Cross-Language Information Retrieval by Domain Restriction using Web Directory Structure

Fuminori Kimura
Faculty of Culture and Information Science,
Doshisha University
1–3 Miyakodani Tatara,
Kyoutanabe-shi, Kyoto, Japan
jt-bnk04@mail.doshisha.ac.jp

Akira Maeda
Department of Media Technology,
College of Information Science and Engineering,
Ritsumeikan University
1-1-1 Noji-Higashi, Kusatsu, Shiga, Japan
amaeda@media.ritsumei.ac.jp

Kenji Hatano
Faculty of Culture and Information Science,
Doshisha University
1–3 Miyakodani Tatara,
Kyoutanabe-shi, Kyoto, Japan
khatano@mail.doshisha.ac.jp

Jun Miyazaki
Graduate School of Information Science,
Nara Institute of Science and Technology
8916–5 Takayama, Ikoma, Nara, Japan
miyazaki@is.naist.jp

Shunsuke Uemura
Faculty of Informatics,
Nara Snagyo University
3–12–1 Tatsuno-kita, Sango-cho, Ikoma-gun, Nara, Japan
uemurashunsuke@nara-su.ac.jp

Abstract

In this paper, we propose a cross-language information retrieval (CLIR) method based on estimating for domains of the query using hierarchic structures of Web directories. To get the most appropriate translation of the queries, we utilize the Web directories written in many different languages as multilingual corpus for disambiguating translation of the query and for estimating the domain of search results using hierarchic structures of Web directories. From experimental evaluations, we found that there is an advantage in retrieval accuracy using our proposal for disambiguating translation in CLIR system. We found that it is effective to restrict to target fields of the query using lower level merged categories in order to acquire suited translation of the query.

1 Introduction

With the worldwide popularity of the Internet, more and more languages are being used for Web documents, and it is now much easier to access documents written in foreign languages. However, existing Web search engines only support the retrieval of documents that are written in the same language as the query, so there is no efficient way for monolingual users to retrieve documents written in non-native languages. There might also be cases, depending on the user's needs, where valuable information is written in a language other than the user's native language. To satisfy these needs in a typical monolingual retrieval system, users have to manually translate queries themselves using a dictionary, etc. This method is not only difficult for the user, it might also result in the query being translated incorrectly, especially when the user is unfamiliar with the language.

To meet these needs, there has been intensive research in recent years on Cross-Language Information Retrieval (CLIR), a technique for retrieving documents written in one language using a query written in another language. A variety of methods, including the use of corpus statistics to translate terms and the disambiguation of translated terms, have been investigated and some useful results have been obtained. However, corpus-based disambiguation methods are significantly affected by the domain of the training corpus, so they may be much less effective for retrieval in other

1530-1605/08 $25.00 © 2008 IEEE

domains. In addition, since the Web consists of documents in various domains or genres, methods used for CLIR of Web documents should be independent of specific domains.

2 Related Work

Approaches to CLIR can be classified into three types: document translation, query translation, and the use of interlingual representation. The approach based on translation of target documents has the advantage of using existing machine translation systems, in which more content information is available for disambiguation. In general, this is a more effective retrieval method than those based on query translation [8]. However, since it is impractical to translate a huge document collection beforehand, and it is difficult to extend this method to new languages, this approach is unsuitable for multilingual, large-scale, frequently updated document collections on the Web. The second approach [9] transfers both documents and queries into an interlingual representation, such as bilingual thesaurus classes or language-independent vector spaces. This last approach requires a training phase using a bilingual (parallel or comparable) corpus as training data. Third approach uses semantic class of thesaurus[3] or Latent Semantic Index[1]. This approach needs not to recognize difference between languages. However, this approach requires huge computing cost of learning when the size of using corpus is large. Thus, third approach also unsuitable for document collections on the Web.

The major problem in using an approach based on the translation and disambiguation of queries is that queries submitted by ordinary users of Web search engines tend to be very short. They consist of approximately two words on average [5], and are usually just an enumeration of keywords (i.e. there is no context). However, one advantage of this approach is that the translated queries can simply be fed into existing monolingual search engines. In this approach, a source language query is first translated into the target language using a bilingual dictionary, and the translated query is then disambiguated. Our method falls into this category.

