The Use of Monolingual Context Vectors for Missing Translations in Cross-Language Information Retrieval

Yan Qu[1], Gregory Grefenstette[2], and David A. Evans[1]

[1] Clairvoyance Corporation, 5001 Baum Boulevard, Suite 700,
Pittsburgh, PA, 15213, USA
{yqu, dae}@clairvoyancecorp.com
[2] LIC2M/SCRI/LIST/DTSI/CEA, B.P.6,
92265 Fontenay-aux-Roses Cedex, France
{Gregory.Grefenstette}@cea.fr

Abstract. For cross-language text retrieval systems that rely on bilingual dictionaries for bridging the language gap between the source query language and the target document language, good bilingual dictionary coverage is imperative. For terms with missing translations, most systems employ some approaches for expanding the existing translation dictionaries. In this paper, instead of lexicon expansion, we explore whether using the context of the unknown terms can help mitigate the loss of meaning due to missing translation. Our approaches consist of two steps: (1) to identify terms that are closely associated with the unknown source language terms as *context* vectors and (2) to use the translations of the associated terms in the context vectors as the surrogate translations of the unknown terms. We describe a query-independent version and a query-dependent version using such monolingual context vectors. These methods are evaluated in Japanese-to-English retrieval using the NTCIR-3 topics and data sets. Empirical results show that both methods improved CLIR performance for short and medium-length queries and that the query-dependent context vectors performed better than the query-independent versions.

1 Introduction

For cross-language text retrieval systems that rely on bilingual dictionaries for bridging the language gap between the source query language and the target document language, good bilingual dictionary coverage is imperative [8,9]. Yet, translations for proper names and special terminology are often missing in available dictionaries. Various methods have been proposed for finding translations of names and terminology through transliteration [5,11,13,14,16,18,20] and corpus mining [6,7,12,15,22]. In this paper, instead of attempting to find the candidate translations of terms without translations to expand existing translation dictionaries, we explore to what extent simply using text context can help mitigate the missing translation problem and for what kinds of queries. The context-oriented approaches include (1) identifying words that are closely associated with the unknown source language terms as *context* vectors and (2) using the translations of the associated words in the context vectors as the surrogate translations of the unknown words. We describe a query-independent

R. Dale et al. (Eds.): IJCNLP 2005, LNAI 3651, pp. 22 – 33, 2005.

version and a query-dependent version using such context vectors. We evaluate these methods in Japanese-to-English retrieval using the NTCIR-3 topics and data sets. In particular, we explore the following questions:

- Can translations obtained from context vectors help CLIR performance?
- Are query-dependent context vectors more effective than query-independent context vectors for CLIR?

In the balance of this paper, we first describe related work in Section 2. The methods of obtaining translations through context vectors are presented in Section 3. The CLIR evaluation system and evaluation results are presented in Section 4 and Section 5, respectively. We summarize the paper in Section 6.

2 Related Work

In dictionary-based CLIR applications, approaches for dealing with terms with missing translations can be classified into three major categories. The first is a do-nothing approach by simply ignoring the terms with missing translations. The second category includes attempts to generate candidate translations for a subset of unknown terms, such as names and technical terminology, through phonetic translation between different languages (i.e., transliteration) [5,11,13,14,16,18,20]. Such methods generally yield translation pairs with reasonably good accuracy reaching about 70% [18]. Empirical results have shown that the expanded lexicons can significantly improve CLIR system performance [5,16,20]. The third category includes approaches for expanding existing bilingual dictionaries by exploring multilingual or bilingual corpora. For example, the "mix-lingual" feature of the Web has been exploited for locating translation pairs by searching for the presence of both Chinese and English text in a text window [22]. In work focused on constructing bilingual dictionaries for machine translation, automatic translation lexicons are compiled using either clean aligned parallel corpora [12,15] or non-parallel comparable corpora [6,7]. In work with non-parallel corpora, contexts of source language terms and target language terms and a seed translation lexicon are combined to measure the association between the source language terms and potential translation candidates in the target language. The techniques with non-parallel corpora save the expense of constructing large-scale parallel corpora with the tradeoff of lower accuracy, e.g., about 30% accuracy for the top-one candidate [6,7]. To our knowledge, the usefulness of such lexicons in CLIR systems has not been evaluated.

