Finding Translation Correspondences from Parallel Parsed Corpus for Example-based Translation

Eiji Aramaki* , Sadao Kurohashi, Satoshi Sato*, Hideo Watanabe*****

* Graduate School of Informatics, Kyoto University
Yoshida-honmachi, Sakyo
Kyoto 606-8501, Japan
{aramaki, sato}@pine.kuee.kyoto-u.ac.jp
** Graduate School of Information Science and Technology, the University of Tokyo
Hongo 7-3-1, Bunkyo-ku ,Tokyo 113-8656,Japan
kuro@kc.t.u-tokyo.ac.jp
*** IBM Research, Tokyo Research Laboratory
1623-14 Shimotsuruma, Yamato,
Kanagawa 242-8502, Japan
hiwat@jp.ibm.com

Abstract

This paper describes a system for finding phrasal translation correspondences from parallel parsed corpus that are collections paired English and Japanese sentences. First, the system finds phrasal correspondences by Japanese-English translation dictionary consultation. Then, the system finds correspondences in remaining phrases by using sentences dependency structures and the balance of all correspondences. The method is based on an assumption that in parallel corpus most fragments in a source sentence have corresponding fragments in a target sentence.

Keywords

Example-based Translation, Finding Phrasal Correspondence, Phrasal Alignment, Parallel Corpus, Dependency Structure

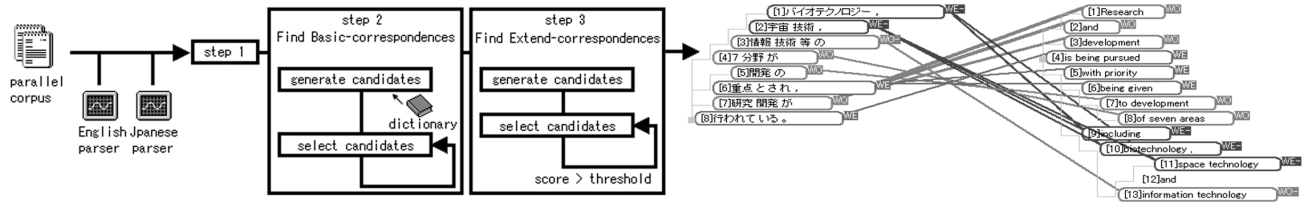

Figure 1: System Image and an Example of System Output

Introduction

Example-based translation system requires a large set of translation patterns [1]. Over the last decade, the sentence alignment and word alignment have been explored and achieved numerous successes by using statistical approach. In contrast, a fewer results are reported in phrasal alignment. In statistical phrasal alignment acquires the bilingual-correspondences appear with high frequency. However the coverage is low [2] [3].

In parallel corpus, we think that fragments in a source sentence usually have corresponding fragments in a target sentence. So, this paper proposes a system finds correspondences by using dependency structures and a balance of correspondences.

This paper is organized as follows: In the next section, we present the overview of our approach. In section 3, we describe our methods in detail. In section 4, experiments and results are given. In section 5, we describe a conclusion.

Overview of our approach

We have developed a system finds phrasal correspondences in parallel parsed corpus that are collections of English and Japanese sentences pairs.
A method of the system has 3 steps (figure 1).

Step1: The system acquires phrasal dependency structures.

Step2: The system finds phrasal correspondences by consultations of a Japanese-English translation dictionary.

Step3: The system finds phrasal correspondences in remaining phrases.

The system has two characteristics. (1) The system first acquires phrasal structure (in step 1) and then finds correspondences. (2) The system finds correspondences with an assumption that fragments in a source sentences have corresponding fragments in a target sentences (in step 3).

In consulting a Japanese-English word dictionary, most words can find a unique translation candidate in a target language, but some words have two or more candidates.

In figure 2, there are two English words, “technology,” and each of them has two candidates. To find phrasal correspondences the system does not determine correspondences in word level. The system acquires word correspondence candidates (we call them *word-links* in this paper) by dictionary consultations and finds phrasal correspondence by using word-links. The basic idea is that a phrasal correspondence candidate consists of phrases connected by many word-links is plausible.

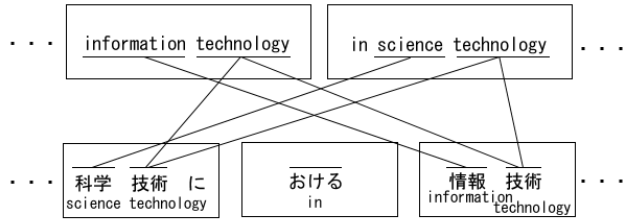

Figure 2. Word-links and Phrasal Correspondences

After finding correspondences by dictionary consultations, there are some remaining phrases. The system finds new correspondences in remaining phrases that are not included in correspondences. This procedure is based on dependency structures and the balance of correspondences.

