A Method of Cross-Lingual Question-Answering Based on Machine Translation and Noun Phrase Translation using Web documents
— Yokohama National University at NTCIR-6 CLQA —

Tatsunori MORI Kousuke TAKAHASHI
Graduate School of Environment and Information Sciences
Yokohama National University
79-7 Tokiwadai, Hodogaya, Yokohama 240-8501, Japan
{mori,coske}@forest.eis.ynu.ac.jp

Abstract

We propose a method of English-Japanese cross lingual question-answering (E-J CLQA) that uses machine translation (MT) and an existing Japanese QA system. We also introduce noun phrase translation using Web documents in order to compensate the insufficiencies in the bilingual dictionary of the MT system. We combine several phrase translation techniques including 1) phrase translation using Wikipedia, 2) phrase translation using Web search results only, 3) phrase translation using Web search results and the information of pronunciation. The experimental result shows that the combination increases the coverage of translation and also improves the accuracy of E-J CLQA. However, the improvement is not so significant because the MT system works well for the NTCIR-6 E-J questions.

Keywords: *E-J CLQA, machine translation, noun phrase translation using the Web.*

1 Introduction

In recent years, the *question answering* (QA) has gained attention as a way of information access to a large amount of text. QA is the technology that extracts answers for user's natural-language question from a knowledge resource, i.e., a large amount of text. Since the knowledge resource may be a collection of documents from all over the world, the *cross-lingual* version of QA (CLQA) becomes one of important topics in the research area. CLQA is a task to answer to a given question written in a language by using a collection of documents written in other languages.

In this paper, we report the evaluation results of our CLQA systems at NTCIR-6 CLQA. We participated in the English-Japanese (E-J) task with three systems. Each of them uses a machine translation (MT) and an existing Japanese QA system. We also introduce noun phrase translation using web documents in order to compensate the insufficiencies in the bilingual dictionary of the MT system. The difference among those three systems lies in the combination of methods for noun phrase translation.

2 Related work

The Cross Language Evaluation Forum (CLEF)[1] introduced a new track termed *QA@CLEF* to test CLQA systems in 2003. QA@CLEF offers cross-language tasks of European languages like Dutch-English, French-English, and so on[13]. With regards to Asian languages, CLQA subtasks of English-Chinese, Chinese-English, English-Japanese and Japanese-English were provided at NTCIR-5 CLQA in 2005[16].

From the viewpoint of the treatment of multiple languages in CLQA, there are, at least, two kinds of approaches as follows:

1. For each target language, an individual QA engine that can treat the language is prepared. The cross-lingual process is achieved as the translation of questions into the target language.

2. One pivot language is assumed and one QA engine for the language is utilized. The cross-lingual process appears in the translation of question and/or the translation of documents.

While some proposals adopt the second approach[5, 12, 17, 14], the majority adopts the first approach because the pivot approach usually requires multiple translation stages, which tend to cause the translation error. We also follow the trend.

In general, the methodology based on the first approach consists of the following steps: 1) translate a given question (or keywords in the question) to a target language using an MT system or bilingual dictionaries, 2) perform passage (or document) retrieval and linguistically analyze the retrieved passages, 3) assign a score to each answer candidate according to the degree of matching between the question and the retrieved passage. Although an NE recognizer is usually adopted as a part of the linguistical analysis of retrieved passages, some systems extract semantic representation of sentences by a more deeper semantic analysis[2]. The documents on the Web is also exploited in some systems. For example, some of them validate the extracted answer candidate by using the Web[10]. The other system employs answer candidates obtained from the Web as well as the candidates in the document collection to be considered[4].

Roughly speaking, the processes following the query analysis may be regarded as a process of mono-lingual QA in the target language. Therefore, when our interest is not in the improvement of mono-lingual

QA systems, our main concern is the improvement of translation accuracy. From the viewpoint of improving translation accuracy, the translation of out-of-vocabulary (OOV) words is one of major problems to be deal with. It is our main focus of interest in this paper as described in the next section.

In many previous works, the set of Web documents is used as a resource for translating new words. For example, Zhang et al. [18] proposed a method to obtain translation candidates from the titles and snippets in the search result provided by a search engine. In the Web documents, Wikipedia is considered as more tractable resource. In order to improve the treatment of named entities and terms, Bouma et al.[3] extracted from English Wikipedia all pairs of lemma titles and their cross-links to the corresponding link to Dutch Wikipedia.