While missing translations have been addressed in dictionary-based CLIR systems, most of the approaches mentioned above attempt to resolve the problem through dictionary expansion. In this paper, we explore non-lexical approaches and their effectiveness on mitigating the problem of missing translations. Without additional lexicon expansion, and keeping the unknown terms in the source language query, we extract context vectors for these unknown terms and obtain their translations as the surrogate translations for the original query terms. This is motivated by the pre-translation feedback techniques proposed by several previous studies [1,2]. Pre-translation feedback has been shown to be effective for resolving translation ambiguity, but its effect on recovering the lost meaning due to missing translations has not been empirically evaluated. Our work provides the first empirical results for such an evaluation.

3 Translation via Context Vectors

3.1 Query-Independent Context Vectors

For a source language term *t*, we define the context vector of term *t* as:

$$C_t = \langle t_1, t_2, t_3, t_4, ..., t_i \rangle$$

where terms t_1 to t_i are source language terms that are associated with term *t* within a certain text window in some source language corpus. In this report, the associated terms are terms that co-occur with term *t* above a pre-determined cutoff threshold.

Target language translations of term *t* are derived from the translation of the known source language terms in the above context vectors:

$$trans(t) = <trans(t_1), trans(t_2), \ldots, trans(t_n)>$$

Selection of the source language context terms for the unknown term above is only based on the association statistics in an independent source language corpus. It does not consider other terms in the query as context; thus, it is query *independent*. Using the Japanese-to-English pair as an example, the steps are as follows:

1. For a Japanese term *t* that is unknown to the bilingual dictionary, extract concordances of term *t* within a window of *P* bytes (we used *P=200* bytes or 100 Japanese characters) in a Japanese reference corpus.
2. Segment the extracted Japanese concordances into terms, removing stopwords.
3. Select the top *N* (e.g., N=5) most frequent terms from the concordances to form the context vector for the unknown term *t*.
4. Translate these selected concordance terms in the context vector into English to form the pseudo-translations of the unknown term *t*.

Note that, in the translation step (Step 4) of the above procedure, the source language association statistics for selecting the top context terms and frequencies of their translations are not used for ranking or filtering any translations. Rather, we rely on the Cross Language Information Retrieval system's disambiguation function to select the best translations in context of the target language documents [19].

3.2 Query-Dependent Context Vectors

When query context is considered for constructing context vectors and pseudo-translations, the concordances containing the unknown terms are re-ranked based on the similarity scores between the window concordances and the vector of the known terms in the query. Each window around the unknown term is treated as a document, and the known query terms are used. This is based on the assumption that the top ranked concordances are likely to be more similar to the query; subsequently, the context terms in the context vectors provide better context for the unknown term. Again, using the Japanese-English pair as an example, the steps are as follows:

1. For a Japanese term *t* unknown to the bilingual dictionary, extract a window of text of P bytes (we used *P=200* bytes or 100 Japanese characters) around every occurrence of term *t* in a Japanese reference corpus.
2. Segment the Japanese text in each window into terms and remove stopwords.
3. Re-rank the window based on similarity scores between the terms found in the window and the vector of the known query terms.
4. Obtain the top *N* (e.g., N=5) most frequently occurring terms from the top *M* (e.g., M=100) ranking windows to form the Japanese context vector for the unknown term *t*.
5. Translate each term in the Japanese context vector into English to form the pseudo-translations of the unknown term *t*.

The similarity scores are based on Dot Product.

The main difference between the two versions of context vectors is whether the other known terms in the query are used for ranking the window concordances. Presumably, the other query terms provide a context-sensitive interpretation of the unknown terms. When M is extremely large, however, the query-dependent version should approach the performance of the query-independent version.