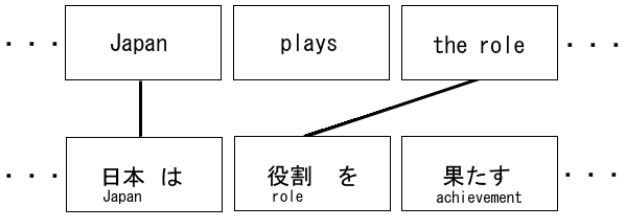

Figure3. Finding Correspondences in Remaining Phrases

In the above example (Figure. 3), the system has already found out two correspondences [Japan / nihon (Japan)], [role / yakuwari (role)]. Near them, there are only two phrases that are not included in correspondences. The system regards the phases [play / hatasu (achievement)] as a plausible correspondence.

However, such correspondences have less accuracy than correspondences found by dictionary consultations. So the system controls how actively the system finds correspondences by a given threshold.

Method

This section describes 3 steps in detail.

STEP 1: Phrasal Dependency Structures

The system parsed English and Japanese sentences and acquires their dependency structures. First, the system applies following rules and acquires phrasal dependency structures.

1. In English, a function word is grouped into a following content word. In Japanese, a function word is grouped into a last content word. However, a parallel-relation word, for example, “and” and “or,” they are not content-words, are considered as a phrase.
2. A compound noun is considered as one phrase.
3. Auxiliary verbs are groped into a following verb.

In Applying above rules, if words that should be grouped into the same phrase have different parents, the system does not group them.

STEP 2: Finding Basic-correspondence

Finding correspondences has 2 steps, (1) generates candidates, and (2) selects candidates. The system consults a Japanese-English word dictionary for content words. Then the system generates correspondence candidates, and selects candidates by 3 criteria. We call such correspondences *basic-correspondences*.

STEP 2-1: Generate Basic-correspondence Candidates

By dictionary consultations all contents words in sentences, the system acquires word-links. A consultation a Japanese-English word dictionary has 2 exception-handlings. (1) When the length of a Japanese word is 2 or more characters, the system consults a dictionary by dividing it into two parts. (2) When the system consults a dictionary, the system ignores a Japanese “SU-RU (-ing)” and English “ly”, “d”, “ed”, “s”, “es”, and “ies.” We admit the exceptions in order to compensate the insufficiency of a dictionary, and they do not change the algorithm in essence.

The system regards phrases are connected by one or more word-links as a candidate. The system generates candidates consist of two or more phrases shown in figure 4. The candidates are subject to the following conditions. Phases in a candidate are adjacent each other. However a parallel-relation word, for example, “and” and “or,” are exceptions and the system can interpose such words between phrases in a candidate.

In figure 4, there are two word-links form a Japanese phrase to two English phrases. Two English phrases are interposed by function words “and.” Then the system generates 3 candidates in figure 4.

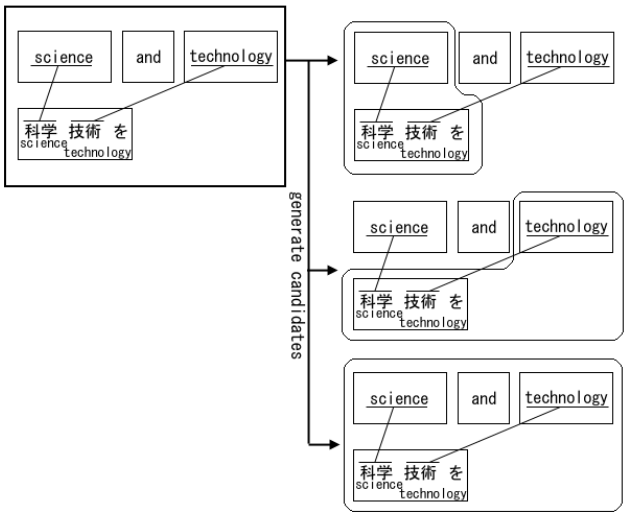

Figure 4. Generate Candidates of Basic Correspondences

STEP 2-2: Selection Basic-correspondence Candidates

The system selects candidates of basic correspondences by follow 3 criteria. For avoiding correspondence conflictions, when the system selects *basic-correspondences*, the other conflicting candidates are rejected. In this paper, a confliction means some correspondences have the same phrases.