We should point out that corpus-based disambiguation methods are significantly affected by differences between the domain of the query and the corpus. Hull suggests that these differences may adversely affect the retrieval efficiency of methods that use parallel or comparable corpora [4]. Lin et al. conducted comparative experiments between three monolingual corpora that had different domains and sizes, and concluded that a large-scale, domain-consistent corpus is needed to obtain useful co-occurrence data [7].

In relation to Web retrieval, which is the target of our research, the system has to cope with queries on many different topics. However, it is impractical to prepare corpora that cover every possible domain. In our previous paper [6], we proposed a CLIR method that uses documents in Web directories that have several language versions (such as Yahoo!), instead of using existing corpora, to improve retrieval effectiveness.

3 Cross-Language Information Retrieval Using Web Directories

Figure 1 illustrates the outline of the proposed system. This system consists of query and target language versions of Web Directory, each language versions of feature term database, bilingual dictionary, and retrieval target document set. The part surrounded by a dotted line illustrates components of translation processing for query.

The processing on our system can be divided into two phases. One is the preprocessing phase, which extracts feature terms from each category of a Web directory, and stores it in the feature term database in advance. Another is the retrieval phase, which translates the given query into the target language, and retrieves documents.

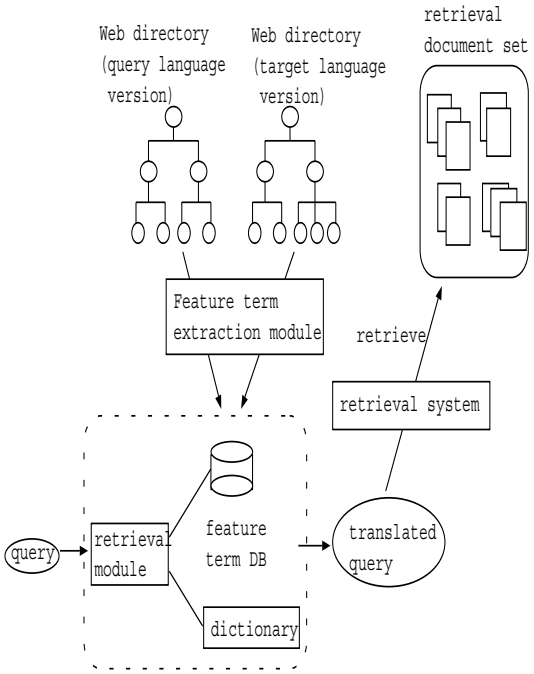

Figure 1. Outline of the proposed system.

3.1 Preprocessing Phase

Figure 2 shows the processing flow of the proposed system. In the preprocessing phase, the system conducts feature term extraction and category matching between the query and target languages in advance. The following procedure is used for this phase:

1. Feature-term extraction
   For each category in all language versions of a Web directory,
   (a) extract terms from Web documents in the required category and calculate the weight of the terms.
   (b) extract the top n ranked terms as the feature terms of the category.
   (c) store the feature terms in the feature-term database.
2. Category matching between languages
   For each category in one language version, estimate the corresponding category in the other language version.

As an example, we explain the process of category matching for a category in the query language version in Figure 2. First, the system extracts terms from Web documents in category a and calculates their weights in that category (1)(a). Secondly, the system extracts the top n ranked terms as feature terms of the category and obtains a set of feature terms f_a (1)(b). Thirdly, the system stores the set of feature terms f_a in the feature-term database (1)(c). Lastly, the system searches for the category that is most similar to the feature-term set f_a from the target language version, and the category a is marked as the corresponding category of the matched category (2). Note that the category matching method is not the focus of this paper. An arbitrary method can be used for category matching. For example, we could calculate the similarity between categories based on extracted feature terms, or we could manually match each category. The category pairs acquired by this process are used in retrieval.

3.1.1 Feature Term Extraction

The features of each category are represented in the feature-term set. The feature-term set is a set of terms that are judged to represent the features of the category. The feature-term set for each category is extracted as follows: (1) the system extracts terms from Web documents that belong to a given category; (2) the system calculates the weights of the extracted terms; and (3) the top n ranked terms are extracted as the feature terms of the category.

