Robust Recursive-Division Method in the Example-Based Machine Translation

Tantely ANDRIAMANANKASINA[†], Kenji ARAKI[††], Yoshio MOMOUCHI[†], *and* Koji TOCHINAI[††],

SUMMARY Using the recursive-division as transfer method in the Example-Based Machine Translation (EBMT) has been proven very promising for less explored languages whose resources and reliable tools are hardly available. However, the method uses an exact segment matching method, which sometimes makes the translation result in disorder since it hardly produces long match to reflect the structure of the sentence to translate. The same reason makes the method unable to select perfectly the appropriate choice for words having no correspondents. To solve these problems, we present in this paper a robust transfer method, which uses not only the exact segment matching method but also the pure POS tag matching one. The POS tag matching method is assumed to extract sentences reflecting the structure of the sentence to be translated. The proposed method process the results of each matching method independently. It is basically recursive-division, with an intelligent selection and application of the example to be applied at each recursion. It is designed for languages where the word-level aligned parallel text is the only resource available. Implementation is done in a French-Japanese machine translation system, and spoken language text are used as examples. Despite a small number of translation examples, the method produces high degree accuracy, and its superiority over the single exact segment matching method is confirmed.

key words: example-based machine translation, transfer method, less explored languages, recursive-division method

1. Introduction

Example-Based Machine Translation (EBMT) [1] has been proposed to avoid the hard and time-consuming task of rule and dictionary maintenance in the traditional Rule-Based Machine Translation (RBMT) [2]. The idea of EBMT is to achieve the translation by imitating similar translation examples, extracted from a set of translation examples. Rules are not required, the system learns from translation examples, instead. The large number of works on EBMT [3]–[5], and the presence of example-based translation systems [6], manifest the public interest of the example-based method as a translation method.

The non-availability of resources and reliable tools in less explored languages, led the authors to propose an EBMT model not depending on syntactic analyzers [7]. POS taggers are merely used with the bilingual parallel and word-level aligned corpus. The use of POS taggers is justified by the high degree accuracy of recent taggers [8], [9], and their portability to new languages. As for the word-level alignment, statistic-based methods have been proposed for large corpus [10]–[12], and a semi-automatic analogy-based method [13] can be applied for small corpus. The translation method is basically recursive-division, selecting a number of examples having a common segment with the sentence to translate, and splitting recursively the non-translated segment according to the selected example at each step, while translating the common segment. It is very promising as far as the result is concerned. However, the exact segment matching method, required by the method, is sometimes unable to extract sentences reflecting the structure of the sentence to be translated, because long matching segment cannot be found. This problem sometimes puts the translation result in disorder. In addition, since sentence structure is not discovered, the method sometimes does not produce the appropriate words having no correspondents, which play an important role in the sentence.

In the present paper, we try to solve these problems and present a robust recursive-division method, by introducing the POS tag matching method, which is assumed to bring a better view of sentence structure. Observation of POS tag is assumed to produce long matching segment. The longer the matching segment is, the better the structure of the input sentence[†] is represented. The introduction of the POS tag matching method itself is not new, but we propose a way to combine its results with ones of the exact matching method at the recursive-division transfer process, not at the matching process. The exact matching method is still needed, not only to produce word translation, but also to be used like the POS matching method when good results, as far as sentence structure is concerned, are outputted. The method intelligently selects whether a result from the POS tag matching or one from the exact matching will be applied at each recursion. That selection is based on word frequency, and on the similarity score. The use of word frequency is justified by the possibility to find long matching sentence, even though exact matching method is considered, for high-frequency segment. As for the similarity score, the local structure of the sentence to match is assured by

[†] High-Tech Research Center, Hokkai-Gakuen University, S-26 W-11, Sapporo, 064-0926 Japan.

[††] Graduate School of Engineering, Hokkaido University, N-13 W-8, Sapporo, 060-8628 Japan.