Among the these related works, our main contribution would be summarized as follows:

- We studied how to join the methods for translating OOV words to off-the-shelf MT systems.
- We examined the impact of the combination of translation methods for OOV words in CLQA.

3 System Overview

With regard to the matter of translation, many off-the-shelf machine-translation (MT) products are available in the market. Therefore we basically utilize one of off-the-shelf MT systems. However, in general, the quality of output of MT is not enough for the basis of CLQA. Especially, some proper nouns are not translated appropriately because of the OOV problem. The problem of OOV has very crucial impact on retrieval of question-related information from the text database.

With regard to the treatment of OOV phrases in combination with an MT product, there are at least two types of approaches: the treatment in the pre-editing phase, and the treatment in the post-editing phase.

The latter may be easily performed independent of an MT system. However, the approach can only treat the phrases that are not translated by an MT system. Incorrect translations by an MT system will still remain in translated question sentences.

On the other hand, the former approach depends on the process of MT systems. Fortunately, some of off-the-shelf English-Japanese machine translation systems treat Japanese strings embedded in an English sentence as unknown noun phrases in the process of translation. The behavior can be used for *pre-translation*, which is one of techniques to utilize Translation Memory (TM)[1].

In the situation of E-J CLQA, the pre-translation module firstly identifies noun phrases and, then, try to translate them using some external translation resources. According to the result of phrase translation, the translated Japanese phrases are substituted for the original English phrases to generate partially translated question sentences. The question sentences are passed to the subsequent MT process. This pre-translation approach has the advantage that we can control the identification of phrase to be translated with external resources.

With regard to the phrase translation using external resources, there are several different approaches that are worth employing.

Therefore, we adopt a hybrid accroach that is a combination of pre-translation and post-translation along with several phrase translation methods. It is shown as Algorithm 3.1. Here, the systems of the run ID $i = 1$, 2, and 3 correspond to 1) the proposed system, namely, a hybrid system with all of translation strategies, 2) the system that only uses the newly introduced translation strategy, and 3) the system that is a hybrid system but that only uses the translation strategies proposed by us in NTCIR-5 CLQA[14], respectively.

Note that $Translation\_Strategy\_A()$, $Translation\_Strategy\_B()$, and $Translation\_Strategy\_C()$ are described in the following sections.

Algorithm 3.1: EJ_CLQA(Q_E, N_{ans}, i)

comment: Q_E and N_{ans} are an English question and the number of answers to be found, respectively. $i = 1$,2,or 3 represents a run ID.

$Q\_set_J \leftarrow EJ\_Trans_i(Q_E)$
$Q\_type \leftarrow Guess\_Q\_Type(Q_E)$
comment: returns guessed question type for the question Q_E.

$A\_set \leftarrow \{\}$
for each $Q_J \in Q\_set_J$
  do $A\_set \leftarrow A\_set \cup Japanese\_QA(Q_J, Q\_type, N_{ans})$
comment: $Japanese\_QA(Q, Q\_type, N)$ returns a set of N-best pairs $\langle Ans, Score\rangle$ of answer and score for the given question Q_J and the question type Q_type.

$A\_selected \leftarrow select\_N\_best(A\_set, N_{ans})$
comment: selects N_{ans} best answers.

return $(A\_selected)$

Algorithm 3.2: $EJ\_Trans_1(Q_E)$

$Q\_set_J \leftarrow Translation\_Strategy\_A(Q_E)$
$Q\_set_J \leftarrow Q\_set_J \cup Translation\_Strategy\_B(Q_E)$
$Q\_set_J \leftarrow Q\_set_J \cup Translation\_Strategy\_C(Q_E)$
return (Q_set_J)

Algorithm 3.3: $EJ\_Trans_2(Q_E)$

$Q\_set_J \leftarrow Translation\_Strategy\_A(Q_E)$
return (Q_set_J)

Algorithm 3.4: $EJ\_Trans_3(Q_E)$

$Q\_set_J \leftarrow Translation\_Strategy\_B(Q_E)$
$Q\_set_J \leftarrow Q\_set_J \cup Translation\_Strategy\_C(Q_E)$
return (Q_set_J)

4 Translation strategy A: newly introduced parts for NTCIR-6 CLQA

Translation strategy A is a newly introduced strategy for NTCIR-6 CLQA, which is based on a

[1] In the pre-translation mode, TM system's proposals of translation are automatically inserted into source text. The produced hybrid text containing a mixture of source and target language elements is presented to human translators for further translation.