We illustrate both versions of the context vectors with topic 23 (金大中大統領の対アジア政策 "President Kim Dae-Jung's policy toward Asia") from NTCIR-3:

First, the topic is segmented into terms, with the stop words removed:

金大中; 大統領; アジア; 政策

Then, the terms are categorized as "known" vs. "unknown" based on the bilingual dictionary:

Unknown:
Query23: 金大中

Known:

Query23:大統領
Query23:アジア
Query23:政策

Next, concordance windows containing the unknown term 金大中 are extracted:

経済危機克服へ８項目－－韓国の金大中・次期大統領、雇用促進など提示
【ソウル３１日大澤文護】韓国の金大中（キムデジュン）次期大統領はく
【ソウル３１日大澤文護】韓国の金大中（キムデジュン）次期大統領は
経世済民」の書を記者団に見せる金大中・次期大統領＝ＡＰ
……

Next, the text in each window is segmented by a morphological processor into terms with stopwords removed [21].

In the query-independent version, we simply select the top 5 most frequently occurring terms in the concordance windows. The top 5 source language context terms for 金大中 are:

3527:金
3399:大中
3035:大統領
2658:韓国
901:キムデジュン[1]

Then, the translations of the above context terms are obtained from the bilingual dictionary to provide pseudo-translations for the unknown term 金大中, with the relevant translations in italics:

金大中 ≅ 金 ⇒ gold
金大中 ≅ 金 ⇒ metal
金大中 ≅ 金 ⇒ money
金大中 ≅大中 ⇒ ∅
金大中 ≅ 大統領 ⇒ chief executive
金大中 ≅ 大統領 ⇒ president
金大中 ≅ 大統領 ⇒ presidential
金大中 ≅ 韓国 ⇒ korea
金大中 ≅キムデジュン ⇒ ∅

With the query-dependent version, the segmented concordances are ranked by comparing the similarity between the concordance vector and the known term vector. Then we take the 100 top ranking concordances and, from this smaller set, select the top 5 most frequently occurring terms. This time, the top 5 context terms are:

1391:大統領
1382:金
1335:大中
1045:韓国
379:キムデジュン

In this example, the context vectors from both versions are the same, even though the terms are ranked in different orders. The pseudo-translations from the context vectors are:

金大中 ≅ 大統領 ⇒ chief executive
金大中 ≅ 大統領 ⇒ president
金大中 ≅ 大統領 ⇒ presidential
金大中 ≅ 金 ⇒ gold
金大中 ≅ 金 ⇒ metal
金大中 ≅ 金 ⇒ money
金大中 ≅大中 ⇒ ∅
金大中 ≅ 韓国 ⇒ korea
金大中 ≅キムデジュン ⇒ ∅

[1] Romanization of the katakana name キムデジュン could produce a correct transliteration of the name in English, which is not addressed in this paper. Our methods for name transliteration can be found in [18,20].

4 CLIR System

We evaluate the usefulness of the above two methods for obtaining missing translations in our Japanese-to-English retrieval system. Each query term missing from our bilingual dictionary is provided with pseudo-translations using one of the methods. The CLIR system involves the following steps:

First, a Japanese query is parsed into terms[2] with a statistical part of speech tagger and NLP module [21]. Stopwords are removed from query terms. Then query terms are split into a list of known terms, i.e., those that have translations from bilingual dictionaries, and a list of unknown terms, i.e., those that do not have translations from bilingual dictionaries. Without using context vectors for unknown terms, translations of the known terms are looked up in the bilingual dictionaries and our disambiguation module selects the best translation for each term based on coherence measures between translations [19].

The dictionaries we used for Japanese to English translation are based on edict[3], which we expanded by adding translations of missing English terms from a core English lexicon by looking them up using BabelFish[4]. Our final dictionary has a total of 210,433 entries. The English corpus used for disambiguating translations is about 703 MB of English text from NTCIR-4 CLIR track[5]. For our source language corpus, we used the Japanese text from NTCIR-3.

When context vectors are used to provide translations for terms missing from our dictionary, first, the context vectors for the unknown terms are constructed as described above. Then the same bilingual lexicon is used for translating the context vectors to create a set of pseudo-translations for the unknown term *t*. We keep all the pseudo-translations as surrogate translations of the unknown terms, just as if they really were the translations we found for the unknown terms in our bilingual dictionary.