[†] or the sentence to be translated

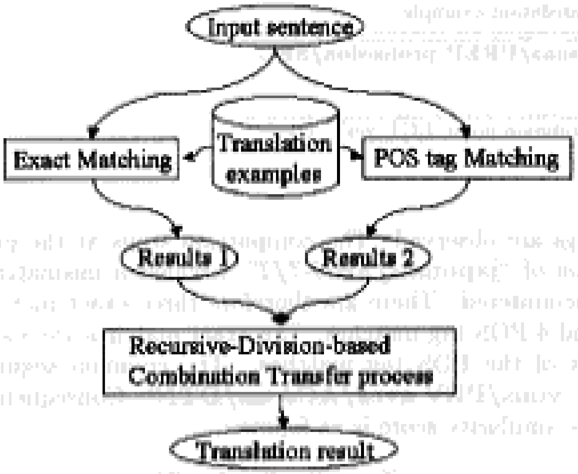

Fig. 1 Overview of the translation system

the presence of a relatively long contiguous exact segment match. The method, in addition, presents a different way of application of a POS tag segment match during the transfer process, because a same way cannot be applied for both exact segment match and POS tag segment match.

The translation system is immediately presented and detailed step by step in the following sections. The second part of the paper describes the experiments, results and discussions.

2. Overview of the translation system

The flow of the translation is presented in Fig. 1. Each matching method, the exact matching method and the POS tag matching method, extracts a number of examples matching the input sentence, from the translation example base. These extracted examples are the input of the recursive-division-based transfer process, which outputs the translation result. The translation model where only a single matching method is considered, is presented in Fig. 2 to give the reader a better understanding of the difference. It is very important to note that the sentences, extracted by the exact matching method, are classified differently from the sentences extracted by the POS tag matching method. We emphasize that difference of classification because it makes the subsequent process special in terms of combining sentences while partially treating them differently.

The structure of a translation example will be presented in the following paragraph, before describing the matching methods and the transfer method.

3. Structure of a translation example

An entry in the translation examples is presented in Table 1. It is composed by the French sentence, its Japanese translation, and a map describing links between words in both sentences. A token is presented with the format "token/POS tag". For the tagging operation, INALF[†]' s EBTI tagging program was used for French sentences and CHASEN1.51 tagger [8] for Japanese sentences. EBTI is an adaptation of the Eric Brill Tagger [9] for French. There are 48 POS tags for French language, and 14 for Japanese. A link has the format "$Wf_1, Wf_2, .../Wj_1, Wj_2, ...$", where "$Wf_i$" are word positions in the French sentence and "Wj_i" word positions in the Japanese sentence. In the example of Table 1, "2/2" means that the French second token "**suis**[††] (be)" corresponds to the Japanese second token "*desu*". By the same way, "3,4/1" means that "**sans profession** (jobless)" corresponds to "*mushoku*". Words or segments of words having no correspondent are not specified, as the case of "**je** (I)".

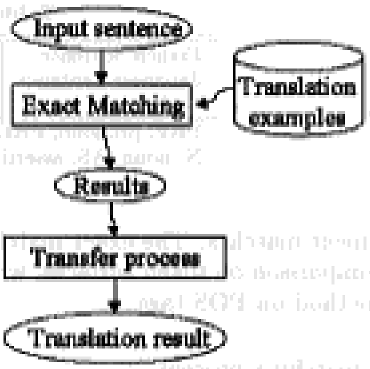

Fig. 2 Translation process for a single matching method

The translation method requires only links which are composed by contiguous words. For example, to align the phrase "**ne va pas** (do not go)" with "*iki masen*", the obvious way might be aligning "**ne pas** (do not)" with "*masen*" and "**va** (go)" with "*iki*". However, since "**ne pas**" is not a contiguous segment, only one link between "**ne va pas**" and "*ikimasen*" is considered. Although "**va**" alone cannot be translated, a contiguous segment map is obtained. This requirement does not raise problems since non-contiguous segment map can always be modified to a contiguous one, by combining segments.

4. Extraction of translation examples

The first step in the translation process is an extraction of sentences whose source sentences match the sentences to be translated. There are two different kinds of extraction, one corresponding to the exact matching method and another to the POS tag matching method. Both methods have a similarity that they search con-

[†] Institut National de la langue française

[††] French and Japanese words are represented with boldface and italic characters respectively.

Table 1 Structure of a translation example

French Sentence	**Je/PRV suis/ECJ sans/PREP profession/SBC**
Japanese Sentence	*mushoku/N desu*/AS
Links	2/2 3,4/1 5/6

PRV: pronoun, PREP: preposition, SBC: common noun, ECJ: verb "être"
N: noun, AS: assertive

tiguous segment matches. The exact matching method bases the comparison on token surfaces, while POS tag matching method on POS tags.