When there are no *basic-correspondences* to select, the system finishes step 2 procedures.

Criterion 1: Sufficiency

The more priority is given to a sufficient correspondence. The sufficiency (S) is defined below:

$$S = \frac{\text{count (word - link)} \times 2}{\text{count (JP content word)} + \text{count (EN content word)}}$$

Count (JP content word) is the number of content words in Japanese phrases in a candidate. Count (EN content word) is the number of content words in English phrases in a candidate. Count (word-link) is the number of word-links acquired by dictionary consultations.

In figure 5, candidate A has more sufficiency than candidate B. A correspondence without remaining content words has the most priority like candidate A.

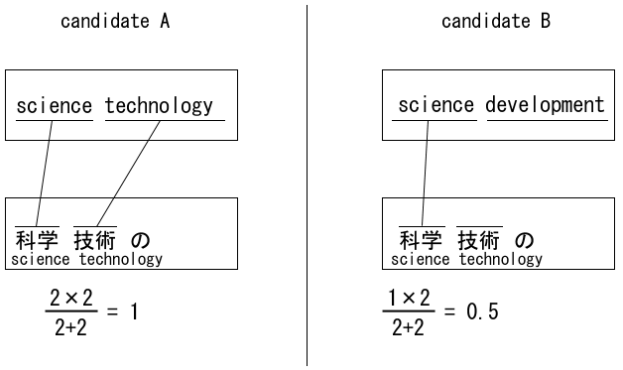

Figure 5. Candidates and their Sufficiency

The candidates are classified into 4 types according to following conditions. These types are used in step 3.

Type 1. Close: All content words in a candidate have word-links correspond each other. In figure 5, candidate A is a close correspondence.

Type 2. EN Shortage: One or more English content words have no word-links correspond Japanese phrases in a candidate.

Type 3. JP Shortage: One or more Japanese content words have no word-links correspond English phrases in a candidate.

Type 4. JP+EN Shortage: One or more English and Japanese content words have no word-links correspond each other phrases in a candidate. In figure 5, candidate B is a JP+EN shortage correspondence.

Criteria 2: Size

The more priority is given to a big size correspondence. The size is defined below:

Size = count (JP phrase) + count (EN phrase)

Because the phrases in a candidate are adjacent, too long phrases are not selected.

Criteria 3: Support

The more priority is given to a correspondence supported by the other candidates. The support is defined below:

Support = count (near-candidates)

The definition of "near," we used 6 as the distance threshold of this procedure currently.

For example, in figure 6, phrase [science] has two candidates. The system gives priority thick line candidates, because there is another candidate [technology,]

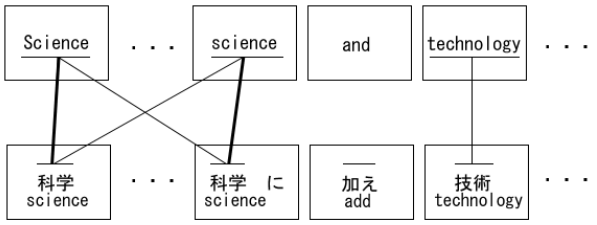

Figure 6. Candidates Supported by Surrounding Correspondences

By the above 3 criteria, a candidate is selected as a *basic-correspondence*. The criteria have different priority. The sufficiency is the most priority. The size has more priority than the support.

So, first the system selects candidates by their sufficiency. When there are the same sufficiency candidates, the system selects them by the size.

STEP 3: Finding extend-correspondence

After finding *basic-correspondences*, there are usually some remaining phrases. Then, the system finds correspondences in remaining phrases. We call such correspondences *extend-correspondences*. Finding correspondences has 2 steps, (1) generates candidates, and (2) selects candidates by their score.

STEP 3-1: Generate extend-correspondence candidates

Remaining phrases really have no appropriate corresponding phrases, otherwise the system should find or modify correspondences as follows.

1: The system adds a remaining phrase to a *basic-correspondence*.

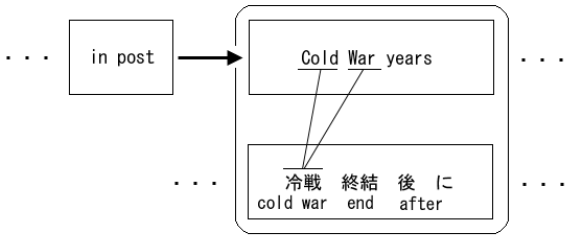

Figure 7: The system modifies a basic-correspondence

2: The system finds a new correspondence consists of two remaining phrases.