We use a corpus-based translation disambiguation method for selecting the best English translations for a Japanese query word. We compute coherence scores of translated sequences created by obtaining all possible combinations of the translations in a source sequence of *n* query words (e.g., overlapping 3-term windows in our experiments). The coherence score is based on the mutual information score for each pair of translations in the sequence. Then we take the sum of the mutual information scores of all translation pairs as the score of the sequence. Translations with the highest coherence scores are selected as best translations. More details on translation disambiguation can be found in [19].

Once the best translations are selected, indexing and retrieval of documents in the target language is based on CLARIT [4]. For this work, we use the dot product function for computing similarities between a query and a document:

---

[2] In these experiments, we do not include multiple-word expression such as 戦争犯罪 (*war crime*) as terms, because translation of most compositional multiple-word expressions can be generally constructed from translations of component words (戦争 and 犯罪) and our empirical evaluation has not shown significant advantages of a separate model of phrase translation.

[3] http://www.csse.monash.edu.au/~jwb/j_edict.html

[4] http://world.altavista.com/

[5] http://research.nii.ac.jp/ntcir/ntcir-ws4/clir/index.html

$$sim(P, D) = \sum_{t \in P \cap D} W_P(t) \bullet W_D(t) \ . \tag{1}$$

where $W_P(t)$ is the weight associated with the query term t and $W_D(t)$ is the weight associated with the term t in the document D. The two weights are computed as follows:

$$W_D(t) = TF_D(t) \bullet IDF(t) \ . \tag{2}$$

$$W_P(t) = C(t) \bullet TF_P(t) \bullet IDF(t) \ . \tag{3}$$

where IDF and TF are standard inverse document frequency and term frequency statistics, respectively. $IDF(t)$ is computed with the target corpus for retrieval. The coefficient $C(t)$ is an "importance coefficient", which can be modified either manually by the user or automatically by the system (e.g., updated during feedback).

For query expansion through (pseudo-) relevance feedback, we use pseudo-relevance feedback based on high-scoring sub-documents to augment the queries. That is, after retrieving some sub-documents for a given topic from the target corpus, we take a set of top ranked sub-documents, regarding them as relevant sub-documents to the query, and extract terms from these sub-documents. We use a modified Rocchio formula for extracting and ranking terms for expansion:

$$Rocchio(t) = IDF(t) \times \frac{\sum_{D \in DocSet} TF_D(t)}{NumDoc} \tag{4}$$

where $IDF(t)$ is the Inverse Document Frequency of term t in reference database, $NumDoc$ the number of sub-documents in the given set of sub-documents, and $TF_D(t)$ the term frequency score for term t in sub-document D.

Once terms for expansion are extracted and ranked, they are combined with the original terms in the query to form an expanded query.

$$Q_{new} = k \times Q + Q_{\exp} \tag{5}$$

in which Q_{new}, Q_{orig}, Q_{exp} stand for the new expanded query, the original query, and terms extracted for expansion, respectively. In the experiments reported in Section 5, we assign a constant weight to all expansion terms (e.g., 0.5)

5 Experiments

5.1 Experiment Setup

For evaluation, we used NTCIR-3 Japanese topics[6]. Of the 32 topics that have relevance judgments, our system identifies unknown terms as terms not present in our expanded Japanese-to-English dictionary described above. The evaluation of the

[6] http://research.nii.ac.jp/ntcir/workshop/OnlineProceedings3/index.html

effect of using context vectors is based only on the limited number of topics that contain these unknown terms. The target corpus is the NTCIR-3 English corpus, which contains 22,927 documents. The statistics about the unknown terms for short (i.e., the title field only), medium (i.e., the description field only), and long (i.e., the description and the narrative fields) queries are summarized below. The total number of unknown terms that we treated with context vectors was 83 (i.e., 6+15+62).

