Utilization of structured queries and web resources in transitive CLIR

Mayya Sharipova
Graduate School of Science and Engineering
Ritsumeikan University

Akira Maeda
College of Information Science and Engineering
Ritsumeikan University

Abstract. Some people argue that CLIR (Cross Language Information Retrieval) has reached its maturity and there is little left for further exploration. However we can see that most web search engines do not utilize CLIR yet, which signifies that the quality of CLIR is far from desirable. This paper introduces an idea of using a new type of structured queries and Web resources for increase in effectiveness and efficiency of CLIR. It was tested on Russian-English-Japanese CLIR and proved to be effective with precision of English-Japanese part best in NTCIR-3, and precision of Russian-English-Japanese CLIR among top in transitive CLIR. Increased effectiveness and efficiency will contribute to the implementation of CLIR in real search engines.

1 Introduction

Transitive cross-language information retrieval is a type of CLIR, where a query is in a source language and document collection is in another target language, and to translate a query from the source language to the target language a third language, known as a pivot language, is used. In CLIR three main problems should be addressed:

- *Problem of translation.* Besides terms that can be easily found in bilingual dictionaries, queries may contain proper nouns, neologisms, or technical terms that are not contained in the standard dictionaries. This so called "Out-of-vocabulary" problem should be resolved through different methods.
- *Problem of disambiguation*. A query term in one language when translated to the document language may have several translations. The choice of the most appropriate translation for a particular situation is called the problem of disambiguation. The acuteness of the disambiguation problem doubles in the transitive CLIR, where at least three different languages are involved.
- *Problem of retrieval.* Being the ultimate goal of any IR system the solution of the retrieval problem in CLIR should take into account special features of cross-language retrieval.

In our paper we propose the following solutions to the stated above problems:

- *Problem of translation.* We will show how this problem can be handled using existing methods in the context of these three languages Russian-English-Japanese, and also introduce a new efficient method for dictionary look-up that can be used for all languages. We will also explain how different Web resources can be utilized to translate query terms that are not in the dictionary (Section 3.3).
- *Problem of disambiguation and retrieval*. In our paper we propose a single combined method for the solution of the disambiguation and retrieval problems. This method is an innovative type of structured queries, specially designed for CLIR, and with the help of our constructed Russian-English-Japanese CLIR system we will demonstrate that it leads to high precision and quick retrieval (Section 2.2 and 2.3).

Moreover, in our paper we will explain how to build Russian-English-Japanese CLIR system and special features of Russian and Japanese language that should be taken into account (Section 3.1 and 3.2).

2 Structured queries

2.1 What has already been done

You have a query $Q_1\ Q_{2...}Q_n$ in one language, where $Q_1\ Q_{2...}Q_n$ are terms of the query. How to understand which document d in a different language is relevant to the query? The common method is to translate every term into the language of the document and obtain for each query term Q_i the set of translations in the document language - $T(Q_i)$. Then, for every document calculate term frequency and document frequency using formulas (1) and (2), where D_k is a document term[1]. Finally, to get the score of the document combine TF and DF into some formula in such a way that the higher TF the greater the score and the higher DF the lower the score. For example, in Apache Lucene IR system[1] the score of a document is calculated according to formula (3).

$$TF_j(Q_i) = \sum_{\{k|Dk \in T(Qi)\}} TF_j(D_k) \quad \textbf{(1)}$$

$$DF\ (Q_i) = \left| \bigcup_{\{k|Dk \in T(Qi)\}} \{d \mid D_k \in d\} \right| \quad \textbf{(2)}$$

$$score(q,d) = coord(q,d) \cdot queryNorm(q) \cdot \sum_{t \in q} (tf(t\ in\ d) \cdot idf(t)^2 \cdot t.getBoost() \cdot norm(t,d)) \quad \textbf{(3)}$$

1. http://lucene.apache.org/

2.2 Our idea

To our mind, retrieval should be founded on the following principles:

- **discrimination of key-terms and other terms** (documents containing none of key-terms are not retrieved). *Key-terms* are terms that have only one translation in the document language (such as proper nouns or very concrete notions), while other terms have multiple translations. Let us take a query "Dioxin Human body Effect Threat" (a query from NTCIR-3 Mainichi document collection). In this query "Dioxin" is a key-term with only one translation in the document language, while "Human body", "Effect" and "Threat" terms have several translations (Table 1).

If a document does not contain any of key-terms in the document language would it be relevant? The answer is "No", because a user explicitly asked about "Dioxin", not about any other chemicals. But the above mentioned formula (3) allow a document not containing "Dioxin" be retrieved in response to this query, if the document has high occurrences of other terms.

Thus, the first thing we should do in the retrieval step is to extract documents containing any of key-terms in the document language and then score them.

- **give priority to diversity rather then to frequency, that is emphasizing diversity in scoring.** Let us compare two documents – one has a lot of occurrences of "Dioxin" and "Effect" and "Threat", but does not mention at all "Human Body", while the other document has occurrences of all terms, but not in a great number. So using above formulas and summing up the contribution for all terms, the fist document may get higher score, while is it obvious that the second document is more relevant. In Lucene IR system there is a *coord* parameter (formula 3) (a number of different query terms in the document divided by the whole number of terms in the query. For a document containing "Dioxin","Effect", "Threat" 3 terms among 4 terms query , this parameter is 3/4), which is multiplied by the sum of the terms scores to give the final score for every document. But this parameter cannot always guarantee that a document containing more terms from the query will have a higher score than documents containing some terms from the query.