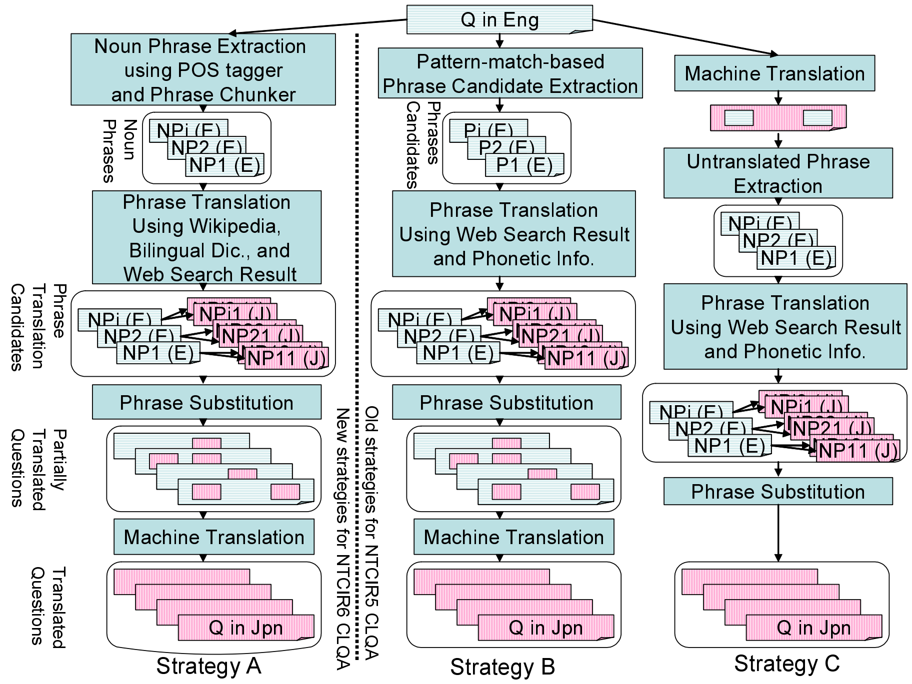

Figure 1. Overview of the proposed method (Part I: Translating a question)

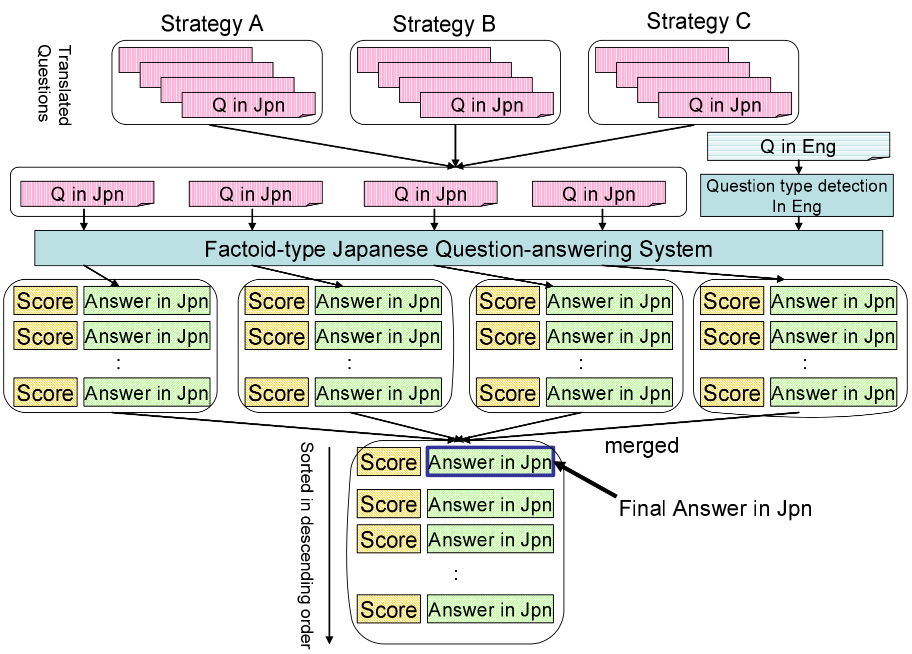

Figure 2. Overview of the proposed method (Part II: Finding answers)

pre-translation approach that utilizes 1) SVM-based noun phrase extraction, 2) phrase translation using Wikipedia, and 3) phrase translation using Web search results.