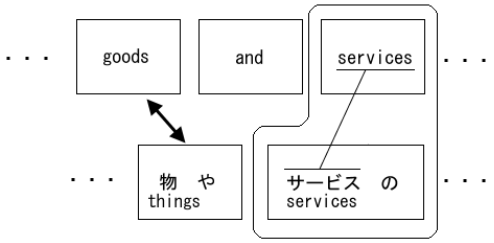

Figure 8: The system finds a new correspondence

The system examines all remaining phrases and generates *extend-correspondence* candidates by accounting above 2 possibilities.

STEP3-2: Selection Extend-correspondence candidates

We defined a score of *extend-correspondence* candidates and give a threshold to the system. The system sorts candidates by score. When the system selects a candidate, the other conflicting candidates are rejected. When the score is less than the threshold, the system finishes step 3. Currently we used a following score defined by dependency structures and a balance of correspondences.

$$\text{score} = \frac{B}{4} + \sum_{k=1}^{n} \frac{X}{J(k) + E(k)}$$

n is the number of *basic-correspondences* which is near an *extend-correspondence* candidate. The definition of "near," we used 2 phrasal distances as the threshold.

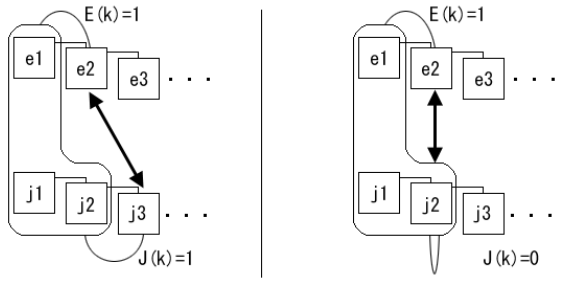

Figure 9: E(k) and J(K)

E(k) and J(k) are distances as shown in figure 9. E(k) is the distance between an English remaining phrase and an English phrase in basic-correspondence(k). J(k) is the distance between a Japanese remaining phrase and a Japanese phrase in basic-correspondence(k).

X is defined by the state near the candidate. X is initially 1 and X is multiplied by the number in table 1 when the condition is submitted.

Table 1: X

N	Conditions
2	The system adds a Japanese remaining phrase to a JP shortage type *basic-correspondence*, or adds an English remain-phrase to an EN shortage type *basic-correspondence*.
1/8	The system adds a remaining phrase to a close type *basic-correspondence*.
1/2	The system adds a remain-phrase to a correspondence that is different from the remaining phrase in part of speech. Where a phrasal part of speech is classified into 2 types according to following. If one or more verbs are involved in a phrase, we consider the phrase as "VP", else "NP".
1/2	The system adds a remaining phrase to a correspondence that has no dependency with the phrase.

B is defined by the ratio of basic-correspondences in English and Japanese sentences. If most of phrases are included in *basic-correspondences*, all correspondences have high score.

$$B = \frac{\text{count (phrases in basic - correspondences)}}{\text{count (phrases in JP and EN sentences)}}$$

Count (phrases in basic-correspondences) is the number of phrases all *basic-correspondences* include. Count (phrases in JP and EN sentences) is the number of phrases in English and Japanese sentences.

Experiments

We used two corpus shown in table 2.

Table 2: Corpus Feature

Corpus	Feature
Corpus A White Paper (2246 sentences) [10]	Long sentences. Word domain is narrow.
Corpus B Example sentences in a dictionary (2817 sentences)	Short sentences

We acquired test-set 200 sentences by extracting 100 sentences form each corpora under follow 3 conditions.

Condition 1: A pair of sentences has one-to-one sentence correspondence.

Condition 2: The number of both English and Japanese phrases differed by less than 2:1 ratio.

Condition 3: The number of both English and Japanese phrases is less than 20 phrases

We made a parsed bilingual corpus by using the KNP[4] Japanese parser (developed by Kyoto University) for Japanese Sentences and ESG[5] English parser (developed by IBM Watson Research Center) for English sentences.

We evaluated the system output as follows. We tagged on correct target phrases every phrase in 200 test-set sentences. If a system output correspondence exactly equal with a pre-aligned correspondence, we regard it as **correct**. If a correspondence which system output partly matches with a pre-aligned correspondence, we regard it as **near-correct**. Else we regard it as **wrong**.