Weights of feature terms are calculated by TF·ICF (term frequency · inverse category frequency). TF·ICF is a variation of TF·IDF (term frequency · inverse document frequency). TF·IDF is calculated by multiplying the term frequency by the inverse document frequency. Instead of using a document as the unit, TF·ICF calculates weights by category. TF·ICF is able to calculate term weights considering the content of the category. It is calculated as follows:

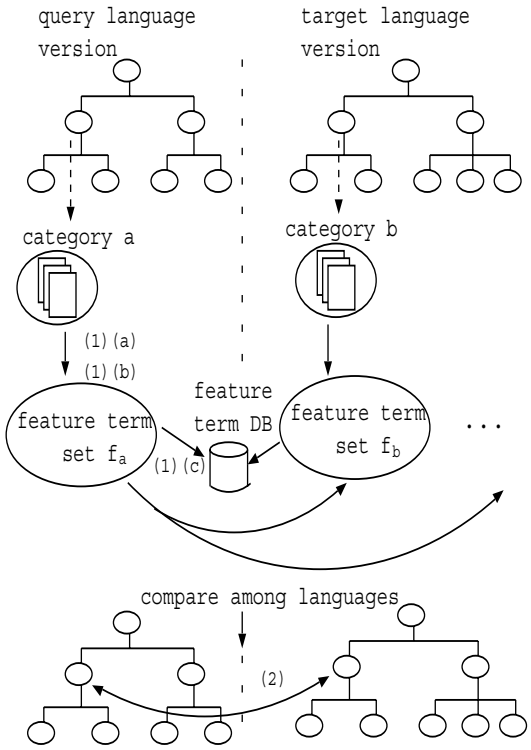

Figure 2. Flow of preprocessing.

$$tf \cdot icf(t_i, c) = \frac{f(t_i)}{N_c} \cdot \log \frac{N}{n_i} + 1$$

where t_i is the term appearing in category c, $f(t_i)$ is the term frequency of term t_i, N_c is the total number of terms in category c, n_i is the number of categories that contain the term t_iCand N is the total number of categories in the directory.

3.1.2 Retrieval Phase

Figure 3 illustrates the processing flow for retrieval. First, the system estimates the relevant category of the query from the query language version. Secondly, the system selects a category corresponding to the relevant category. Thirdly, the system translates the query terms into the target language using the feature-term set for the corresponding category. Finally, the system retrieves documents using the translated query. The procedure for the retrieval phase is as follows:

(1) For each category in the query language version, calculate the relevance between the query and the feature-term set for the category.

(2) Determine the category with the highest relevance as the relevant category for the query.

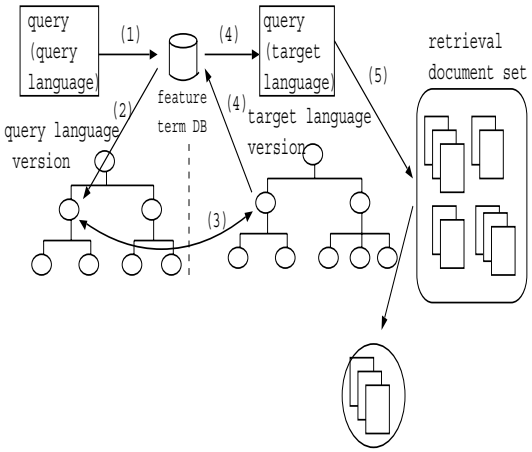

Figure 3. Flow of retrieval.

(3) Select the category corresponding to the most relevant category from the target language version.

(4) Translate the query terms into the target language using the feature-term set of the corresponding category.

(5) Retrieve documents using the translated query.

3.1.3 Selection of Relevant Category

In our system, queries consist of keywords, not sentences. We define the query vector $\vec{q}$ as follows:

$$\vec{q} = (q_1, q_2, \ldots, q_n)$$

where q_k is the weight of the k-th keyword in the query. We define the values of all q_k as 1.

First, the system calculates the relevance between the query and each category in the query language version, and determines the most relevant category to the query in the query language version. The relevance between the query and each category is calculated by multiplying the inner product between the query terms and the feature-term set of the target category by the angle of these two vectors.The relevance between query q and category c is calculated as follows:

$$rel(q, c) = \vec{q}\vec{c}\frac{\vec{q} \cdot \vec{c}}{|\vec{q}||\vec{c}|}$$

where $\vec{c}$ is a vector of category c and is defined as follows:

$$\vec{c} = (w_1, w_2, \ldots, w_n)$$

where w_k is the weight of the k-th keyword in the feature term set of c.