4.1 Noun phrase extraction

In order to extract noun phrases (NPs) in question sentences, we employ YamCha, which is an SVM-based chunking system[11], to identify phrase chunks in a sentence. For training the system to chunk phrases, the training and test data of the shared task for CoNLL-2000[2] is adopted. Since not only surface expression but also part-of-speech (POS) information are utilized as features for leaning, we also adopt Brill's Tagger for tagging POS to unseen sentences.

Basically, we extract NP chunks detected by the chunker for translation. However, it is possible that the an NP chunk extracted by the chunker may be a constituent of other larger phrase. Therefore, we also introduce the following rule to construct larger phrase.

- If NP chunks are joined with the possessive expression "'s" or words whose POS is "IN[3]", the sequence of chunks is regarded as a larger NP.

4.2 Translation of noun phrases using Wikipedia

Wikipedia is a Web-based, free content encyclopedia that is compiled collaboratively by volunteers[4]. Since it has a lot of articles in 250 languages, it can be used as a multilingual resource. If there are other articles in different languages that correspond to an article in another language, they are usually connected with hyper-links. Therefore, we can easily obtain multilingual translation of an entry term. Fukuhara et al. [6] propose use of Wikipedia to translate keywords, which users interested in, into multiple languages in order to analyze the occurrence of the concepts in multilingual weblog articles. Bouma et al. [3] also utilize Wikipedia in CLQA as described before.

In the E-J CLQA situation, if we want to translate an English phrase into Japanese, we firstly search for the phrase in English-version Wikipedia, then find out the link to a corresponding Japanese entry. The entry name is highly expected to be a proper translation of the source phrase.

In translation of English phrases into Japanese, we may use not only English entries, but also other entries in different languages that have almost same alphabet set as English, because many phrases transliterated from Japanese may be expected to have same spelling as English. For example, the word "Chiyoda," which is a name of ward in Tokyo, Japan, appears not in English Wikipedia but in versions of other languages like German, French, and so on[5].

The translation process is shown in Algorithm 4.1. Here, $langCode$ represents one of language codes defined in ISO 639.

[2]http://www.cnts.ua.ac.be/conll2000/chunking/

[3]Preposition or subordinating conjunction

[4]http://en.wikipedia.org/wiki/Main_Page

[5]English Wikipedia has not the entry "Chiyoda" but the entry "Chiyoda, Tokyo."

Algorithm 4.1: WIKIPEDIATRANSEJ($PhraseE$)

for each $langCode \in$ {'en', 'de', 'fr', 'es', 'it', 'nl', 'pt', 'lt', 'no', 'pl'}
  do { $art \leftarrow Wikipedia(PhraseE, langCode)$
    if (art has a link to a Japanese entry $PhraseJ$)
      then return ($PhraseJ$) }
return ('')

4.3 Translation of noun phrases using Web search results

When we submit a phrase in the source language to a Web search engine to find out documents in the target language, many of retrieved documents are expected to contain not only the query phrase but also the phrases in the target language that related to the query phrase. Based on the expectation, Zhang et al. [18] proposed a method to obtain translation candidates from the titles and snippets in the search result provided by a search engine. They also proposed a scoring method that estimate the appropriateness of the candidate in terms of translation. We adopt the technique as one of methods to translate OOV phrases, while we modify the scoring function so as to be able to utilize in the situation of CLQA. We also introduce a method to determine whether proper translation candidates are obtained or not.

4.3.1 Stage I: Obtaining translation candidates

Firstly, a list of translation candidates with frequency information is obtained by Algorithm 4.2, which is exactly same as Zhang's method. In the situation of E-J CLQA, an English phrase $Phrase$ is passed to the algorithm in order to find out Japanese translation candidates $\{C_i\}$ from the result of retrieval of Japanese documents.

Algorithm 4.2: WEBTRANSCAND($Phrase$)

procedure LCCS(str_1, str_2)
  comment: extract set of the common longest contiguous substrings with frequency.
  return ($\{\langle lccs_1, freq_1\rangle, \langle lccs_2, freq_2\rangle, \ldots\}$)

procedure MERGEFREQ(set_1, set_2)
  comment: merge two sets of strings by accumulating the frequency.
  return ($\{\langle str_1, freq_1\rangle, \langle str_2, freq_2\rangle, \ldots\}$)

main
  $\langle T, S\rangle \leftarrow CallExternalSEngine(Phrase, N_d)$
  comment: T and S are the arrays of titles and snippets, respectively. The index corresponds to the rank in the search result. N_d is the number of document to be retrieved.
  $TC \leftarrow \{\}$
  for $i \leftarrow 1$ **to** $n - 1$
    do { **for** $j \leftarrow i + 1$ **to** n
      do { $LCSS_T \leftarrow LCCS(T_i, T_j)$
        $LCSS_S \leftarrow LCCS(T_i, S_j)$
        $TC \leftarrow MergeFreq(TC, LCSS_T)$
        $TC \leftarrow MergeFreq(TC, LCSS_S)$ } }
  return (TC)
  comment: $TC = \{\langle C_1, freq_1\rangle, \langle C_2, freq_2\rangle, \ldots\}$, where C_i and $freq_i$ are a translation candidate of $Phrase$ and its frequency, respectively.

