Post-Ordering by Parsing with ITG for Japanese-English Statistical Machine Translation

ISAO GOTO, National Institute of Information and Communications Technology and Kyoto University
MASAO UTIYAMA and EIICHIRO SUMITA, National Institute of Information and Communications Technology

Word reordering is a difficult task for translation between languages with widely different word orders, such as Japanese and English. A previously proposed post-ordering method for Japanese-to-English translation first translates a Japanese sentence into a sequence of English words in a word order similar to that of Japanese, then reorders the sequence into an English word order. We employed this post-ordering framework and improved upon its reordering method. The existing post-ordering method reorders the sequence of English words via SMT, whereas our method reorders the sequence by (1) parsing the sequence using ITG to obtain syntactic structures which are similar to Japanese syntactic structures, and (2) transferring the obtained syntactic structures into English syntactic structures according to the ITG. The experiments using Japanese-to-English patent translation demonstrated the effectiveness of our method and showed that both the RIBES and BLEU scores were improved over compared methods.

Categories and Subject Descriptors: I.2.7 [**Artificial Intelligence**]: Natural Language Processing—*Machine translation*

General Terms: Design, Algorithms, Experimentation

Additional Key Words and Phrases: Machine translation, post-ordering, parsing, inversion transduction grammar

ACM Reference Format:
Goto, I., Utiyama, M., and Sumita, E. 2013. Post-ordering by parsing with ITG for Japanese-English statistical machine translation. *ACM Trans. Asian Lang. Inform. Process.* 12, 4, Article 17 (October 2013), 22 pages.
DOI: http://dx.doi.org/10.1145/2518100

1. INTRODUCTION

Reordering target language words into an appropriate word order in the target language is one of the most difficult problems for statistical machine translation (SMT), in particular when translating between languages with widely different word orders, such as Japanese and English. In order to handle this problem, a number of reordering methods have been proposed in statistical machine translation research. Those methods can be classified into the following three types.

— *Type-1: Conducting target word selection and reordering jointly*. These methods include phrase-based SMT [Koehn et al. 2003], hierarchical phrase-based SMT

This article is based on Goto et al. [2012] in *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*.
Authors' address: I. Goto (corresponding author), M. Utiyama, and E. Sumita, Multilingual Translation Laboratory, National Institute of Information and Communications Technology, 3-5 Hikaridai, Keihanna Science City, Kyoto, 619-0289, Japan; emails: {igoto, mutiyama, eiichiro.sumita}@nict.go.jp.

DOI: http://dx.doi.org/10.1145/2518100

[Chiang 2007], and syntax-based SMT [Chiang 2010; Ding and Palmer 2005; Galley et al. 2004; Liu et al. 2006; Quirk et al. 2005; Yamada and Knight 2002].

— *Type-2: Pre-ordering [Isozaki et al. 2010b; Xia and McCord 2004].* First, these methods reorder the source language sentence into a target language word order. Then, they translate the reordered source word sequence using SMT methods.

— *Type-3: Post-ordering [Matusov et al. 2005; Sudoh et al. 2011b].* First, these methods translate the source sentence almost monotonously into a target language word sequence. Then, they reorder the target language word sequence into a target language word order. In other words, the order of the word reordering and selection processes in post-ordering are the reverse of those in pre-ordering.

Sudoh et al. [2011b] indicated that type-3 performed better than existing type-1 methods for Japanese-to-English translations. As for type-2, different translation directions have different reordering problems, even if the language pair is the same, because the performance of pre-ordering methods using a parser depends on the difficulty of estimating the target language word order and the parse accuracy for the source language. In fact, one type-2 method for English-to-Japanese translation obtained a large gain, but another type-2 method for Japanese-to-English translation could not obtain a large gain [Goto et al. 2011; Sudoh et al. 2011a]. The reason for the high performance of the English-to-Japanese translation is that estimating a Japanese word order based on English is not difficult. This is because Japanese-like word order can be obtained by simply moving an English headword to the end of its syntactic siblings, since Japanese is a typical head-final language [Isozaki et al. 2010b]. On the other hand, English is not a head-final language, which makes estimating English word order more difficult than estimating Japanese word order. Namely, pre-ordering is effective for translating into a target language where estimating word order is not difficult. In contrast, type-3 post-ordering is thought to be effective for translating from a source language where estimating word order is not difficult. The reason is as follows: a post-ordering model is built using a parallel corpus consisting of target language sentences and corresponding sentences containing the same words, but in the source language word order. The sentences in the source language word order are produced by changing the target language word order into the source language word order. This change is reliable when estimating source language word order is not difficult.

We employ the post-ordering framework for Japanese-English translation. The post-ordering method consists of a two-step process: (1) almost monotonously translating a Japanese sentence into an English word sequence in a Japanese-like word order; (2) reordering the English word sequence in a Japanese-like word order into an English word order. The first process can be conducted by traditional phrase-based SMT methods. For the second process, Sudoh et al. [2011b] proposed a method using phrase-based SMT for the English word reordering.