4.1 Exact matching process

Sentences having a common segment with the input sentence will be the target of the extraction. To cover the sentence to be translated, a number of examples are extracted. The matching algorithm is as follows.

1. For each token of the input sentence or the sentence to be translated, search a same token in the source sentence.
2. If found, from that position, start a forward and backward token comparison. The comparison starts with exact matches. It continues with POS tag matches when a POS tag match or a non-contiguous match is encountered, and stops when a mismatch or the head or end of sentence is encountered.

To select the best matching sentences, the following similarity score is used:

$$SC = \alpha \times NE + NP \qquad (1)$$

where SC is the similarity score, NE the number of exact matches, NP the number of POS tag matches. α is set to 10 to make the presence of exact matches being considered first, compared to the POS tag matches.

One sentence is selected for each token of the input sentence. It is the sentence having that token as part of the common segment and having the highest value of similarity score.

This algorithm, since it always starts the search from an exact match, has a subsidiary advantage that processing time can be reduced considerably by indexing the corpus on each token.

An illustration is presented in Fig. 3. Sentence 1 and 2 mean "Do you have a Japanese newspaper ?" and "Do you have an ashtray ?" respectively. For example, consider the token "**avez/ACJ**" as the search start token, an exact match is detected at the second position in both sentences. A backward comparison produces one exact match, "**vous/PRV-vous/PRV**", and a forward one yields one exact match, "**un/DTN-un/DTN**". The following match, "**journal/SBC-cendrier/SBC**", is not an exact match, but a POS tag match. Therefore, from that position, only POS tags are observed. The comparison stops at the position of "**japonais/SBC-?/?**" because a mismatch is encountered. There are therefore three exact matches and 4 POS tag matches. The exact matches are a subset of the POS tag matches. The common segment is "**vous/PRV avez/ACJ un/DTN**". Consequently, the similarity score is as follows:

$$SC = 10 \times 3 + 4 = 34$$

It is very important to note that during the search of similar sentences, one translation example is extracted for each token of the input sentence. However, since a same sentence may be extracted for multiple consecutive tokens and exact match may not be discovered for unregistered words, the number of extracted examples is usually fewer than the number of tokens.

4.2 POS tag matching process

The goal of the introduction of the POS tag matching is to capture the structure of the sentence to be translated. Exact segment matching method sometimes does not produce sentences having the same structure as the input sentence. POS-tag-based comparison is assumed to produce such results, since long match can be discovered. The algorithm for the POS tag matching method is similar to the exact matching method, but the comparison is based on POS tag, not on token surfaces. Therefore, the algorithm has a slight modification, as follows:

1. For each token of the sentence to be translated, search a token having a same POS tag in the source sentence.
2. If found, from that position, start a forward and backward token comparison. The comparison considers only token' s POS tag, and stops when a mismatch or the head or end of sentence is encountered

The similarity score is also modified as follows:

$$SC = \beta \times NP + NE \qquad (2)$$

where SC is the similarity score, NE the number of exact matches, NP the number of POS tag matches and β is set to 10. Normally NP would be enough to compute the similarity score, since only POS tags are considered. However NE is introduced to make the difference when NP is equal. This is also the reason of

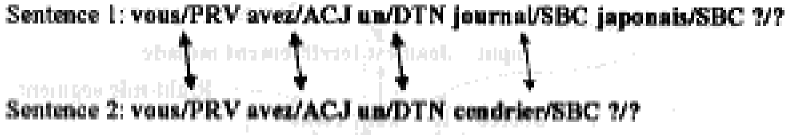
Sentence 1: vous/PRV avez/ACJ un/DTN journal/SBC japonais/SBC ?/?
Sentence 2: vous/PRV avez/ACJ un/DTN cendrier/SBC ?/?

Fig. 3 Illustration of the matching

the introduction of β. High value of β means that POS tag match segment will be considered first.