However, we applied the square of the cosine factor of the ordinary cosine distance to the relevance. The reason why we used this measure is to emphasize the relevance for the queries which have more than one term.

When there is more than one category whose relevance to the query exceeds a certain threshold, all are selected as relevant categories for the query.

3.1.4 Query Translation

Figure 4 illustrates the processing flow for query translation. First, for each query term q, the system looks up the term in a bilingual dictionary and extracts all translation candidates for the feature term. Next, the system checks whether each translation candidate is included in the feature-term set of the corresponding category. If it is, the system checks the weight of the candidate in the feature-term set. Lastly, the highest-weighted translation candidate in the feature-term set of the corresponding category is selected as the translation of the feature term.

If there is no translation candidate for a feature term in the feature-term set of the corresponding category, that term is ignored in the retrieval. However, in some cases, the source language term itself is useful as a feature term in the target language. For example, some English terms (mostly abbreviations) are commonly used in documents written in other languages (e.g. “WWW”, “HTML”). Therefore, when there is no translation candidate for a feature term in the feature-term set of the corresponding category, the feature term itself is checked to see whether it is included in the feature-term set of the corresponding category. If it is, the feature term itself is treated as the translated term.

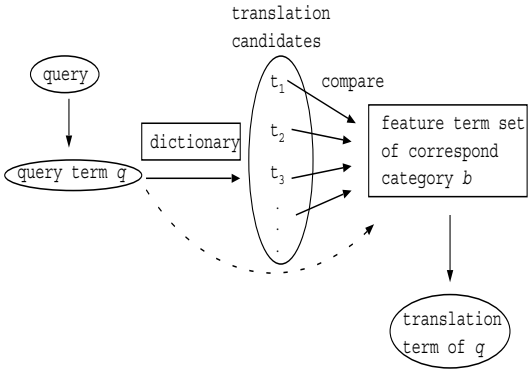

Figure 4. Translation of a query.

As an example, we consider that the English term “system” is translated into Japanese when the relevant category for the query is the category “Computers and Internet”. The English term “system” has the following translation candidates in a dictionary: “uchu

(universe/space)”C“houhou (method)”C“soshiki (organization)”C“kikan (organ)”C“sisutemu (system)”, etc. We check each of these translation candidates in the feature-term set of the category “Computers and Internet.” Then the highest-weighted term for these translation candidates in the category “Computers and Internet” is determined as the translation of the English term “system” in this category. If no translation candidate are included in the feature-term set for the category “Computers and Internet,” the English term “system” is itself treated as the translation.

In the next section, we propose some revisions of the query translation method described above.

3.1.5 Retrieval of Documents

The system retrieves documents using queries translated by the method described in Section 3.1.4 The documents to be retrieved need not be those registered in the Web directory. Instead, the system may use an existing retrieval system.

3.2 Method of Category Merging

Each category in Web directory is useful to specify the fields of the query. However, some categories have insufficient web documents. The system cannot acquire sufficient statistical information to resolve translation disambiguation. This problem might be caused by the following reasons; one possible reason is that there are some categories which are too close in topic, and it might cause poor accuracy. Another possible reason is that some categories have insufficient amount of text in order to obtain statistically significant values for feature term extraction.

Considering the above observations, we might expect that the accuracy will be improved by merging child categories at some level in the category hierarchy in order to merge some categories similar in topic and to increase the amount of text in a category. Figure 5 illustrates the result of category merging. Each category existing some level in the category hierarchy includes all sub categories under the category.

4 Experiments

We conducted experiments on the proposed method using English and Japanese versions of Yahoo! category. In these experiments, we used Japanese queries and retrieved English documents. The purpose of the experiments was to investigate what level of category merging for Web directory is most effective to improve the precision of CLIR. We conducted experiments in the three cases, used the category of Web directory is merged into top level from the top of Web directory(hereafter called the “1-lv” for short) or second level of it(hereafter called the “2-lv” for short) or third level of it(hereafter called the “3-lv” for short). We also conducted experiments on the case of no disambiguation of translations for comparison (hereafter called the “baseline” for short). In the baseline, we used all the translation candidates in a dictionary as query terms, except for multi-word terms. Processing after the translation of a query was done using our proposed method.