In this article, we propose a reordering method based on parsing with inversion transduction grammar (ITG) [Wu 1997] for the post-ordering framework. The focus of this article is the second process of the post-ordering framework, which reorders an English word sequence in a Japanese-like word order into an English word order. Our method uses syntactic structures, which are essential for improving the target word order in translating long sentences between Japanese (a subject-object-verb (SOV) language) and English (an SVO language). Our reordering model parses an English word sequence in a Japanese-like word order using ITG to obtain derivations of Japanese-like syntactic structures, then reorders by transferring the Japanese-like syntactic structures into English structures based on the ITG. Experiments found that our reordering model improved translation quality as measured by both RIBES [Isozaki et al. 2010a] and BLEU [Papineni et al. 2002].

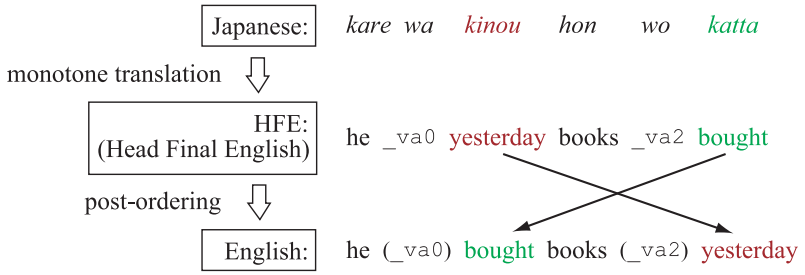

Fig. 1. Post-ordering framework.

The rest of this article is organized as follows. Section 2 shows the post-ordering framework and a previous method; Section 3 describes the proposed reordering model for post-ordering; Section 4 explains the proposed method in detail; Section 5 gives and discusses the experiment results; Section 6 shows related work; and Section 7 concludes.

2. POST-ORDERING FOR SMT

In this article, we take a post-ordering approach [Sudoh et al. 2011b] for Japanese-English translation, which performs translation as a two-step process of word selection and reordering. The translation flow for the post-ordering method is shown in Figure 1, where "HFE" is an abbreviation of "Head Final English", which is English words in a Japanese-like structure.[1] The two-step process is as follows.

(1) Translating first almost monotonously transforms Japanese into HFE, which is an English word sequence in almost the same word order as Japanese, using a method such as phrase-based SMT [Koehn et al. 2003], which can produce accurate translations when only local reordering is required.
(2) Reordering then transforms the HFE into English.

In the post-ordering framework, the reordering model that reorders HFE into English is important. Sudoh et al. [2011b] proposed a reordering model that consisted of an HFE-English phrase-based SMT, which reordered by translating an HFE sentence into an English sentence. In general, syntactic structures are important for reordering in translating between languages with widely different word orders. However, the reordering model consisted of phrase-based SMT for post-ordering cannot fully use syntactic structures. In contrast, our reordering model for post-ordering can utilize these useful syntactic structures, which gives our reordering model an advantage.

In order to train a Japanese-HFE SMT model and an HFE-English reordering model, a Japanese-HFE parallel corpus and an HFE-English parallel corpus are needed. These corpora can be constructed by parsing the English sentences in a Japanese-English parallel corpus and applying the head-finalization rules [Isozaki et al. 2010b] to the parsed English sentences. The head-finalization rules change English sentences into HFE sentences, which is in Japanese-like word orders. Then a Japanese-HFE-English parallel corpus is built.

Here, we explain how the head-finalization rules change English into HFE. Japanese is a typical head-final language, where a syntactic head word comes after nonhead (dependent) words. The head-finalization rules move each syntactic head to the end of its siblings. English sentences are parsed by a parser, Enju [Miyao and Tsujii 2008], which outputs syntactic heads. Consequently, the parsed English sentences can be

[1]The explanations of pseudo-particles (_va0 and _va2) and other details of HFE is given in Section 4.4.

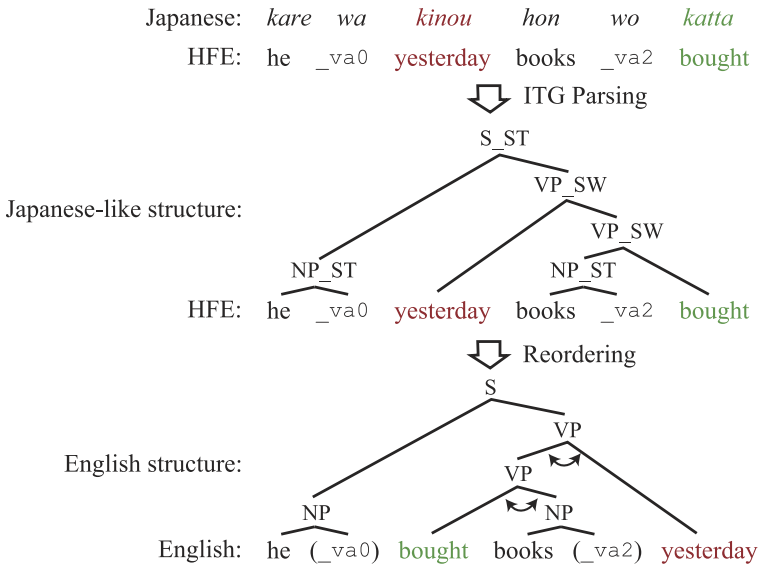

Fig. 2. Example of post-ordering by parsing.

reordered into Japanese-like word ordered HFE sentences using the head-finalization rules.