As an illustration, the example in Fig. 3 has four POS tag matches "**vous/PRV-vous/PRV** ", "**avez/ACJ-avez/ACJ**", "**un/DTN-un/DTN**" and "**journal/SBC-cendrier/SBC**", where three of them are exact matches. Therefore, the similarity score will be as follows:

$$SC = 10 \times 4 + 3 = 43$$

5. Recursive-division transfer process

To obtain the translation result, the sentence to be translated is splitted successively, translating a sub-segment at each step. At each step, an example is selected among the previously extracted examples, and the splitting process is performed according to the structure of that example. This splitting process will be described in the following paragraphs. The case where a translation example from the exact matching method is applied, is separated from the case of a sentence from the POS tag matching method. The decisions on whether an exact matching result or a POS tag matching one should be applied at a given step, and on which one should be applied among the extracted examples in that category, are the important issues in the method and will be presented in the last paragraph.

5.1 Application of an exact matching result

A simple illustration of the recursive-division method for the exact matching result is presented in Fig. 4. The input sentence and the source sentence mean "Jean is seriously ill" and "He is rich" respectively. The input sentence will be basically splitted in three segments: a common segment, a left-side segment and a right-side segment. Here, it is divided at the position of the word "est (be)", which is a common segment for both sentences. In the source sentence of the selected example, the segment "**Il** (He)" is located on the left side of the common segment and "**riche** (rich)" on its right side. If one observes their correspondents in the target sentence, the structure "(left side) *ha* (right side) **desu**" of the target sentence will be discovered. This structure will be applied to the input sentence, and finally it can be rewritten like "**Jean** *ha* **terriblement malade** *desu*". Using other examples, the same process will be applied again and again to non-translated segments, while they exist. In other words, the recursion is the application of the same process to the result of its previous execution. Here, there are two non-translated segments "**Jean**" and "**terriblement malade**". "**Jean**" is a single word segment and can be translated without division. On the other hand, "**terriblement malade**" can again be divided into "**terriblement**" and "**malade**" by the same process, using another example, or be translated directly if it appears somewhere in the corpus. Whether they need a division or not, appropriate examples must be selected and applied to translate them.

The method requires that the translation of the common segment must also be a single contiguous segment. A part of the common segment is only considered at one step if that condition is not satisfied. As for the left-side and right-side segments, their correspondents may be splitted. In these cases, their positions are determined by the position of the nearest sub-segment to the correspondent of the common segment.

When the exact segment match appears at the beginning of the source sentence, the position of the left-side segment cannot be determined since it is not available. Such is also the case of a right-side segment for a match at the end of sentence, or any segment having no correspondent on which its position should be predicted. For these cases, results from the POS tag matching method are observed to determine their position. If they still could not be solved by the POS tag matching results, left-side segment will be put at the left and right-side segment at the right. These cases rarely happen, but a corresponding process must be provided.

The recursive-division method has two important advantages. First, utilization of syntactic analyzer is unnecessary because sentences can be translated without understanding their syntactic structures. Second, since any sentence can always be divided, the method is able to translate long sentences. If non-translated segments can be divided at the right position at every step, the correct translation result will be reached.

5.2 Application of a POS tag matching result

The algorithm for the case where a POS tag segment match is applied, has a similarity with the case of an exact match. The difference is on keeping the common segment untranslated. As an illustration, for the above example, if the same translation example was

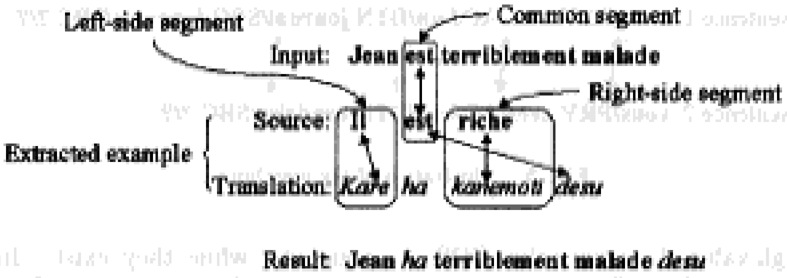
Left-side segment
Common segment
Input: Jean est terriblement malade
Right-side segment
Extracted example
Source: Il est riche
Translation: Kare ha kanemoti desu
Result: Jean ha terriblement malade desu

Fig. 4 Illustration of the recursive-division method

extracted for an input sentence like "**je suis terriblement malade** (I am seriously ill)", "**suis**" and "**est**" have a same POS tag but are different, the result will therefore be "**je** *ha* **terriblement malade suis**", keeping "**suis**" untranslated. Then after that, the result of the exact matching method, used as a dictionary, is consulted to give the translation of "**suis**", which is also "*desu*". And finally, the result "**Jean** *ha* **terriblement malade** *desu*" is obtained.