These were two principles that, we think, the retrieval in CLIR should be based on. The next section explains how to embed them into the retrieval process.

2.3 How to implement our idea

We propose a new way of scoring documents in CLIR. Suppose, we want to find Japanese documents corresponding to the English query "Dioxin Human body Effect Threat". First we translate the English query into Japanese language. Table 1 shows all possible translations for query terms, acquired through a dictionary and Web resources.

Table 1. Query terms with translations.

Dioxin	Human body	Effect	Threat
ダイオキシン	人体 人身	エフェクト,効 ,効き目 ,効験,効能 ,効目 ,効用 ,効力 ,甲斐 ,手答え ,趣 ,趣き,出来映え ,出来栄え,出来具合 ,利き目, 効果 ,影響 ,作用	威迫 ,恐嚇 ,恐喝 ,脅威 ,脅嚇,脅喝,脅迫 ,威嚇 ,脅かし,脅し ,劫 ,剽 ,恫喝

Since a query term "Dioxin" has only one translation, its Japanese translation will be a *key-term*, while translations of other terms will constitute *synonym groups*. The scoring and retrieval is conducted in the following sequence:

1. **Extraction of all documents containing at least one of the key-terms**. Then we will work and assign scores only to these documents, thus discarding from the beginning documents not containing key-terms.
2. **Score calculation for every document in the extracted set**. A score is calculated according to formula (4), where *diversity_score* (the number of different groups which terms a document contains), multiplied by α, and then summed up with the term-frequency value. α is an arbitrary parameter for prioritizing diversity. For our experiments we have chosen α=1000. Table 2 demonstrates some examples of score calculations.

$$score(q,d)=\alpha \cdot diversity\_score + \sum_{t \in q} tf(t \text{ in } d) \quad \textbf{(4)}$$

3. **Sorting documents in descending order** by the score field.

This scoring method will allow to prioritize diversity (how many different terms from the query a document contains) and discriminate key-terms, which to our mind is an effective way to retrieve documents.

3 Translation

3.1 Russian-English translation and disambiguation

The web site "Sdictionary community" freely provides dictionary bases for many languages, including Russian. "Russian-English Full Dictionary"[2] from this web site

2. http://sdict.com/en/view.php?file=rus_eng_full2.dct

Table 2. Score calculation of the documents.

Doc. Num.	Number of terms				Diversity score	Term freq.	Final score
	Dioxin- ダイオキシン	Human body 人体, 人身	Effect エフェクト,効 ,効き目 , 効験,効能 ,効目 ,効用 , 効力 ,甲斐 ,手答え ,趣 , 趣き,出来映え ,出来栄え,出来具合 ,利き目, 効果 ,影響 ,作用	Threat 威迫 ,恐嚇 ,恐喝 ,脅威 ,脅嚇, 脅喝,脅迫 ,威嚇 ,脅かし,脅し ,劫,剽 ,恫喝			
50	1				1	1	1*1000+1=1001
1245	1				1	1	1*1000+1=1001
1503	7	1	1		3	9	3*1000+9=3009
1506	2		2		2	4	2*1000+4=2004
1662	7	1	1		3	9	3*1000+9=3009
2306	1		1		2	2	2*1000+2=2002
…							
3224	1	1	1	1	4	4	4*1000+4=4004
….							

was used as a dictionary-base for translating queries from Russian to English. The richly inflected nature of Russian language, when a single word may have more than 50 different forms, complicates its computer processing. A form of the word used in a query may not be found in the dictionary which usually has only the base form of the word. Therefore translation resources from Russian cannot be limited only to a bilingual list; other resources for getting a base form of the word should also be employed. List of Russian words in the base form and other possible forms - stems.zip can be freely obtained from the web site of the multitran dictionary[3]. The web site of the multitran dictionary also contains the list of the most frequent Russian words. All this can allow creating a powerful Russian Analyzer. The whole list of translation resources from English to Russian is following:

- *Bilingual Russian-English list*
- *List of Russian words in base form and other forms*
- *Russian Wikipedia* - for the translation of terms not found in the dictionary
- *Transliteration table*- for translation of terms not found even in Wikipedia. We constructed a transliteration table-correspondence between Russian letters / combination of letters and English letters / combination of letters. The transliteration table was created based on the observation how foreign-origin words, typically from English, translated into Russian (Table 3). The first column of the table contains Russian letters or combination of letters, the second column how they are normally represented in English. Some Russian letters may have several English equivalents; in this case several English words may be originated — several possible translations of a Russian word.

Table 3. Example from Russian- English transliteration table.