Training for the post-ordering method is conducted via the following steps: first, the English sentences in a Japanese-English parallel corpus are converted into HFE sentences using the head-finalization rules. Next, a monotone phrase-based Japanese-HFE SMT model is built using the Japanese-HFE parallel corpus whose HFE sentences were converted from English sentences. Finally, an HFE-to-English word reordering model is built using the HFE-English parallel corpus.

3. POST-ORDERING MODEL

In this section, we describe our reordering model for post-ordering, which we concentrate on in this article. We explain how the reordering model reorders HFE into English and how to train the reordering model.

3.1. Reordering by the ITG Parsing Model

The proposed reordering model for post-ordering, which we have called the *ITG parsing model*, is based on two fundamental frameworks: (i) parsing using probabilistic context free grammar (PCFG) and (ii) the inversion transduction grammar (ITG) [Wu 1997]. ITG between HFE and English is used as the PCFG for parsing. In this article, parsing using ITG is called *ITG parsing*.

We assume that there is an underlying HFE binary tree derivation that produces English word order. The reordering process by the ITG parsing model is shown in Figure 2. An HFE sentence is parsed using ITG to obtain an HFE binary tree derivation, which is similar to the syntactic tree structure of the input Japanese sentence. Each nonterminal node that has two child nodes is augmented by either an "_ST" (indicating "straight") suffix or an "_SW" (indicating "swap/inversion") suffix. The English word order is determined by the binary tree derivation and the suffixes of the nonterminal nodes. We swap the child nodes of the nodes augmented with the "_SW" suffix in the binary tree derivation in order to produce an English sentence.

3.2. Training the ITG Parsing Model

In order to train the ITG parsing model, the structures of the HFE sentences with "_ST" and "_SW" suffixes are used as the training data. The training data can be obtained from the corresponding English sentences as follows.

First, each English sentence in the training Japanese-English parallel corpus is parsed into a binary tree structure by applying the Enju parser. Then, for each nonterminal node in the English binary tree structure, the two child nodes of each node are swapped if the first child is the head node (see [Isozaki et al. 2010b] for more information on head-finalization rules). At the same time, these nodes with swapped child nodes are annotated with "_SW". When the two child nodes of each node are not swapped, these nodes are annotated with "_ST". A node with only one child is not annotated with "_ST" or "_SW". The result is an HFE sentence in a binary tree structure augmented with straight or swap/inversion suffixes.

Binary tree structures can be learnable by using an off-the-shelf PCFG learning algorithm. Therefore, HFE binary tree structures can also be learnable. HFE binary tree structures augmented with the straight or swap/inversion suffixes can be regard as derivations of ITG [Wu 1997] between HFE and English. Therefore, a parsing model learned from the HFE binary tree structures using a PCFG learning algorithm is an ITG model between HFE and English.

In this article, we used the state split probabilistic CFG [Petrov et al. 2006] for learning the ITG model. The learned ITG model for parsing is the *ITG parsing model*. The HFE sentences can be parsed by using the ITG parsing model. Then the derivations of the HFE structures can be converted into their corresponding English structures by swapping the child nodes of the nodes with the "_SW" suffix. Note that this ITG parsing model jointly learns how to parse and swap the HFE sentences.

4. DETAILED EXPLANATION OF THE TRANSLATION METHOD

This section explains the proposed translation method, which is based on the post-ordering framework using the ITG parsing model, in detail.

4.1. Derivation of Two-Step Translation

Machine translation is formulated as a problem of finding the most likely target sentence E given a source sentence F.

$$\hat{E} = \arg\max_{E} P(E|F).$$

In the post-ordering framework, we divide the translation process into two processes using an HFE sentence M.

$$\begin{aligned}\hat{E} &= \arg\max_{E} \sum_{M} P(E, M|F) \\ &= \arg\max_{E} \sum_{M} P(E|M, F)P(M|F).\end{aligned}$$

The summation is approximated by maximization to reduce computational costs and weighting parameters λ_x (x is r, s, or others) are introduced to be tunable by weighting each model in the same manner as a log-linear model.

$$\begin{aligned}\hat{E}, \hat{M} &\approx \arg\max_{E,M} P(E|M, F)P(M|F) \\ &\approx \arg\max_{E,M} P(E|M, F)^{\lambda_r} P(M|F)^{\lambda_s}. \end{aligned} \quad (1)$$

$P(M|F)$ in Equation (1) is the probability of translation from a Japanese sentence F into an HFE sentence M. We use the SMT score S of a log-linear SMT model as the logarithm of $P(M|F)^{\lambda_s}$, that is, $\lambda_s log(P(M|F)) = S$. For the experiment, we used the Moses SMT score [Koehn et al. 2007] from F to M translation as S $(= \lambda_s log(P(M|F)))$. When the Moses SMT score is calculated, feature values, such as a language model probability, are scaled by a set of weighting parameters. The set of weighting parameters are usually tuned by a tuning algorithm (e.g., minimum error rate training (MERT) [Och 2003]). λ_s approximately represents the scaling by the set of weighting parameters.

We compared two reordering models for estimating $P(E|M,F)^{\lambda_r}$ in Equation (1).