In *extend-correspondences* selection, the system has the threshold. The threshold is about 3 at the highest. When the threshold=3, the system finds no extend-correspondences, and all found correspondences are basic-correspondences. By contrast, when the threshold=0, the system finds extend-correspondences the most actively.

Initially we set threshold=0, then the threshold increase by 0.5. Results are follows.

The number of correspondence phrases is shown in table 3. *Basic-correspondences* are bigger than *extend-correspondences*, because most of new correspondences are one-to-one phrases.

Table 3. The Number of Phrases in Correspondences

	English	Japanese
Basic-correspondence	2.20	3.20
Extend-correspondence	1.67	2.73

In figure 10-13, graphs show the ratio and the number of found correspondences.

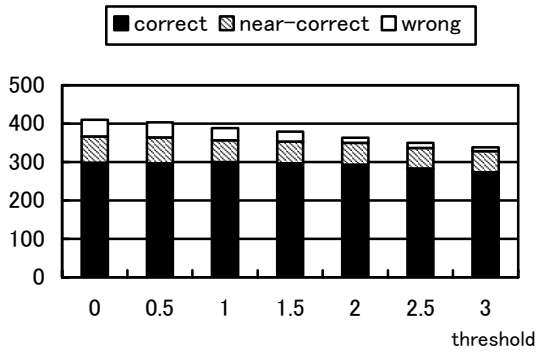

Figure 10: The Number of Corpora A Evaluation

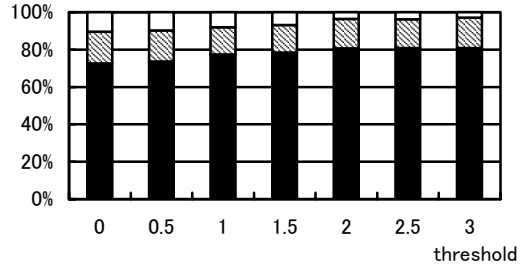

Figure 11: The Ratio of Corpora A Evaluation

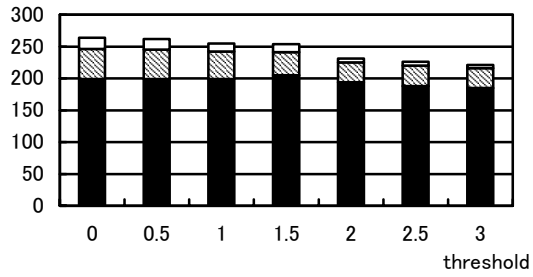

Figure 12: The Number of Corpora B Evaluation

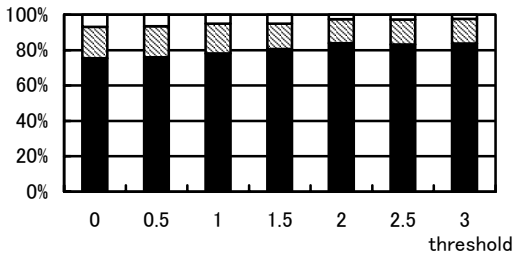

Figure 13: the Ratio of Corpora B Evaluation

The more the system finds *extend-correspondences*, the more wrong correspondences are found, because *extend-correspondences* lack accuracy. However *extend-correspondences* are not found by dictionary consultations, so they are important. We defined precision and recall as follow.

$$\text{precision} = \frac{\text{count(correct)} + \frac{1}{2} \times \text{count (near - correct)}}{\text{count(found)}}$$

$$\text{recall} = \frac{\text{count(correct)} + \frac{1}{2} \times \text{count(near - correct)}}{\text{count (pre - aligned)}}$$

Figure 14 shows the system finds 77% of pre-aligned correspondences as correct or near-correct correspondences. When recall is 77%, the threshold is 0 and the system finds *extend-correspondences* the most actively.

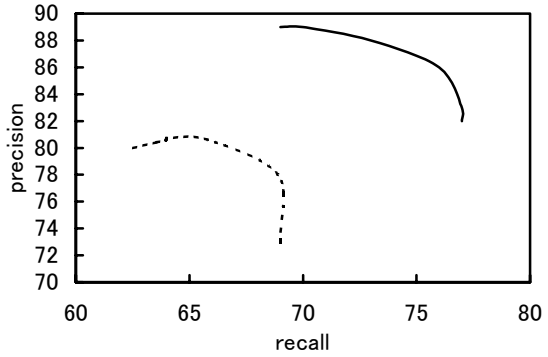

Figure 14: Precision-Recall Graph
A dotted line is P-R when we regards near-corrects as wrong.