It is very important to note here that the result keeps the POS tag matching result structure, with the necessary no-correspondent words like "*ha*" in the above example. If the POS tag matching method had selected the Japanese sentence "*kare ha kanemochi desu ka*", and "*ka*"[†] had no correspondents, "*ka*" is assumed to depend on "*desu*", which is the common segment, and the result would be "**je** *ha* **terriblement malade suis** *ka*", before becoming "**je** *ha* **terriblement malade** *desu ka*". Details on dealing with words having no correspondents will be described in the next paragraph.

The difficulty with the application of POS tag matching results is on the definition of the common segment. Considering the whole POS tag match segment as common segment sometimes helps. However, there are cases where its correspondent is not a contiguous segment. In addition, there is a risk of lost of information. As an illustration, in the Japanese language, "*watashi* (I)" can be omitted. Matching the sentence "**nous sommes malades** (we are ill)" with "**je suis professeur** (I am professor)", which may simply mean "*sensei desu* (am professor)" in Japanese, risks to loose the information "**nous** (we)", if "**nous sommes** (we are)" is taken as common segment, since "je (I)" has no correspondent. It is therefore preferred to always refer to the exact matching results, and use the same common segment. Referring to the exact match assures that a common segment will have its exact meaning. As far as exact match is concerned, "**nous sommes**" cannot be a common segment because their correspondents are splitted. The solution will be taking "**sommes** (are)" as common segment, and leaving "**nous**" as a left-side segment to be treated in a further step.

5.3 Dealing with words having no correspondents

In the above example, the word "*ha*" has no correspondent. However, since it is an element located in the middle of the right side and left side segments, it is kept. The absence or presence of these words sometimes modifies completely the translation result. Two cases, where segments having no correspondent are kept, are proposed.

1. They are located between the translation of the right side and one of the left side segments, as the case of "*ha*" of the above example. Since the common segment is located between the left-side and right-side segments, it is assumed that a segment located between the translation of the right side and one of the left side segments in the target sentence plays an important role when the common segment exist.
2. They are closely related to the translation segment of the common segment (prefixes, postfixes, particles), as the case of "*ka*" in the above assumption. This is very obvious because if they depend on the translation of the common segment, they should automatically come with it.

5.4 Ordering and selecting the examples to be applied

A series of translation examples are applied successively to reach the final translation result. The order of application of the extracted translation examples has to be considered carefully. Wrong order will produce a completely different final result. We propose three conditions to decide this priority order.

1. Top priority is given to examples having common segments dividing successfully the sentence without dispersion of each part: the left-side, the common, and the right-side segment. Punctuation, conjunctions and so forth generally fall into this category.
2. Next, examples having common segments contain-

[†] "*ka*" is a particle which marks the interrogative

ing a verb are considered. This is explained by the importance of verbs in recognizing the structure of the whole sentence.

3. For the rest, the similarity score will make the difference, on condition that examples having non-functional words like nouns, adverbs or adjectives as common segment will be the last to be considered.

As for the decision on whether the POS tag matching result or the exact matching result will be applied, at a given recursion step, we introduce the word frequency with the similarity score to determine the choice. The extracted example is assumed to be good enough to reflect the structure of the input sentence, if all the exact word matches are on high frequency words or the exact common segment is relatively long. Therefore, if the common segment contains a word whose frequency is below a threshold frequency f, and the length of the common segment is lower than a given length l, the result from the POS tag matching will be applied. Otherwise, the result from the exact matching is directly applied.

6. Experiments and results

The initial bilingual corpus was composed by 2,500 examples. Sentences were taken from French-Japanese conversation books [14], [15]. The average sentence lengths are 7.74 tokens for Japanese and 7.84 for French. New 469 French sentences, taken from the same sources, are entered one by one into the system to be translated.