4.2. Translation Using Reordering Model 1

The first reordering model is independent of F given M, and we assume that an underlying HFE tree derivation T_M, which is augmented with "_SW" and "_ST", produces an English word order.

$$\begin{aligned}
\hat{E}, \hat{T}_M, \hat{M} &\approx \arg\max_{E,T_M,M} P(E, T_M|M)^{\lambda_r} P(M|F)^{\lambda_s} \\
&= \arg\max_{E,T_M,M} P(E|T_M, M)^{\lambda_r} P(T_M|M)^{\lambda_r} P(M|F)^{\lambda_s} \qquad (2) \\
&\approx \arg\max_{E,T_M,M} P(E|T_M, M)^{\lambda_{r_1}+\lambda_{r_2}} P(T_M|M)^{\lambda_{r_3}} P(M|F)^{\lambda_s} \\
&\approx \arg\max_{E,T_M,M} P(E)^{\lambda_{r_1}} P(E|T_M, M)^{\lambda_{r_2}} P(T_M|M)^{\lambda_{r_3}} P(M|F)^{\lambda_s}. \qquad (3)
\end{aligned}$$

We use the ITG parsing model as $P(T_M|M)$. That is, to obtain high probability T_M, we parse M using the ITG parsing model described in Section 3.1. Equation (2) is approximated by introducing independent weight parameters λ_{r_1}, λ_{r_2}, and λ_{r_3} instead of λ_r to be tunable by weighting each model in the same manner as a log-linear model; dividing $P(E|T_M, M)^{\lambda_r}$ into two models; and omitting conditions of one of the divided models. E is produced from T_M and M deterministically by swapping the child nodes of the nodes with the "_SW" suffix described in Section 3.1. This production process is expressed by $P(E|T_M, M)$. Thus, $P(E|T_M, M)^{\lambda_{r_2}}$ is 1 for E produced from T_M deterministically and is 0 for other E. $P(E)$ is the language model probability of an English sentence E.

Here, we explain why we introduce $P(E)$, which has fewer conditions than $P(E|T_M, M)$. (i) In general, actual models used for calculating probabilities are approximations of equations and not perfect. For example, an n-gram language model appropriately smoothed by a liner combination of an n-gram model and an (n−1)-gram model is usually better than a simple n-gram language model based on the maximum likelihood estimation by relative frequencies. (ii) When the architectures of the two models that calculate the probabilities of the same object are quite different, each model can capture different aspects. Therefore, the n-gram language model of $P(E)$ will remedy the deficiencies of the ITG parsing model of $P(T_M|M)$, which should evaluate generative probability of E because the word order of E is produced from T_M determinately.

4.3. Translation Using Reordering Model 2

The first reordering model (reordering model 1) is independent of F. If some noise is included in M when M is produced from F using SMT or if tree derivations of M are more ambiguous than tree structures of F, the tree structure of F will be useful in obtaining a tree derivation of M. This is because F is not a translation result, and a correct tree derivation of M is expected to be similar to a correct tree structure of F, since an HFE sentence is regarded as English words in a Japanese structure.

In this section, we introduce the second reordering model that uses a Japanese syntactic structure. The second reordering model uses the maximum probability Japanese syntactic structure T_F and the maximum probability word alignments A between F and M to obtain an underlying HFE tree derivation T_M, and we also assume that T_M produces the following English word order.

$$\begin{aligned}
\hat{E}, \hat{T}_M, \hat{M} &\approx \arg\max_{E,T_M,M} P(E, T_M, A, T_F|M, F)^{\lambda_r} P(M|F)^{\lambda_s} \\
&= \arg\max_{E,T_M,M} P(E|T_M, A, T_F, M, F)^{\lambda_r} P(T_M|A, T_F, M, F)^{\lambda_r} P(A|T_F, M, F)^{\lambda_r} \\
&\qquad \times P(T_F|M, F)^{\lambda_r} P(M|F)^{\lambda_s} \qquad (4) \\
&= \arg\max_{E,T_M,M} P(E|T_M, M)^{\lambda_r} P(T_M|A, T_F, M)^{\lambda_r} P(A|M, F)^{\lambda_r} P(T_F|F)^{\lambda_r} P(M|F)^{\lambda_s} \qquad (5) \\
&= \arg\max_{E,T_M,M} P(E|T_M, M)^{\lambda_r} P(T_M|A, T_F, M)^{\lambda_r} P(M|F)^{\lambda_s} \qquad (6) \\
&\approx \arg\max_{E,T_M,M} P(E|T_M, M)^{\lambda_{r_1}+\lambda_{r_2}} P(T_M|A, T_F, M)^{\lambda_{r_3}} P(M|F)^{\lambda_s} \\
&\approx \arg\max_{E,T_M,M} P(E)^{\lambda_{r_1}} P(E|T_M, M)^{\lambda_{r_2}} P(T_M|A, T_F, M)^{\lambda_{r_3}} P(M|F)^{\lambda_s}. \qquad (7)
\end{aligned}$$

In Equation (4), we assume that E is conditionally independent of A, T_F, and F given T_M and M; that T_M is conditionally independent of F given A, T_F, and M; that A is conditionally independent of T_F given M and F; and that T_F is conditionally independent of M given F. $P(T_F|F)$ in Equation (5) is constant given F.[2] $P(A|M, F)$ in Equation (5) is approximately assumed as a constant. Equation (6) is approximated by introducing independent weight parameters λ_{r_1}, λ_{r_2}, and λ_{r_3} instead of λ_r in the same manner as a log-linear model; dividing $P(E|T_M, M)^{\lambda_r}$ into two models; and omitting conditions of one of the divided models. We use the ITG parsing model with consideration of T_F as $P(T_M|A, T_F, M)$, that is, to obtain high probability T_M, we parse M by the ITG parsing model with consideration of T_F. $P(E|T_M, M)$ represents the deterministic production of E from T_M and M described in Section 3.1. $P(E|T_M, M)^{\lambda_{r_2}}$ is 1 for E produced from T_M deterministically and is 0 for other E.