	Short	Medium	Long
No. of topics containing unknown terms	5[7]	14[8]	24[9]
Avg No. of terms in topics (total)	3.2 (16)	5.4 (75)	36.2 (86.9)
Avg. No. of unknown terms (total)	1 (6)	1.1 (15)	2.6[10] (62)

For evaluation, we used the mean average precision and recall for the top 1000 documents and also precision@30, as defined in TREC retrieval evaluations.

We compare three types of runs, both with and without post-translation pseudo-relevance feedback.

- Runs without context vectors (baselines)
- Runs with query-dependent context vectors
- Runs with query-independent context vectors

5.2 Empirical Observations

Tables 1-4 present the performance statistics for the above runs. For the runs with translation disambiguation (Tables 1-2), using context vectors improved overall recall, average precision, and precision at 30 documents for **short** queries. Context vectors moderately improved recall, average precision (except for the query independent version), and precision at 30 documents for **medium** length queries.

For the long queries, we do not observe any advantages of using either query-dependent or query-independent versions of the context vectors. This is probably because the other known terms in long queries provide adequate context for recovering the loss of missing translation of the unknown terms. Adding candidate translations from context vectors only makes the query more ambiguous and inexact.

When all translations were kept (Tables 3-4), i.e., when no translation disambiguation was performed, we only see overall improvement in recall for short and medium-length queries. We do not see any advantage of using context vectors for improving average precision or precision at 30 documents. For longer queries, the performance statistics were overall worse than the baseline. As pointed out in [10], when all translations are kept without proper weighting of the translations, some terms get more favorable treatment than other terms simply because they contain more translations. So, in models where all translations are kept, proper weighting schemes should be developed, e.g., as suggested in related research [17].

[7] Topics 4, 23, 26, 27, 33.
[8] Topics 4, 5, 7, 13, 14, 20, 23, 26, 27, 28, 29, 31, 33, 38.
[9] Topics 2, 4, 5, 7, 9, 13, 14, 18, 19, 20, 21, 23, 24, 26, 27, 28, 29, 31, 33, 37, 38, 42, 43, 50.
[10] The average number of unique unknown terms is 1.4.

Table 1. Performance statistics for short, medium, and long queries. Translations were disambiguated; no feedback was used. Percentages show change over the baseline runs.

No Feedback	Recall	Avg. Precision	Prec@30
Short			
Baseline	28/112	0.1181	0.05
With context vectors (query independent)	43/112 (+53.6%)	0.1295 (+9.7%)	0.0667 (+33.4%)
With context vectors (query dependent)	43/112 (+53.6%)	0.1573 (+33.2%)	0.0667 (+33.4)
Medium			
Baseline	113/248	0.1753	0.1231
With context vectors (query independent)	114/248 (+0.9%)	0.1588 (-9.5%)	0.1256 (+2.0%)
With context vectors (query dependent)	115/248 (+1.8%)	0.1838 (+4.8%)	0.1282 (+4.1%)
Long			
Baseline	305/598	0.1901	0.1264
With context vectors (query independent)	308/598 (+1.0%)	0.1964 (+3.3%)	0.1125 (-11.0%)
With context vectors (query dependent)	298/598 (-2.3%)	0.1883 (-0.9%)	0.1139 (-9.9%)

Table 2. Performance statistics for short, medium, and long queries. Translations were disambiguated; for pseudo-relevance feedback, the top 30 terms from top 20 subdocuments were selected based on the Rocchio formula. Percentages show change over the baseline runs.

With Feedback	Recall	Avg. Precision	Prec@30
Short			
Baseline	15/112	0.1863	0.0417
With context vectors (query independent)	40/112 (+166.7%)	0.1812 (-2.7%)	0.0417 (+0.0%)
With context vectors (query dependent)	40/112 (+166.7%)	0.1942 (+4.2%)	0.0417 (+0.0%)
Medium			
Baseline	139/248	0.286	0.1513
With context vectors (query independent)	137 (-1.4%)	0.2942 (+2.9%)	0.1538 (+1.7%)
With context vectors (query dependent)	141 (+1.4%)	0.3173 (+10.9%)	0.159 (+5.1%)
Long			
Baseline	341/598	0.2575	0.1681
With context vectors (query independent)	347/598 (+1.8%)	0.2598 (+0.9%)	0.1681 (+0.0%)
With context vectors (query dependent)	340/598 (-0.3%)	0.2567 (-0.3%)	0.1639 (-2.5%)

Table 3. Performance statistics for short, medium, and long queries. All translations were kept for retrieval; pseudo-relevance feedback was not used. Percentages show change over the baseline runs.