After a series of preliminary tests, the frequency threshold f is set to 0.1. That extracts 117 high frequency words among 2,830 different French words in the corpus. The exact match segment length threshold l, which determines whether the sentence from the exact matching method or one from the POS tag matching method will be applied, is set to 2. It means that if the exact matching segment is composed by more than 2 tokens, the sentence from the exact matching method is applied.

To be able to compare the results with the case where only a single matching method is considered, two experiments are performed, one for the case of the robust recursive-division method, which combines two matching methods, and the other for the case of a single exact matching method. Since unregistered words exist and a dictionary is not used, French words sometimes remain within the translation result. Evaluation of such results by sight is very difficult. We focused on segment position and consider the translation as correct if it has the same structure as the correct translation and all segments are put at their right position.

As far as translation accuracy is concerned, among 469 results, 325 (69.2%) are judged accurate, in contrast to 290 (61.8%) for the single matching method. Sample results showing the superiority of the method are selected from the output and presented in Table 2.Observation of the failures shows that 127 sentences, or 70.9% of the failures, suffer a disorder of sentence and/or an inappropriateness of words having no correspondents. The rests are almost from an inappropriateness of the selected words or expressions themselves, because the corpus is not rich enough to contain appropriate expressions for different situations.

7. Discussions

French words still remain non-translated in some results, but the improvement of 7.4 points compared to the single matching method, and the overall translation accuracy of 69.2%, despite the small size of the corpus, are considered as good results. Those confirm the effectiveness of the combination method. The sample results, given in Table 2, shows the effectiveness of the method in the selection of words having no correspondents, compared to the single matching method, as the case of the word "*ga*" in the first sentence, or "*yori*" of the third sentence. In addition, improvements in sentence structures, as the case of the second and fifth sentences, and word orders, as the case of the position of "*ashita*" in the fourth sentence, are clearly manifested. In short, the POS tag matching method produced sentences, which have improved the structure of the translation results.

Multiplication of the number of translation examples is the obvious way to reduce failures. A larger corpus will produce better sentences, in terms of structure and expression, at the output of the matching methods. The final output will be improved accordingly. There are, however, cases of failures which need to be considered particularly.

First, the duplication of words having no correspondents: for example, an attempt to translate "Jean" in the halfway result "**Jean** *ha* **terriblement malade** *desu*", may produce "*Jan san ha ha* **terriblement malade** *desu*", because a new particle "*ha*" could follow the corresponding word in the extracted sentence. Insertion of these words having no correspondents is very helpful when they were not inserted at a previous step, but falsify the result when such is not the case. These failures can be corrected by considering the words having no correspondents only when they depend on the common segment, and discard them when they are located in the middle of the left-side and right-side segments. However, that consideration would lose important information, especially when the common segment has no correspondents, because for such case, the segment between the left-side segment and right-side segment plays an important role. As an illustration, for the extracted example "**je ne sais pas s' il est riche ou non** (I don' t know if he is rich or not)", which

Table 2 Sample comparative results

Input sentence	**De la viande, s' il vous plait.**
Meaning	Meat, please.
Result (single matching)	*Niku ga onegai shimasu.*
Result (combination)	*Niku wo onegai shimasu.*
Input sentence	**C' est pres d' ici ?**
Meaning	Is it near here ?
Result (single matching)	*Kono chikaku ni desuka.*
Result (combination)	*Koko kara chikai desuka.*
Input sentence	**Les homes sont plus nombreux que les femmes.**
Meaning	Men are more numerous than women.
Result (single matching)	*Danshi ga joshi no yori mo* (**nombreux**) *desu.*
Result (combination)	*Danshi ga joshi yori* (**nombreux**) *desu.*
Input sentence	**Je voudrais prendre le petit dejeuner dans ma chambre demain.**
Meaning	I want to have the breakfast in my room tomorrow.
Result (single matching)	*Choushoku ha heya de toritai no desu ga ashita.*
Result (combination)	*Choushoku ha ashita heya de toritai no desu ga.*
Input sentence	**La porte est ouverte.**
Meaning	The door is open.
Result (single matching)	*Doa ha desu ka.*
Result (combination)	*Doa ha* (**ouverte**) *you desu ne.*

means "*kare ha kanemochi ka dou ka wakarimasen*" in Japanese, "s'" and "*ka dou ka*" do not have a specific correspondent. Splitting the sentence at the position of "s'" would give "*kare ha kanemochi wakarimasen*" if "*ka dou ka*", which is located in the middle of the left-side and the right-side segments, is not considered. The high degree accuracy of the translation shows that the described method is still the best way to handle these situations.