What differs between Equation (3) of the previous reordering model 1 and Equation (7) of this reordering model 2 is that Equation (7) uses $P(T_M|A, T_F, M)$ instead of the $P(T_M|M)$ of Equation (3). We use the following simple method using a weighting parameter w ($0 < w < 1$), which is tuned using development data, as one implementation of $P(T_M|A, T_F, M) = P(T_M|A, T_F, M; w)$: a correct T_M is expected to be similar to a correct T_F, since an HFE sentence is regarded as English words in a Japanese structure. To reflect this expectation, we change the rule probabilities of the state split PCFG slightly, depending on T_M and T_F, using a weighting parameter w ($0 < w < 1$) as follows.

— If a subtree in T_M does not cross the word span of any subtree in T_F (Rule 1 in Case 1 in Figure 3), the rule probability p of the corresponding CFG rule instance is raised to p^w.
— If a subtree in T_M crosses the word span of any subtree in T_F (Rule 2 in Case 2 in Figure 3; in this case, the Rule 2 subtree word span 3 to 6 crosses the Japanese

[2]Note that in these equations, T_F and A are not the argument of the maximum, because we use the maximum probability Japanese syntactic structure as T_F and the maximum probability word alignments as A.

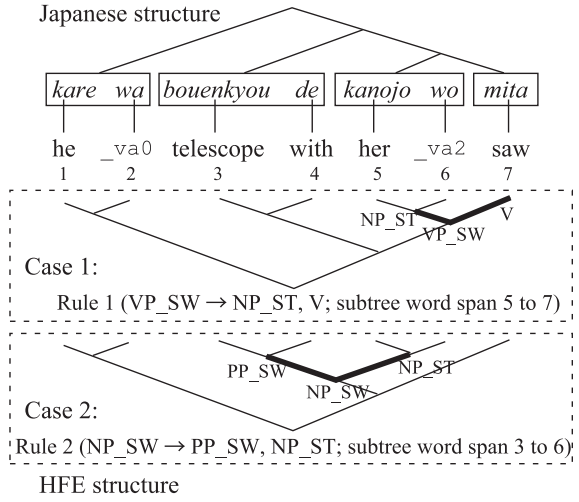

Fig. 3. Example of subtree spans.

subtree word span 5 to 7), the rule probability p of the corresponding CFG rule instance is reduced to p^{2-w}.

p^{2-w} is used to reduce the probability because p^{2-w} is thought to be a symmetric form of p^w, since when w is 1, both p^w and p^{2-w} are the same as p, and as w becomes smaller, the effects increase for both p^w and p^{2-w}. Note that the rule score for each application of the same rule can vary depending on the situation.

Although the resulting rule scores are ad hoc, this assists in making the analysis of T_M closer to T_F.

4.4. HFE

This section gives more details about HFE [Sudoh et al. 2011b]. In HFE sentences, the following hold.

(1) Each syntactic head is moved toward the end of its siblings except for coordination.
(2) Pseudo-particles are inserted after verb arguments: `_va0` (the subject of the sentence head), `_va1` (the subject of a verb), and `_va2` (the object of a verb).
(3) Articles (a, an, the) are dropped.

Although these were specified by Sudoh et al. [2011b], we attempt to explain the reasons for the specifications. The reason for (1) is that Japanese is a head-final language. The reasons for (2) and (3) are because translating is usually easier in SMT when words in a parallel sentence correspond one to one than when words correspond one to null. Specifications (2) and (3) try to reduce the one-to-null word correspondences. Japanese sentences contain particles that are case markers for subjects and objects, but English has no such corresponding words. The pseudo-particles in HFE correspond to these Japanse particles. On the flip side, Japanese does not contain articles, and thus they are dropped.

There is one point of difference between our HFE construction and that of Sudoh et al. [2011b]: in our method, plural nouns were left as plural instead of being

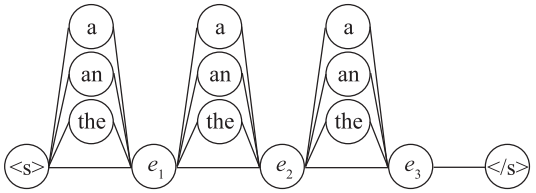

Fig. 4. Example of a lattice structure.

converted to singular, because our reordering model does not change words; it only reorders them.

4.5. Article Insertion

Applying our reordering model to an HFE sentence produces an English sentence that does not have articles but does have pseudo-particles. We removed the pseudoparticles from English sentences produced from HFE sentences before calculating the probabilities of $P(E)$ in Equations (3) and (7) because the language model $P(E)$ without pseudoparticles is simpler than that with pseudo-particles and is more robust than that with pseudoparticles, since E without pseudo-particles is not influenced by insertion errors from inserting pseudo-particles into training data. A language model $P(E)$ was trained from English sentences whose articles were dropped.