4.3.2 Stage II: Assigning score to each candidate

Secondly, a score is assigned to each candidate to select more plausible candidates. The original proposal adopts the following equation for the scoring function.

$$\begin{aligned} Scorg(C_i) &= \alpha \cdot \frac{freq_i}{maxFreq(TC)} \cdot ITF(C_i) \\ &\quad +(1-\alpha) \cdot \frac{1}{Rank(C_i)+\beta} \qquad (1) \\ maxFreq(TC) &= \max\{freq_j \mid \langle C_1, freq_1\rangle \in TC\} \end{aligned}$$

Here, the term $ITF(C_i)$ is "Inverse Translation Frequency," which is the inverse of translation frequency that represents how many times the translation candidate C_i (a Japanese phrase in E-J CLQA) appears in different candidate lists for other (English) phrases. However, it should to be noted that $ITF(C_i)$ is properly calculate only when we want to translate a number of phrases simultaneously. It is not suitable for the CLQA situation. In stead of ITF, therefore we introduce the factor of length of candidate. Since the algorithm of extracting translation candidate tends to produce shorter translation candidates, we give "reward" to longer candidates. Our revised version of score is shown in Equation (2).

$$\begin{aligned} Screv(C_i) = \log_2(length(C_i)+\gamma)\cdot \qquad (2) \\ (\alpha \cdot \frac{\log_2(freq_i)}{\log_2(maxFreq(TC))} + (1-\alpha)\cdot \frac{1+\beta}{Rank(C_i)+\beta}) \end{aligned}$$

4.3.3 Stage III: Selecting an appropriate translation candidate

Until this stage, we obtain an ordered translation candidate list for a source noun phrase. However, it is not guaranteed that the list contains *proper* translation. Therefore the system have to determine whether a given list should be employed or not.

In order to cope with the problem, we introduce the same method as a list-type QA processing proposed by Ishioroshi et al. [7] and Mori et al. [15]. They assume that the distribution of answer scores contains a mixture of two normal distributions, $\phi_p(x;\mu_p,\sigma_p)$ and $\phi_n(x;\mu_n,\sigma_n)$, i.e., those of the correct answers and incorrect answers, where μ and σ are the average and the standard deviation, respectively. Under these assumptions, the correct answers may be separated from the mixture of the distributions by using the EM algorithm. Figure 3 shows an example of the score distribution in the case that the score distribution of the correct answers is separable from that of the wrong answers.

In their proposal, the appropriateness of an answer candidate list is assumed to be measured by $\mu_p - \mu_n$ in Figure 3. When the system cannot find appropriate answer candidates, the scores of highly ranked answer candidates are not very high and have almost the same distribution as that of the lower ranked candidates. Conversely, if the value $\mu_p - \mu_n$ is relatively large, we can expect that an appropriate list is obtained.

In this paper, we apply the same method described above to the score distribution of translation candidate list. If the value $\mu_p - \mu_n$ is larger than a certain threshold Min_avg_diff, we judge that the list surely contains appropriate translation candidates. In the situation, each candidate whose normalized score is larger than a certain threshold $\mu_p + N_\sigma \cdot \sigma$ is regarded as appropriate in terms of score.

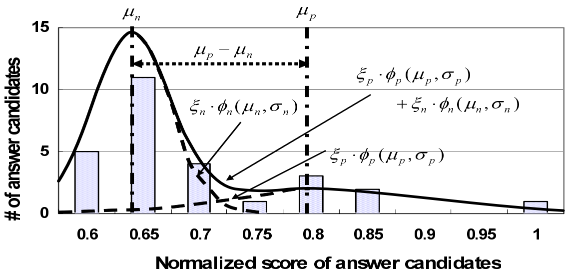

Figure 3. An example distribution of answer scores

4.3.4 Stage IV: Filter out garbage candidate

The candidate list usually may contain a number of garbage strings because the algorithm depends on a very simple extraction method. At this stage, "obvious garbage strings" should be deleted. Therefore, we introduce a criterion that selects candidate $\langle C_i, freq_i\rangle$ that satisfy all of the following conditions and eliminate other candidates: 1) C_i is not a Web related phrase, 2) C_i only consists of Japanese characters, 3) $freq_i \geq Min\_freq$, and 4) $length(C_i)/\#ofwords(Phrase) \geq Min\_length\_ratio$.