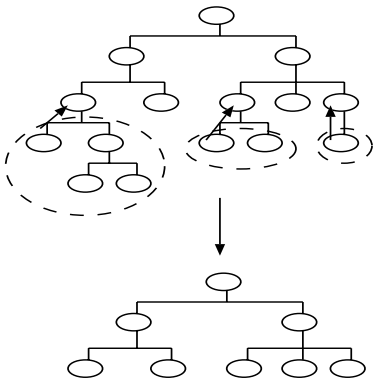

Figure 5. Category Merging.

4.1 Method of Experiments

In these experiments, we used document sets and queries presented in the CLIR task at The 3rd NTCIR Workshop[1] (hereafter called the ”NTCIR3 test collection” for short). We used two document sets from the NTCIR3 test collection: EIRB010, which consists of several English newspapers published in Taiwan from 1998 to 1999, and Mainichi Daily 1998-1999, which consists of English newspaper articles published in Japan from 1998 to 1999. The Japanese query set for the NTCIR3 test collection, which consists of 50 queries, was used in these experiments.

To resolve ambiguities, we used English and Japanese versions of Yahoo! category as the Web directory. We excluded all sub-categories in the category “Regional” in each version. We eliminated this category because it is unsuitable for translation since it consists of documents written about regions all over the world. In these experiments, we merged ’child’ categories into the ’parent’ categories at a certain level using the directory hierarchy. Finally, each category in top three levels takes in all of its sub categories. In the case of 1-lv, the categories were finally merged into 13 categories that were linked from the top of the Yahoo! category for each language version. We merged the categories because some categories did not have enough Web documents to provide sufficient statistical information. Table 1 shows the number of categories in each level. In English, 1-lv has 13 categories, 2-lv has 397 categories and 3-lv has 4066

[1] http://research.nii.ac.jp/ntcir/index-en.html

Table 1. The number of categories in each level.

		1-lv	2-lv	3-lv
English	all	13	397	4066
	eliminated	13	255	644
Japanese	all	13	391	2953
	eliminated	13	154	153

categories. In Japanese, 1-lv has 13 categories, 2-lv has 391 categories and 3-lv has 2953 categories. We merged categories in order to resolve shortage of statistical information. However, some of the merged categories cannot acquire sufficient statistical information. We eliminate such categories that have less than 10,000 feature terms. Table 1 also shows the number of categories in each level after eliminating categories that have less than 10,000 feature terms. In English, 1-lv has 13 categories, 2-lv has 255 categories and 3-lv has 644 categories. In Japanese, 1-lv has 13 categories, 2-lv has 154 categories and 3-lv has 153 categories.

In extracting terms from English Web documents, the terms were transformed into the original form, and stop words were eliminated. We used the stop word list published by Frakes and Baeza-Yates [2] and we used the Japanese morphological analyzer, "Chasen"[2]. In these experiments, to extract terms from Japanese Web documents, sentences were separated by Chasen, and the system extracted nouns, verbs, adjectives, and unknown terms.

For translation, we used the "EDR Electronic Dictionary: Jpn.-Eng. Bilingual Dictionary"[3] (hereafter called the "EDR dictionary" for short). The average number of translation candidates for translating the Japanese queries in the NTCIR3 test collection was 5.17.

In addition, category matching between languages was done manually. This was because the correspondence of the 13 child categories directly linked from the top page of each version of Yahoo! was obvious.

We used the query that were extracted from the "TITLE" fields of the Japanese query set in the NTCIR3 test collection. We used these fields, which contain comparatively fewer terms, because ordinary users generally use about two terms for a single query [5]. Each query was subjected to morphological analysis by Chasen, and we used nouns, verbs, adjectives, and unknown terms as query terms.

4.2 Result of Experiments

Table 2 shows retrieval effectiveness of these experiments. This result shows 11 point average precision about each query in the case of 1-lv, 2-lv, 3-lv and baseline. In the case of 1-lv, the system used 13 categories linked from the top page of Yahoo! category. In the case of 2-lv, the system used child categories of 1-lv. In the case of 3-lv, the system used child categories of 2-lv. In each case, each category in top three levels contains all sub categories.