Second, the disorder of the translation result: these failures come from the wrong order of the application of the examples, or the inappropriateness of the position of the left-side or right-side segment, when applied to the sentence to be translated. A further study on these cases is necessary, with a large corpus, however, we emphasize here the 7.4 points improvement of the present method, compared to the single matching method, and the high degree accuracy of the translation.

8. Conclusions

A robust recursive-division-based EBMT transfer method, which combines results from a POS tag matching method with ones from an exact matching method, has been presented. The method is proposed to solve disorder of translation results, as well as the inappropriateness of words having no correspondents, in the case where only a single exact matching method is considered. The recursive-division can be applied for less explored languages since utilization of syntactic analyzer is unnecessary. POS taggers and the parallel word-level aligned bilingual corpus are the only resource required.

Experiments confirmed the effectiveness of the method, as a translation method, and its superiority over the single exact matching method. Despite the small size of the corpus and the presence of sentence not following grammar rules in the spoken languages, high accuracy rate of 69.2% and 7.4 points of accuracy improvement, compared to the single matching method, are earned.

An experiment with large corpus is necessary, not only to simulate a real full translation system, but especially to adjust the selection of words having no correspondents, and the order of the application of examples.

Acknowledgment

We express our gratitude to the High-Tech Research Center of the Hokkai-Gakuen University for fully supporting this research. We extend our thanks to the "Institut National de la Langue Française" for providing the EBTI tagging program for the French language.

References

[1] M. Nagao, "A Framework of a Mechanical Translation Between Japanese and English by Analogy Principle," Artificial and Human Intelligence, pp.173–180, 1984.

[2] J. Hutchins and H. Somers, "An Introduction to Machine Translation," Academic Press, London, 1992.

[3] S. Sato and M. Nagao, "Towards Memory-Based Machine Translation," Proceedings of COLING-90, pp.247-252, 1990.

[4] H. Kitano, "A Comprehensive and Practical Model of Memory-Based Machine Translation," Proceedings of IJCAI-93, pp.1276–1282, 1993.

[5] H. Watanabe, "A Model of Bi-Directional Transfer Mechanism Using Rule Combinations," Journal of Machine Translation, vol.10, no.4, pp.269–291, 1995.

[6] S. Nirenburg, "The Pangaloss Mark III Machine Translation System," A Joint Technical Report by NMSU CRL, USC ISI and CMU CMT, April 1995.

[7] T. Andriamanankasina, K. Araki and K. Tochinai, "Example-Based Machine Translation of Part-Of-Speech Tagged Sentences By Recursive Division," Proceedings of MT-SUMMIT VII, pp.509–517, Singapore, September 1999.

[8] T. Yamashita, "ChaSen Technical Report," Nara Advanced

Institute of Science and Technology, 1996.

[9] E. Brill E, "Some advances in rule-based part of speech tagging," Proceedings of the Twelfth National Conference on Artificial Intelligence, Seattle, 1994.

[10] P.F. Brown, S.A.D. Pietra, V.J.D. Pietra and R.L. Mercer, "The Mathematics of Statistical Machine Translation: Parameter estimation," Computational Linguistics, vol.19, no.2, pp.263–311, 1993.

[11] D. Melamed, "A Word-to-Word Model of Translational Equivalence," Proceedings of the 35th Conference of the Association for Computational Linguistics, pp.490–497, 1997.

[12] M. Kitamura and Y. Matsumoto, "Automatic Extraction of Translation Patterns in Parallel Corpora," Transactions of the IPSJ, vol.38, no.4, pp.727–736, 1997.

[13] T. Andriamanankasina, K. Araki and K. Tochinai, "Example-Based Sub-Sentential Alignment Method by Analogy," Transaction of the IPSJ, vol.40, no.7, July 1999.

[14] S. Meguro, "Manuel de Conversation Française," Hakusuisha, 1987.

[15] F. Sato, "Locutions de base," Hakusuisha, 1990.