Above precision and recall are defined by the number of correspondences. We defined the coverage by the number of correct phrases and near-corrects correspondences.

$$\text{coverage} = \frac{\text{count (phrases in correct and near - correct *)}}{\text{count (phrases in sentence)}}$$

* We counted only correct phrases included in near-correct correspondences.

When threshold=3, the coverage is 51.9%. It is considered as the coverage of *Basic-correspondences*. By contrast, when threshold=0, the coverage is 68.1%. It is considered as the coverage when the system finds *extend-correspondences* the most actively.

In follow tables, samples of found correspondences are shown.

Table 4: correct examples of Basic-correspondences

English	Japanese
in particular	Toku ni
among major countries	Syuyou koku no
with end of Cold War	Reisen syuuketu to tomoni
in world market	Sekai sizyou ni okeru
by monthly instalments	Geppu barai de

Table 5: near-correct examples of Basic-correspondences

English	Japanese
(crossing) borders	Kokkyou wo
in that area	(Kyuu higasi doitu) tiku no
(like) home	Wagaya ni
into his suitcase	(Kare ha) su-tuke-su ni
transnational	Kuni wo [koete]

Table 6: correct examples of Extend-correspondences

English	Japanese	Score
is being pursued	Okonawarete iru	2.75
of G7 nations	Sensin 7 kakoku no	2.6
is vital	Zyuyou da	1.5
of TFP	Zenyouso seisanseiga	1.5
with priority- being given	Zyuuten to suru	0.33

Table 7: near-correct examples of Extend-correspondences

English	Japanese	Score
tree (become)	Sono ki ha	1.2
went [to bed]	Neru	1.0
is (also) important	Zyuuyou de aru	1.0
She (held)	Kanozyo ha	0.5
by companies	(Teimei suru) kigyou no	0.33

In tables, examples in English are written without articles. In near-correct examples, segments to be deleted to become correct patterns are embraced by "()". Segments to be added are embraced by "[]."

Most of *basic-correspondences* are compound nouns and they rarely include verbs. *Extend-correspondences* sometimes include verbs and some of them are abbreviations, such as G7, TFP and so on.

Conclusion

In this paper, we have proposed the system for finding phrasal correspondences. We think this method can be used only for parallel corpus. In comparable corpus, a statistical approach proved to be effective, however in parallel corpus, we think our approach is effective.

In our method, the accuracy is up to 80%. With manually correction of correspondences, we can reduce the cost of translation pattern accumulation.

As future directions, we have not used found correspondences in a translation-system yet. It is what remains to be done.

References

[1] Hideo Watanabe, Sadao Kurohashi, Eiji Aramaki, "Finding Structural Correspondences from Bilingual Parsed Corpus for Corpus-based Translation," Proc. of 18th Coling, pp. 906--912, 2000.

[2] Dekai Wu, "An Algorithm for Simultaneously Bracketing Parallel Texts by Aligning Words," ACL95

[3] Kaoru Yamamoto, Yuji Matsumoto, "Acquisition of Phrase-level Bilingual Correspondence using Dependency Structure," COLING-2000.

[4] Sadao Kurohashi, Makoto Nagao, "A Syntactic Analysis Method of Long Japanese Sentences based on the Detection of Conjunctive Structures," Computational Linguistics, Vol. 20, No, 4, 1994.

[5] McCord, C.M., Slot Grammars, "Computational Linguistics," Vol. 6, pp. 31-43, 1980.

[6] Nagao. M., "A Framework of a Mechanical Translation between Japanese and English by Analogy Principle," Elithorn, A. and Banerji, R. (eds.): Artificial and Human Intelligence, North-Holland, pp. 173-180, 1984.

[7] Sato, S. and Nagao, M., "Toward Memory-based Translation", Proc. of 13th Coling 90, Vol.3, pp. 247-252, 1990.

[8] Sumita, E., Iida, H., "Translating with Examples: A New Approach to Machine Translation", Proc. of Info-Japan '90, 1990.

[9] Hideo Watanabe, "A Similarity-Driven Transfer System," Proc. of 14th Int. Conf. of Computational Linguistics '92, pp.770-776, 1992.

[10] Hitoshi Isahara and Masahiko Haruno, "Japanese-English aligned bilingual corpora", in Parallel Text Processing: Alignment and use of translation corpora. (Text, Speech and Language Technology, Vol. 13), p. 313-334, Kluwer Academic, 2000.