No Feedback	Recall	Avg. Precision	Prec@30
Short			
Baseline	33/112	0.1032	0.0417
With context vectors (query independent)	57/112 (+72.7%)	0.0465 (-54.9%)	0.05 (+19.9%)
With context vectors (query dependent)	41/112 (+24.2%)	0.1045 (-0.2%)	0.0417 (+0%)
Medium			
Baseline	113/248	0.1838	0.0846
With context vectors (query independent)	136/248 (+20.4%)	0.1616 (-12.1%)	0.0769 (-9.1%)
With context vectors (query dependent)	122/248 (+8.0%)	0.2013 (+9.5%)	0.0769 (-9.1%)
Long			
Baseline	283	0.1779	0.0944
With context vectors (query independent)	295/598 (+4.2%)	0.163 (-8.4%)	0.0917 (-2.9%)
With context vectors (query dependent)	278/598 (-1.8%)	0.1566 (-12.0%)	0.0931 (-1.4%)

Table 4. Performance statistics for short, medium, and long queries. All translations were kept for retrieval; for pseudo-relevance feedback, the top 30 terms from top 20 subdocuments were selected base on the Rocchio formula. Percentages show change over the baseline runs.

With Feedback	Recall	Avg. Precision	Prec@30
Short			
Baseline	40/112	0.1733	0.0417
With context vectors (query independent)	69/112 (+72.5%)	0.1662 (-4.1%)	0.1583 (+279.6%)
With context vectors (query dependent)	44/112 (+10.0%)	0.1726 (-0.4%)	0.0417 (+0.0%)
Medium			
Baseline	135/248	0.2344	0.1256
With context vectors (query independent)	161/248 (+19.3%)	0.2332 (-0.5%)	0.1333 (+6.1%)
With context vectors (query dependent)	139/248 (+3.0%)	0.2637 (+12.5%)	0.1154 (-8.1%)
Long			
Baseline	344/598	0.2469	0.1444
With context vectors (query independent)	348/598 (+1.2%)	0.2336 (-5.4%)	0.1333 (-7.7%)
With context vectors (query dependent)	319/598 (-7.3%)	0.2033 (-17.7%)	0.1167 (-19.2%)

6 Summary and Future Work

We have used context vectors to obtain surrogate translations for terms that appear in queries but that are absent from bilingual dictionaries. We have described two types of context vectors: a query-independent version and a query-dependent version. In the empirical evaluation, we have examined the interaction between the use of context vectors with other factors such as translation disambiguation, pseudo-relevance feedback, and query lengths. The empirical findings suggest that using query-dependent context vectors together with post-translation pseudo-relevance feedback and translation disambiguation can help to overcome the meaning loss due to missing translations for short queries. For longer queries, the longer context in the query seems to make the use of context vectors unnecessary.

The paper presents only our first set on experiments of using context to recover meaning loss due to missing translations. In our future work, we will verify the observations with other topic sets and database sources; verify the observations with other language pairs, e.g., Chinese-to-English retrieval; and experiment with different parameter settings such as context window size, methods for context term selection, different ways of ranking context terms, and the use of the context term ranking in combination with disambiguation for translation selection.

