu-fr-j-cje-d-normalized-top10-2	0.0635	1277 / 4053
ntu-fr-j-cje-d-round-robin-2	0.0516	1225/ 4053
ntu-fr-j-cje-d-centralized-2	0.0990	1177 / 4053

Table 5. Normalized-by-top-k with translation penalty (C_1=0, C_2=0.4, C_3=0.6)

Run	Average Precision	Recall
ntu-fr-j-cje-d-01-3	0.0877	1205 / 4053
ntu-fr-j-cje-d-02-3	0.0883	1203 / 4053
ntu-fr-j-cje-d-04-3	0.0880	1196 / 4053

retrieval and Japanese-English cross-lingual retrieval are much better than that of Japanese-Chinese cross-lingual retrieval. Therefore, the final result list should not contain too many Chinese documents. The over-weighting phenomenon in centralized architecture increases the scores of Japanese and English documents, so that more Japanese and English documents are included in the result list of run ntu-fr-j-cje-d-centralized. This makes the performance better.

In the initial experiments, we use English as a pivot language to derive Chinese translation equivalents of Japanese query terms. Experimental results show that the performance of Japanese-Chinese cross-lingual run is very bad. In the further tests, we translated Japanese queries into Chinese directly by using BitEx online Japanese-Chinese dictionary (http://www.bitex-cn.com/). Table 4 lists the performances of new Japanese-Chinese cross-lingual run and multilingual runs. The performance of the new Japanese-Chinese cross-lingual run is improved only a little, i.e., from 0.0146 to 0.0289. The major reason is that many query terms have no translation. After analysis, there are 351 distinct query terms in the description field of Japanese queries. Among these, total 232 terms have no translation. The remaining query terms have only one translation. Furthermore, our system did not return any documents in the new Japanese-Chinese cross-lingual run for topics 16, 19 and 23. This is because all the translated Chinese terms of these three topics are unigrams and our system uses bigrams as index terms. In such a case, no relevant document is proposed.

The performance of centralized architecture is still the best. When considering the translation penalty, normalized-by-top-*k* (i.e., runs ntu-fr-j-cje-d-02-2 and ntu-fr-j-cje-d-04-2) is better than the other merging strategies. Compared to the old translation scheme, the performances of all merging strategies except raw-score and normalized-by-top-*k* with translation penalty are increased when the Japanese queries are translated into Chinese by using BitEx online dictionary. Since the Japanese query terms have only one Chinese translation, emphasizing the degree of translation ambiguity part in formulas 2-4 increases the merging weight of Japanese-Chinese cross-lingual run. That decreases the performance of normalized-by-top-*k* with translation penalty. Thus, we adjusted the values of parameters c_1, c_2 and c_3 to 0, 0.4 and 0.6, respectively. Table 5 shows that the performances are improved. In summary, the translation penalty is an important factor in merging, and we should also consider the quality factor of dictionaries (e.g., its coverage).

6. Concluding Remarks

This paper considers the two architectures in MLIR. The centralized approach performed well in all the experiments. However, the centralized architecture is not suitable in practice, especially for very huge corpora. Centralized architecture needs spending more time to index and to retrieve documents in all languages. Distributed architecture is more flexible. It is easy to add or delete corpora in different languages and employ different retrieval systems in distributed architecture.

Merging problem is critical in distributed architecture. This paper proposed several merging strategies to integrate the result lists of collections in different languages. The experimental results showed that the performance of normalized-by-top-*k* with translation penalty was better than raw-score merging, normalized-score merging and round-robin merging.

References

[1] Chen, H.H., Bian, G.W., and Lin, W.C., 1999. Resolving translation ambiguity and target polysemy in cross-language information retrieval. In *Proceedings of 37th Annual Meeting of the Association for Computational Linguistics*, Maryland, June 1999. Association for Computational Linguistics, 215-222.

[2] Church, K., Gale, W., Hanks, P., and Hindle, D., 1989. Parsing, Word Associations and Typical Predicate-Argument Relations. In *Proceedings of International Workshop on Parsing Technologies*, Pittsburgh, PA, August 1989. Carnegie Mellon University, Pittsburgh, PA, 389-398.

[3] Dumais, S.T., 1993. LSI meets TREC: A Status Report. In *proceedings of the First Text REtrieval Conference (TREC-1)*, Gaithersburg, Maryland, November, 1992. NIST Publication, 137-152.

[4] Harman, D.K., 1997. TREC-6 Proceedings. Gaithersburg, Maryland. National Institute of Standards and Technology.

[5] Huang, C.R. and Chen, K.J., 1995. Academic Sinica Balanced Corpus. Technical Report 95-02/98-04. Academic Sinica, Taipei, Taiwan.

[6] Matsumoto, Y., Kitauchi, A., Yamashita, T., and Hirano, Y., 1999. Japanese Morphological Analysis System ChaSen version 2.0 Manual. Technical Report NAIST-IS-TR99009, Nara Institute of Science and Technology Technical Report.

[7] Oard, D.W. and Dorr, B.J., 1996. A Survey of Multilingual Text Retrieval. Technical Report UMIACS-TR-96-19, University of Maryland, Institute for Advanced Computer Studies.

[8] Voorhees, E.M., Gupta, N.K., and Johnson-Laird, B., 1995. The Collection Fusion Problem. In *proceedings of the Third Text REtrieval Conference (TREC-3)*, Gaithersburg, Maryland, November, 1994. NIST Publication, 95-104.