In order to output a genuine English sentence E' from E, articles must be inserted into E. A language model trained using genuine English sentences is used for this purpose. E' is obtained by

$$\hat{E}' = \arg\max_{E' \in S} P(E'),$$

where S is a set consisting of E with articles. We calculate the maximum probability word sequence through a dynamic programing technique for obtaining a genuine English sentence.

Articles are inserted by building a lattice structure which inserts one of the articles {a, an, the} or no article for each word e_i in $E = e_1 e_2 ... e_I$. Figure 4 shows the lattice structure in the case of $I = 3$. In Figure 4, <s> is a special word representing beginning of sentence, and </s> is a special word representing end of sentence. The maximum probability word sequence is calculated by applying the Viterbi algorithm for the lattice structure and an n-gram language model.

5. EXPERIMENT

We investigated the effectiveness of our method by comparing it with other methods for Japanese to English translation.

5.1. Setup

We used patent sentence data for the Japanese-to-English translation subtask from the NTCIR-9 [Goto et al. 2011] and NTCIR-8 [Fujii et al. 2010]. The training data and the development data for NTCIR-9 and NTCIR-8 are the same, but the test data is different. There were 2,000 test sentences for NTCIR-9 and 1,251 for NTCIR-8. There were approximately 3.18 million sentence pairs for the training data and 2,000 sentence pairs for the development data. XML entities included in the data were decoded to UTF-8 characters before use.

We used Enju [Miyao and Tsujii 2008] to parse the English side of the training data. Mecab[3] was used for the Japanese morphological analysis and Cabocha[4] for the Japanese dependency parsing. We adjusted the tokenization of alphanumeric characters and parentheses in Japanese to be the same as for the English. The translation model was trained using sentences of 64 words or less from the training data [Sudoh et al. 2011b]. Approximately 2.97 million sentence pairs were 64 words or less. We used 5-gram language models with modified Kneser-Ney discounting [Chen and Goodman 1998] using SRILM [Stolcke et al. 2011]. The language models were trained using all of the English sentences from the bilingual training data.

We used the Berkeley parser [Petrov et al. 2006], which is an implementation of the state split PCFG based parser, to train the ITG parsing model for HFE and to parse HFE. The ITG parsing model was trained using 0.5 million sentences randomly selected from training sentences of 40 words or less. We performed six split-merge iterations as the same iteration of the parsing model for English [Petrov et al. 2006]. We used the phrase-based SMT system Moses [Koehn et al. 2007] to calculate SMT scores and to produce HFE sentences. The SMT score S was used as the logarithm of $P(M|F)^{\lambda_s}$ in Equation (1), that is, $\lambda_s log(P(M|F)) = S$. The distortion limit of the phrase-based SMT was set to 0. With this setting, the phrase-based SMT translates almost monotonously. The SMT weighting parameters were tuned by MERT using the first half of the development data.

For the process of Equation (1) through the intermediary M, we used a beam search using the ten-best results of M from Moses outputs. For the processes of parsing M to produce T_M, which is represented by $P(T_M|M)$ in Equation (3) and $P(T_M|A, T_F, M, F)$ in Equation (7), we used the ten-best parsing results. The probabilities of the ten-best parsing results were approximated to a constant. With this approximation, the value of $P(T_M|M)^{\lambda_{r3}}$ in Equation (3) and the value of $P(T_M|A, T_F, M)^{\lambda_{r3}}$ in Equation (7) are constant for the ten-best parsing results. Therefore, the value of λ_{r_3} does not affect the results and λ_{r_3} does not need to set for this experiment. As explained in Sections 4.2 and 4.3, $P(E|T_M, M)^{\lambda_{r2}}$ in Equations (3) and (7) is 1 for the E produced from T_M deterministically and is 0 for the other E. Therefore, the value of λ_{r_2} does not affect the results and λ_{r_2} does not need to set for this experiment.

Consequently, the parameters to be set for this experiment are λ_{r_1} and w. The parameter λ_{r_1} scales $P(E)$ in Equations (3) and (7). We used the value of the weighting parameter for the language model feature in the Japanese-HFE SMT model as the value of λ_{r_1} in order to adjust the scale of $P(E)^{\lambda_{r1}}$ in Equations (3) and (7) to the scale of $P(M|E)^{\lambda_s}$, which represents the exponent of the score of the Japanese-HFE SMT in Equation (1). The parameter w adjusts the strength of the effect from T_F for parsing M for the reordering model 2. w was tuned[5] using the second half of the development data. The tuning was based on the BLEU score [Papineni et al. 2002]. In the experiment, using the Moses SMT score S from F to M translation, we searched for the maximum $\lambda_{r_1} log(P(E)) + S$ in the beam search to obtain $\hat{E}$ for Equations (3) and (7).

5.2. Compared Methods

We used the following six comparison methods.

— Phrase-based SMT (PBMT) [Koehn et al. 2003].
— Hierarchical phrase-based SMT (HPBMT) [Chiang 2007].

[3] http://mecab.sourceforge.net/

[4] http://code.google.com/p/cabocha/

[5] We selected the value of w from {0.7, 0.8, 0.9}. $w = 0.8$ was used.

— String-to-tree syntax-based SMT (SBMT) [Hoang et al. 2009].
— Post-ordering based on phrase-based SMT (PO-PBMT) [Sudoh et al. 2011b].
— Post-ordering based on hierarchical phrase-based SMT (PO-HPBMT).
— Post-ordering based on string-to-tree syntax-based SMT (PO-SBMT).