Table 2. 11 point average precision about each query.

query number	1-lv	2-lv	3-lv	baseline
2	0.0971	0.0870	**0.1042**	0.0270
5	0.0027	0.0027	**0.0029**	0.0022
9	0.0084	0.0086	**0.0193**	0.0146
12	0.0001	0.0001	**0.0009**	**0.0009**
13	**0.0222**	**0.0222**	0.0102	0.0067
14	0.0059	0.0059	**0.0075**	0.0059
18	0.1321	0.0961	**0.1523**	0.0951
19	**0.0056**	0.0053	0.0052	0.0016
20	0.1627	0.2048	**0.2390**	0.1313
21	0.0245	0.0245	**0.0281**	0.0189
23	0.1962	**0.2059**	**0.2059**	0.0000
24	0.0003	0.0005	**0.0008**	0.0002
26	**0.0031**	0.0004	0.0005	0.0018
27	0.1640	**0.2200**	0.1640	0.0160
28	0.0098	**0.0126**	0.0005	0.0001
29	0.1596	0.1596	**0.2332**	**0.2332**
31	0.0015	0.0015	0.0015	0.0015
32	0.0087	0.0087	**0.0165**	0.0090
33	**0.0158**	**0.0158**	**0.0158**	0.0084
34	**0.0040**	0.0036	0.0016	0.0036
35	0.0052	**0.0057**	0.0055	0.0044
36	**0.1078**	**0.1078**	**0.1078**	0.0050
37	0.0086	0.0228	**0.0355**	0.0084
38	0.0057	0.0057	**0.0072**	0.0057
39	0.0121	**0.0141**	0.0040	0.0035
42	**0.0016**	**0.0016**	0.0008	0.0005
43	**0.0012**	**0.0012**	0.0005	0.0001
45	0.0000	0.0000	0.0000	0.0000
46	0.0179	**0.0198**	0.0000	0.0024
50	0.0151	**0.0212**	0.0151	0.0137
average	0.0400	0.0429	**0.0462**	0.0203

Table 3 shows results of T-test between proposed method and baseline. We tested if there are significant difference between each three levels and baseline. We assumed no difference between each three levels and baseline, and tested by two-tailed paired T-test. Table 4 shows results of T-test between 1-lv and 2-lv or 2-lv and 3-lv. The condition of testing is same as Table 3.

Besides, table 5 shows translation list about each query that has difference in average precision among three levels.

[2] http://chasen-legacy.sourceforge.jp/
[3] http://www2.nict.go.jp/r/r312/EDR/J_index.html

Table 3. Probability value of T-test between Proposed Method and Baseline.

merged level	1-lv	2-lv	3-lv
probability	0.0447	0.0434	0.0113

Table 4. Probability value of T-test among Proposed Method.

merged level	1, 2-lv	2, 3-lv	3, 1-lv
probability	0.2987	0.4103	0.1069

4.3 Discussion

In average of all queries, the average precision of all our proposed method exceed the average precision of baseline. This result verified that our proposed method is effective for Cross-Language Information Retrieval. Table 3 shows probabilities all of three levels are below 0.05. This means that assumption of non-difference between each three levels and baseline is rejected, and there are significant difference. This result also verified effectiveness of our proposed method.

In 2-lv, there are 16 queries that changed its average precision comparing with 1-lv queries as table 2 indicates. 11 queries improved, 5 queries got worse. In these queries, some increase in the number of its translations, others decrease. The query no. 50 is one of increasing case. In this query, translation of the Japanese term "fasshon (fashion)" increase two translations (fashion → fashion, fashinable closes, vogue). In increasing case, queries tend to acquire derivations and synonyms. On the other hand, the query no. 20 is one of decreasing case. In this query, translation of the Japanese term "teikei (cooperation)" decrease one translation (joint business, cooperation → cooperation). This tendency indicates that restricting to target fields of the query is effective to acquire suited translation of the query.

In 3-lv, there are 16 queries that changed its average precision comparing with 2-lv queries as table 2 indicates. 14 queries improved, 11 queries got worse. In the query no. 2, 2-lv has six translations for the Japanese term "kanyu (adherence)", which are "joining", "subscription", "affiliation", "entry", "admission" and "joint business cooperation". On the other hand, 3-lv has only two translations ("subscription", "adherence"). These two translations are proper terms in diplomacy field. This result shows that restricting to narrower fields is more effective to acquire suited translation of the query.

However, excessive restriction also has risk of causing a bad influence. In the query no. 50, translations of 3-lv are removed two terms from the translations of 2-lv.