5 Translation strategy B and C: introduced parts for NTCIR-6 CLQA

Both of translation strategies B and C are developed by us for the former CLQA at NTCIR-5[14]. Each of them employs a translation strategy that searches for the loan words that are originally Japanese words, then, translates the loan words into the original Japanese words using the Web documents and the information of pronunciation. They also utilize a simple pattern-match-based method to find proper Japanese translations for English phrases using the Web documents.

With regard to the combination of phrase translation and MT, the translation strategy B is a pre-translation method. On the other hand, the translation strategy C is a post-translation method.

Because of the limitation of space, we cannot enter into further details. Please see [14].

6 Experimental result and discussion

6.1 Experimental settings

Our formal runs are performed in November 4-5, 2006. As a machine translation system and a bilingual dictionary, we adopt an off-the-shelf MT product[8] and the EDR E-J dictionary[9]. The MT product has the "pre-translation" function described in Section 3.

Translation strategy A

All parameters are tuned with a small preliminary experiment. The setting of system parameters is as follows: $N_d = 30$ in Algorithm 4.2, $\alpha = 0.5$, $\beta = 10.0$, $\gamma = 1.0$ in Equation (2), $Min\_avg\_diff = 0.2$, $N_\sigma = 1.0$ in Stage III of translation using Web search results, $Min\_freq = 4$, $Min\_length\_ratio = 0.2$ in Stage IV. As a web search engine, Web service by Yahoo! Japan was used[6].

Translation strategies B and C

All parameters and other settings are same as our formal run in NTCIR-5 CLQA[14]. As a web search engine, Google SOAP Search API was used[7]

6.2 Results and discussion

6.2.1 Performance in translation of proper noun phrases

There are 195 proper noun phrases in the question set for the E-J task. We evaluate the performance in translation of these phrases using the following measures. The result is shown in Table 1.

- Measures for evaluation of proper noun detection

$$Recall(R) = \frac{N_C^d}{N_T^d} \quad (3)$$

$$Precision(P) = \frac{N_C^d}{N_E^d} \quad (4)$$

where

N_T^d Number of source proper noun phrases that are to be translated, but are not in translation dictionaries,

N_E^d Number of source phrases extracted by the system,

N_C^d Number of source proper noun phrases that are extracted correctly.

- Measures for evaluation of proper noun translation

$$Hit(H) = \frac{N_H^t}{N_E^d} \quad (5)$$

$$Trans.Accuracy1(A1) = \frac{N_{C-JJ}^t}{N_H^d} \quad (6)$$

$$Trans.Accuracy2(A2) = \frac{N_{C-Sem}^t}{N_H^d} \quad (7)$$

where

N_H^t Number of source extracted phrases for which the system can find some translation candidates in the target language,

N_{C-JJ}^t Number of phrases which the system can translate into a set of translation candidates that includes at least one corresponding target phrase in the Japanese questions for the J-J task,

N_{C-Sem}^t Number of phrases which the system can translate into a set of translation candidates that includes at least one semantically correct translation.

It should be noted that some methods of proper noun translation return plural translation candidates.

Since the newly introduced method (A) detects all candidates of noun phrases, the recall in detection is much higher than the method for NTCIR-5 (B+C). To the contrary, the precision in detection becomes lower. The combined method (A+B+C) can detect almost all proper noun, while the precision is still low. With respect to translation accuracy, the new method has better performance than the method for NTCIR-5.

We also examined how many proper noun phrases in questions are correctly translated. As the result in Table 2 shows, the combination of each translation method improves the coverage of proper noun translation. It should be noted the MT system works well for the NTCIR-6 E-J questions. The difference between “MT+A+B+C” (155) and “MT only” (138) is only 17 phrases. The actual improvement is shown in Table 3.

On the contrary, according to the our experiment in NTCIR-5, the question set for the NTCIR-5 E-J contains more phrases that are difficult for the MT system to translate. For example, the system “MT+B+C” could translate 102 phrases while “MT only” was only 64 phrases.

6.2.2 Performance in terms of E-J CLQA

The performance in terms of E-J CLQA is evaluated with the final answer candidate that are compiled on the Google spreadsheet in cooperation of active participants. The result is shown in Table 4.