We used Moses [Koehn et al. 2007; Hoang et al. 2009] for these systems. PO-PBMT was the method proposed by Sudoh et al. [2011b]. For PO-PBMT, a distortion limit 0 was used for the Japanese-to-HFE translation, and a distortion limit 20 was used for the HFE-to-English translation. These distortion limit values are the values that achieved the best results in the experiments by Sudoh et al. [2011b]. The PO-HPBMT method changes the post-ordering method of PO-PBMT for the HFE-to-English translation from a phrase-based SMT to a hierarchical phrase-based SMT. The PO-SBMT method changes the post-ordering method of PO-PBMT for the HFE-to-English translation from a phrase-based SMT to a string-to-tree syntax-based SMT. We used a `max-chart-span` of ∞ (unlimited) for the hierarchical phrase-based SMT of PO-HPBMT and the string-to-tree syntax-based SMT of PO-SBMT. We used distortion limits of 12 or 20 for PBMT and `max-chart-spans` of 15 or ∞ (unlimited) for HPBMT and SBMT. For PBMT, a lexicalized reordering model [Koehn et al. 2005], that is, `msd-bidirectional-fe` configuration was used. The default values were used for the other system parameters.

The SMT weighting parameters were tuned by MERT. For PBMT, HPBMT, and SBMT, all of the development data was used for tuning. For the Japanese-to-HFE translation of PO-PBMT, PO-HPBMT, and PO-SBMT, the first half of the development data was used for tuning. For the HFE-to-English translation of PO-PBMT, PO-HPBMT, and PO-SBMT, the following three kinds of data were used for tuning.

— *dev1*. The second half of the development data with HFE produced by translating Japanese using the Japanese-to-HFE SMT.
— *dev1-oracle*. The second half of the development data with HFE that are oracle-HFE made from reference English.
— *dev2-oracle*. The first half of the development data with HFE that are oracle-HFE made from reference English.

5.3. Translation Results and Discussion

We evaluated translation quality based on the case-insensitive automatic evaluation scores RIBES v1.01 [Isozaki et al. 2010a] and BLEU-4 [Papineni et al. 2002]. RIBES is an automatic evaluation measure based on the word-order correlation coefficients between reference sentences and translation outputs. The results are shown in Table I.

The method using reordering model 1 described in Section 4.2 is "Proposed (without T_F)", and the method using reordering model 2 described in Section 4.3 is "Proposed (with T_F)".

We compare the proposed method with T_F to the comparison methods.

First, we made a comparison based on RIBES. For the NTCIR-9 data, the score of the proposed method without T_F was 6.05 points higher than the best score from PO-PBMT and 2.60 points higher than the best score from all of the compared methods (the best method was PO-SBMT (dev1-oracle)). For the NTCIR-8 data, it was 5.64 points higher than the best score from PO-PBMT and 2.69 points higher than the best score from all of the compared methods (the best method was PO-SBMT (dev1-oracle)). The proposed method is thought to be better than the compared methods for global word ordering, since RIBES is sensitive to global word order.

Next, we made a comparison based on the widely used BLEU. For the NTCIR-9 data, the score of the proposed method without T_F was 2.56 points higher than the best score from PO-PBMT and 0.75 points higher than the best score from all of the compared

Table I. Evaluation Results (Case Insensitive)

Japanese-to-English	NTCIR-9		NTCIR-8	
	RIBES	BLEU	RIBES	BLEU
Proposed (without T_F)	75.12	32.95	75.91	34.19
Proposed (with T_F)	**75.48**	**33.04**	**76.44**	**34.47**
PBMT (distortion limit 12)	68.61	29.95	68.93	31.01
PBMT (distortion limit 20)	68.28	30.20	69.10	31.26
HPBMT (max chart span 15)	69.98	30.47	70.65	31.32
HPBMT (max chart span ∞)	70.64	30.69	71.65	31.82
SBMT (max chart span 15)	71.28	31.01	71.84	32.00
SBMT (max chart span ∞)	71.84	31.91	72.53	32.73
PO-PBMT (dev1)	67.16	28.75	68.04	30.21
PO-PBMT (dev1-oracle)	69.08	30.01	70.26	31.55
PO-PBMT (dev2-oracle)	68.81	30.39	69.80	31.71
PO-HPBMT (dev1)	70.28	30.54	71.68	32.07
PO-HPBMT (dev1-oracle)	70.54	30.34	71.62	31.89
PO-HPBMT (dev2-oracle)	70.60	30.40	72.13	32.09
PO-SBMT (dev1)	71.80	32.20	73.02	33.21
PO-SBMT (dev1-oracle)	72.52	32.04	73.22	33.21
PO-SBMT (dev2-oracle)	72.31	31.52	72.90	32.76

methods (the best method was PO-SBMT (dev1)). For the NTCIR-8 data, it was 2.48 points higher than the best score from PO-PBMT and 0.98 points higher than the best score from all of the compared methods (the best method was PO-SBMT (dev1 and dev1-oracle)). The proposed method is also thought to be better than the compared methods for local word ordering, since BLEU is sensitive to local word order.

The differences between the scores of the proposed method without T_F and the top scores from the compared methods were statistically significant at a significance level of $\alpha = 0.01$ for both RIBES and BLEU, using a bootstrap resampling method [Koehn 2004] for a statistical significance test. These comparisons demonstrate the effectiveness of the proposed method without T_F for reordering.