Table 5. Translation list about each query.

query number	lv	translations
2	1-lv	WTO subscription affiliation entry admission joint business cooperation
	2-lv	WTO joining subscription affiliation entry admission joint business cooperation
	3-lv	WTO subscription entry adherence
20	1-lv	Nissan Renault funds capital fund investment money joint business cooperation
	2-lv	Nissan Renault capital fund investment money cooperation
	2-lv	Nissan Renault capital fund investment money cooperation
50	1-lv	fashion mode style
	2-lv	fashionable clothes vogue fashion mode style
	3-lv	fashion mode style

These terms improved average precision comparing with 1-lv. This result indicates that if the specified fields of the query are too narrow, there is a possibility of omitting important translations from the field.

In conclusion from above discussion, restricting the target fields of the query is effective to acquire suited translation of the query. However, excessive restriction causes decline in retrieval effectiveness. Besides, Table 4 shows that there are no significance of difference between each levels. Thus, It is needed to find appropriate level of the merged category in Web directory in order to resolve translation disambiguity.

5 Conclusion

In this paper, we proposed a query disambiguation method for Cross-Language Information Retrieval using Web directories. In addition, we conducted experiments of retrieval using NTCIR3 test collection and verified that the proposed method is effective for Cross-Language Information Retrieval. We found that it is effective to restrict to target fields of the query using lower level merged categories in order to acquire suited translation of the query. However, excessive restriction has possibility of causing decline in retrieval effectiveness.

The proposed method is independent of a particular domain because it uses documents in a Web directory as the corpus. Our method is particularly effective in cases where the document collection covers a wide range of domains

such as the Web. The method does not require expensive linguistic resources except for a dictionary. Therefore, it could easily be extended to other languages as long as there is a Web directory in those languages and a suitable dictionary is available.

In future work, we need to detect most suited level of using merged categories in order to acquire more proper translations of query term. Besides, we consider to use Yahoo! category as linguistic resource. There is possibility of improving to retrieval precision. However, lower category has difficulty of category matching among different language category. Lower category also has a problem that it has insufficient Web documents. Thus, we have to consider using suitable linguistic resource for Yahoo! category (e.g. Wikipedia).

Acknowledgements

This work was partly supported by MEXT Grant-in-Aid for Scientific Research on Priority Areas #19024058.

References

[1] S. Deerwester, S. T. Dumais, T. K. Furnas, G. W.and Landauer, and R. A. Harshman. Indexing by latent semantic analysis. *journal of the American Society for Information Science*, 46(1):391–407, 1990.

[2] W. Frakes and R. Baeza-Yates. *Information Retrieval: Data Structures and Algorithms, chapter 7*. Prentice-Hall, 1992.

[3] J. Gonzalo, F. Verdejo, C. Peters, and N. Calzolari. Applying EuroWordNet to cross-language text retrieval. *Computers and the Humanities*, 32:185–207, 1998.

[4] D. A. Hull. Using structured queries for disambiguation in cross-language information retrieval. *Electronic Working Notes of the AAAI Symposium on Cross-Language Text and Speech Retrieval*, 1997.

[5] B. J. Jansen, A. Spink, and T. Saracevic. Real life, real user queries on the Web. *Information Processing & Management*, 36(2):207–227, 2000.

[6] F. Kimura, A. Maeda, M. Yoshikawa, and S. Uemura. Cross-Language Information Retrieval using Web Directories. *Proceedings of IEEE Pacific Rim Conference on Communications, Computers and Signal Processing (PACRIM '03)*, pages 911–914, 2003.

[7] C.-J. Lin, W.-C. Lin, G.-W. Bian, and H.-H. Chen. Description of the NTU Japanese-English cross-lingual information retrieval system used for NTCIR workshop. *First NTCIR Workshop on Research in Japanese Text Retrieval and Term Recognition*, pages 145–148, 1999.

[8] T. Sakai. MT-based Japanese-English cross-language IR experiments using the TREC test collections. *Proceedings of The Fifth International Workshop on Information Retrieval with Asian Languages (IRAL2000)*, 2000.

[9] H.-C. Seo, S.-B. Kim, H.-C. Rim, and S.-H. Myaeng. Improving Query Translation in English-Korean Cross-Language Information Retrieval. *Information Processing and Management*, 41(3):507–522, May 2005.