Although the system “MT+A+B+C” has better performance than other settings, the difference between “MT+A+B+C” and “MT only” is not significant. The reason should be that the MT system works well for the NTCIR-6 E-J questions and the actual improvement by phrase translation proposed here is not so significant, as described Section 6.2.1.

6.3 Failure analysis of phrase translation

6.3.1 Failure in extracting noun phrases

The noun phrase extractor sometimes wrongly extract adjacent proper nouns as one word like the following example.

Question Where did former Spice Girl Posh Spice hold her wedding ceremony?

Extracted NP “Spice Girl Posh Spice”

Correct NPs “Spice Girl” and “Posh Spice”

Proper nouns in single word like “Laila” also tend not to be extracted.

[6]http://developer.yahoo.co.jp/search/
[7]http://code.google.com/apis/soapsearch/

Table 1. Performance in translation of noun phrases

Strategy	R	P	H	A1	A2
B and C	0.569	0.689	0.416	0.343	0.403
A only	0.769	0.344	0.317	0.522	0.725
A,B and C	0.979	0.380	0.339	0.512	0.665

R: Recall P: Precision
H: Hit A1: Trans. Accuracy 1 A2: Trans. Accuracy 2

Table 2. Number of correctly translated proper noun phrases

Evaluation criterion	MT only	B+C	A	A+B+C	MT+A+B+C	NG
J-J Q.	132	24	63	79	149	45
Sem.	138	28	89	104	155	41

J-J Q.: Translation candidate appears in the Japanese question for J-J task.
Sem. : Translation candidate is semantically correct.
A, B, and C : Strategy A, B, and C
NG : Any correct translation are not obtained.

Table 3. Number of correctly translated proper noun phrases which the MT cannot translate

Evaluation criterion	MT NG	B+C	A	A+B+C	NG
J-J Q.	65	5	15	18	47
Sem.	61	5	19	22	39

J-J Q.: Translation candidate appears in the Japanese question for J-J task.
Sem. : Translation candidate is semantically correct.
MT NG: The MT system cannot correctly translate it.
A, B, and C : Strategy A, B, and C
NG : Any correct translation are not obtained.

Table 4. Performance in E-J CLQA

Strategy	Run ID	Acc	MRR	TOP5	Acc+U	MRR+U	TOP5+U
MT only		0.175	0.195	0.230	0.185	0.230	0.315
MT+B+C	Forst-E-J-03	0.170	0.193	0.235	0.180	0.229	0.325
MT+A	Forst-E-J-02	0.170	0.192	0.230	0.180	0.231	0.325
MT+A+B+C	Forst-E-J-01	0.175	0.197	0.230	0.195	0.244	0.320
JJ QA	Forst-J-J-01	0.310	0.361	0.440	0.335	0.410	0.525

Acc: Accuracy
+U: Unsupported answers are allowed
JJ QA: Japanese monolingual QA system with correct Japanese questions.

6.3.2 Failure in phrase translation by using Wikipedia

Translation using Wikipedia mostly works well when it is applicable. However, it has unwilling tendency to translate a target noun phrase into an official name of translation instead of a popular translation. For example, "Akutagawa Prize" was translated into the phrase "akutagawa ryunosuke shou" instead of "akutagawa shou." The phrase "Morse code" was translated into "mo-rusu fugou" instead of "mo-rusu shingou," which is more popular.

6.3.3 Failure in phrase translation by using Web search results

The method tends to fail in translation of long noun phrases. For example, the phrase "University of Hawaii at Manoa" should be translated into "hawai daigaku manoa kou," but it was insufficiently translated into "hawai daigaku."

It also has a tendency to translate a word into a related word in the target language. For example, the phrase "FIFA president" is wrongly translated into a related word "sakkah (football)" instead of the correct translation "FIFA-kaichou."

7 Conclusion

In this paper, we introduced noun phrase translation using Web documents in order to compensate the insufficiencies in the bilingual dictionary of the MT system for CLQA. We combine several phrase translation techniques including 1) phrase translation using Wikipedia, 2) phrase translation using Web search results only, 3) phrase translation using Web search results and the information of pronunciation. The experimental result shows that the combination increases the coverage of translation and also improves the accuracy of E-J CLQA. However, the improvement is not so significant because the MT system works well for the NTCIR-6 E-J questions.