References

1. Ballesteros, L., and Croft, B.: Dictionary Methods for Cross-Language Information Retrieval. In Proceedings of Database and Expert Systems Applications (1996) 791–801.
2. Ballesteros, L., Croft, W. B.: Resolving Ambiguity for Cross-Language Retrieval. In Proceedings of SIGIR (1998) 64–71.
3. Billhardt, H., Borrajo, D., Maojo, V.: A Context Vector Model for Information Retrieval. Journal of the American Society for Information Science and Technology, 53(3) (2002) 236–249.
4. Evans, D. A., Lefferts, R. G.: CLARIT–TREC Experiments. Information Processing and Management, 31(3) (1995) 385–395.
5. Fujii, A., Ishikawa, T.: Japanese/English Cross-Language Information Retrieval: Exploration of Query Translation and Transliteration. Computer and the Humanities, 35(4) (2001) 389–420.
6. Fung, P.: A Statistical View on Bilingual Lexicon Extraction: From Parallel Corpora to Non-parallel Corpora. In Proceedings of AMTA (1998) 1–17.
7. Fung, P., Yee, L. Y.: An IR Approach for Translating New Words from Nonparallel, Comparable Texts. In Proceedings of COLING-ACL (1998) 414–420.
8. Hull, D. A., Grefenstette, G.: Experiments in Multilingual Information Retrieval. In Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (1996) 49–57.
9. Grefenstette, G.: Evaluating the Adequacy of a Multilingual Transfer Dictionary for Cross Language Information Retrieval. In Proceedings of LREC (1998) 755–758.
10. Grefenstette, G.: The Problem of Cross Language Information Retrieval. In G. Grefenstette, ed., Cross Language Information Retrieval, Kluwer Academic Publishers (1998) 1–9.

11. Grefenstette, G., Qu, Y., Evans, D. A.: Mining the Web to Create a Language Model for Mapping between English Names and Phrases and Japanese. In Proceedings of the 2004 IEEE/WIC/ACM International Conference on Web Intelligence (2004) 110–116.
12. Ido, D., Church, K., Gale, W. A.: Robust Bilingual Word Alignment for Machine Aided Translation. In Proceedings of the Workshop on Very Large Corpora: Academic and Industrial Perspectives (1993) 1–8.
13. Jeong, K. S., Myaeng, S, Lee, J. S., Choi, K. S.: Automatic Identification and Back-transliteration of Foreign Words for Information Retrieval. Information Processing and Management, 35(4) (1999) 523–540.
14. Knight, K, Graehl, J.: Machine Transliteration. Computational Linguistics: 24(4) (1998) 599–612.
15. Kumano, A., Hirakawa, H.: Building an MT dictionary from Parallel Texts Based on Linguistic and Statistical Information. In Proceedings of the 15th International Conference on Computational Linguistics (COLING) (1994) 76–81.
16. Meng, H., Lo, W., Chen, B., Tang, K.: Generating Phonetic Cognates to Handel Named Entities in English-Chinese Cross-Language Spoken Document Retrieval. In Proc of the Automatic Speech Recognition and Understanding Workshop (ASRU 2001*)* (2001).
17. Pirkola, A., Puolamaki, D., Jarvelin, K.: Applying Query Structuring in Cross-Language Retrieval. Information Management and Processing: An International Journal. Vol 39 (3) (2003) 391–402.
18. Qu, Y., Grefenstette, G.: Finding Ideographic Representations of Japanese Names in Latin Scripts via Language Identification and Corpus Validation. In Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (2004) 183–190.
19. Qu, Y., Grefenstette, G., Evans, D. A.: Resolving Translation Ambiguity Using Monolingual Corpora. In Peters, C., Braschler, M., Gonzalo, J. (eds): Advances in Cross-Language Information Retrieval: Third Workshop of the Cross-Language Evaluation Forum, CLEF 2002, Rome, Italy, September 19–20, 2002. Lecture Notes in Computer Science, Vol 2785. Springer (2003) 223–241.
20. Qu, Y., Grefenstette, G., Evans, D. A: Automatic Transliteration for Japanese-to-English Text Retrieval. In Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (2003) 353–360.
21. Qu, Y., Hull, D. A., Grefenstette, G., Evans, D. A., Ishikawa, M., Nara, S., Ueda, T., Noda, D., Arita, K., Funakoshi, Y., Matsuda, H.: Towards Effective Strategies for Monolingual and Bilingual Information Retrieval: Lessons Learned from NTCIR-4. ACM Transactions on Asian Language Information Processing. (to appear)
22. Zhang, Y., Vines, P: Using the web for automated translation extraction in cross-language information retrieval. In Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (2004) 162–169.