Russian	English
ция	tion
сия	sion
….	…..
а	a
б	b
в	v;w
…	…

In translation phase we obtained a number of English translation candidates for our original Russian query. Since we did not have a document collection in English, we had to conduct disambiguation using Web resources. We submitted all English candidates to Exalead[4] search engine with a proximity search option, and chose the translation candidate with the biggest number of retrieved documents. This English translation of the Russian query was then translated into Japanese to retrieve Japanese documents.

3.2 English-Japanese translation

For English-Japanese translation we have chosen EDICT Dictionary File and English Wikipedia. For indexing and retrieving documents we applied Apache Lucene toolkit. Within Apache Lucene project a Japanese analyzer is provided, which was also utilized in our system. Disambiguation and retrieval steps were

3. http://www.multitran.ru/

4. http://www.exalead.com/search

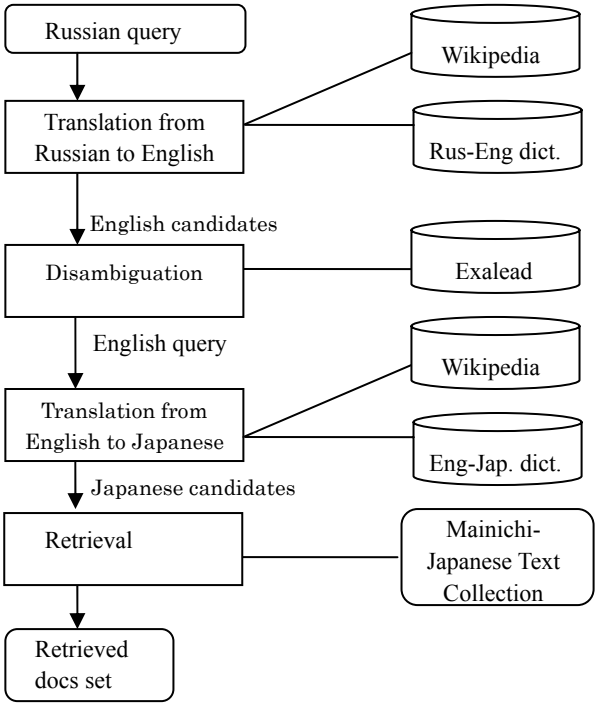

Fig.1 Outline of the system.

combined in our system, and solved through our proposed structured queries (Section 2.2).
Fig.1 summarizes the flow of the system.

3.3 Dictionary lookup

The loading into memory extensive dictionaries with thousands of words using hash-table may lead to the out-of-memory problem. Indexing a dictionary via Apache Lucene or some other IR system and using the index of this dictionary will not only solve out-of-memory problem, but also provide additional features to look up the dictionary through fuzzy queries or wild-card queries, in case when exact translation could not be found. Furthermore, the index file can preserve additional fields, besides words and their translations, such as a part of speech, which also can be helpful in CLIR.

4 Experiments

For experiments of our system we used NTCIR-3 Mainichi-98,99 newspaper collection in Japanese. English queries were provided by this collection, and they were translated into Russian by one of the authors, whose native language is Russian. Among 55 queries for CLIR, 16 contained proper nouns that are not included in standard dictionaries.
We retrieved Japanese documents according to the procedure, described in the Section 2.3.
We conducted two runs: transitive Russian-English-Japanese CLIR and English-Japanese part of it, both on title field. Table 4 sums up the results of the experiments. It should be said that, since our transitive CLIR depends on the results of Exalead search engine, the MAP (Mean average precision) of Russian-English-Japanese CLIR is not stable; the average values of the obtained results are demonstrated in Table 4.

Table 4. MAP of the experiments, title field.

	Eng.-Jap.	**Rus.-Eng.-Jap.**
Rigid	0.1975	0.1556
Relax	0.2813	0.1904

In the retrieval of documents we used the following set of operations:

1. Extraction of all documents containing at least one key-term. In case there are no key-terms, containing at least one term from the query (OR operation).
2. Calculation of the score of the every document from the extracted document set (Sum and multiplication operations).
3. Sorting of the documents (Sorting operation).

Score calculation is computed utilizing only sum and multiplication operations. If the extracted document set from the step one is not big, the whole procedure is instantaneously fast. If it is big, the retrieval is still faster most of other methods, employing complex logarithmic expressions, exponential equations, disambiguation methods, etc.

5 Conclusion

In our paper we proposed an alternative way CLIR can be implemented, based on the principles of the diversity priority and discrimination of query terms, where discrimination information is learned through translation. This is a very simple way without complex calculations, but it was proved to be effective.
At the beginning of the paper we said that we still cannot see CLIR part in the most web search engines. This partially can be attributed to the complexity of the most CLIR researches. Under these circumstances, our proposed retrieval method being simple and effective can be an asset in the realization of CLIR in the web search engines.

References

[1] Oard, D. W. and F. Ertunc. Translation-Based Indexing for Cross-Language Retrieval, Proc. ECIR2002, pp. 324-333, 2002.

[2] Kuang-hua Chen, Hsin-Hsi Chen, Noriko Kando, Kazuko Kuriyama, Sukhoon Lee, Sung Hyon Myaeng, Kazuaki Kishida, Koji Eguchi, and Hyeon Kim. Overview of CLIR Task at the Third NTCIR Workshop, Proc. NTCIR-3, 2003.