Acknowledgment

The authors would like to express their thanks to Masami Kawagishi, who developed the earlier version of our system. This study was partially supported by Grant-in-Aid for Scientific Research on Priority Areas (No.18049031 and No.19024033) from the Ministry of Education, Culture, Sports, Science and Technology, Japan.

References

[1] Cross language evaluation forum. http://clef.iei.pi.cnr.it/, 2005.

[2] K. Ahn, B. Alex, J. Bos, T. Dalmas, J. L. Leidner, and M. B. Smillie. Cross-lingual question answering with QED. In *Working Notes for the CLEF 2004 Workshop*, Sept. 2004.

[3] G. Bouma, I. Fahmi, J. Mur, G. van Noord, L. van der Plas, and J. Tiedemann. The University of Groningen at QA@CLEF 2006: Using Syntactic Knowledge for QA. In *Working Notes for the CLEF 2006 Workshop*, Sept. 2006.

[4] G. Bourdil, F. Elkateb, B. Grau, G. Illouz, L. Monceaux, I. Robba, and A. Vilnat. How to answer in English to questions asked in French: by exploiting results from several sources of information. In *Working Notes for the CLEF 2004 Workshop*, Sept. 2004.

[5] M. Bowden, M. Olteanu, P. Suriyentrakorn, J. Clark, and D. Moldovan. LCC's PowerAnswer at QA@CLEF 2006. In *Working Notes for the CLEF 2006 Workshop*, Sept. 2006.

[6] T. Fukuhara, T. Murayama, and T. Nishida. Analyzing concerns of people from weblog articles. *AI & Society*, (to appear).

[7] M. Ishioroshi and T. Mori. A method of list-type question-answering based on the distribution of answer score generated by a ranking-type Q/A system. SIG Technical Reports 2005-NL-169, Information Processing Society of Japan, Sept. 2005. (in Japanese).

[8] I. Japan. *Internet Hon'yaku-no Ousama (The king of Internet-translation Bilingual) Bilingual Version 5*, 2002.

[9] Japan Electrronic Dictionary Research Institute. *EDR Electronic Dictionary Specification (version 2.0)*, 2001.

[10] V. Jijkoun, G. Mishne, M. de Rijke, S. Schlobach, D. Ahn, and K. Müller. The University of Amsterdam at QA@CLEF 2004. In *Working Notes for the CLEF 2004 Workshop*, Sept. 2004.

[11] T. Kudo and Y. Matsumoto. Chunking with support vector machines. In *Proceedings of NAACL 2001*, pages 1–8, 2001.

[12] D. Laurent, P. Seguela, and S. Negre. Cross Lingual Question Answering using QRISTAL for CLEF 2006. In *Working Notes for the CLEF 2006 Workshop*, Sept. 2006.

[13] B. Magnini, D. Giampiccolo, P. Forner, C. Ayache, P. Osenova, A. P. nas, V. Jijkoun, B. Sacaleanu, P. Rocha, and R. Sutcliffe. Overview of the CLEF 2006 Multilingual Question Answering Track. In *Working Notes for the CLEF 2006 Workshop*, Sept. 2006.

[14] T. Mori and M. Kawagishi. A Method of Cross Language Question-Answering Based on Machine Translation and Transliteration: Yokohama National University at NTCIR-5 CLQA1. In *Proceedings of the Fifth NTCIR Workshop Meeting*, pages 215–222, 12 2005.

[15] T. Mori, S. Kawaguchi, and M. Ishioroshi. Answering Contextual Questions Based on the Cohesion with Knowledge. In *Computer Processing of Oriental Languages. Beyond the Orient: The Research Challenges Ahead (Proceedings of the 21st International Conference on the Computer Processing of Oriental Languages (ICCPOL 2006))*, volume 4285 of *Lecture Notes in Computer Science*, pages 1–12. Springer, Dec. 2006.

[16] Y. Sasaki, H.-H. Chen, K. hua Chen, and C.-J. Lin. Overview of the ntcir-5 cross-lingual question answering task (CLQA1). In *Proceedings of the Fifth NTCIR Workshop Meeting*, Dec. 2005.

[17] K. Shimizu, T. Akiba, A. Fujii, and K. Itou. Bidirectional Cross Language Question Answering using a Single Monolingual QA System. In *Proceedings of the Fifth NTCIR Workshop Meeting*, pages 455–462, Dec. 2005.

[18] J. Zhang and L. Sun. Using the web corpus to translate the queries in cross-lingual information retrieval. In *Proceedings of 2005 IEEE International Conference on Natural Language Processing and Knowledge Engineering*, 2005.