When comparing the proposed method with T_F and without T_F, with T_F is higher than without T_F for both RIBES and BLEU for both NTCIR-9 and NTCIR-8. Since the improvements were not large, we calculated a statistical significance test using a bootstrap resampling method [Koehn 2004] for the differences. For the NTCIR-9 RIBES scores, the difference was statistically significant at a significance level of $\alpha = 0.05$. For the NTCIR-8 RIBES scores, the difference was statistically significant at a significance level of $\alpha = 0.01$. For the NTCIR-9 BLEU scores, the difference was not statistically significant at a significance level of $\alpha = 0.05$, but was statistically significant at a significance level of $\alpha = 0.1$. For the NTCIR-8 BLEU scores, the difference was statistically significant at a significance level of $\alpha = 0.01$. This demonstrates that the method using a Japanese syntactic structure for parsing does have some effectiveness.

In order to investigate the effects of our ITG parsing model more fully, the results with different settings are given here.

We checked different beam widths for the K-best parsing results. Changing the beam widths for K of the K-best parsing results is shown in Figure 5 for the NTCIR-9 test data and in Figure 6 for the NTCIR-8 test data. The beam width K has a slight effect. However, even when K is 1, that is, only the best parsing results were used, the differences between its RIBES and BLEU scores and the best scores were not large. This indicates that the top-ranked parsing results were relatively trustworthy compared to the non-top-ranked parsing results. The top ranked parsing results, for example, three- to ten-best, seem almost sufficient.

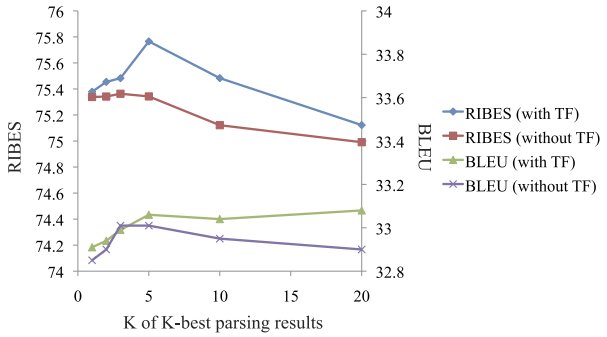

Fig. 5. Different beam widths K of the K-best parsing results for NTCIR-9.

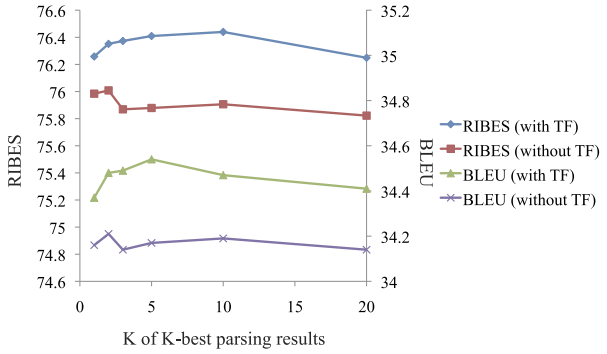

Fig. 6. Different beam widths K of the K-best parsing results for NTCIR-8.

Figure 7 shows the ranking rates of the ten-best parsing results used to produce the final translations for the NTCIR-9 test data.[6] The top-ranked parsing results were used to produce the final translations. This also indicates that the top-ranked parsing results were relatively trustworthy compared to the non-top-ranked parsing results for the following reason: the E of a large $P(E)$ in Equations (3) and (7) is used to produce the final translation. The English sentence E produced from a correct tree derivation T_M will be a natural English sentence E, whose $P(E)$ will be large, and will be used to produce the final translation.

We checked different beam widths for the N-best results of M. The different beam widths N of the N-best results of M are shown in Figure 8 for the NTCIR-9 test data and in Figure 9 for the NTCIR-8 test data. From these figures, a beam width of at least 3 is needed to produce the best results, a beam width of 10 is almost sufficient, and a beam width of 50 is thought to be sufficient.

In these experiments, we did not compare our method to pre-ordering methods. However, some groups used pre-ordering methods in the NTCIR-9 Japanese-to-English translation subtask. The NTT-UT group [Sudoh et al. 2011a] used a pre-ordering method that used parsing trees and manually defined pre-ordering rules. The NAIST group [Kondo et al. 2011] used a pre-ordering method [Tromble and Eisner 2009] that learned a pre-ordering model automatically. These groups were unable to produce

[6]Almost the same results were found for the NTCIR-8 test data, so they are omitted to avoid redundancy.

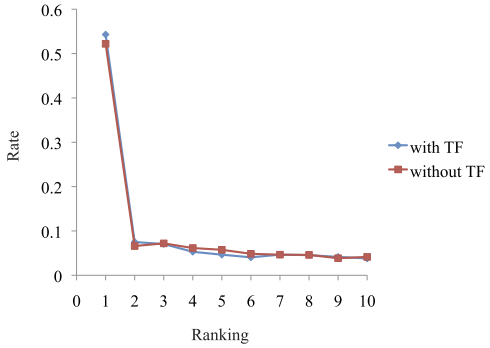

Fig. 7. The ranking rates of the ten-best parsing results used to produce final translations for NTCIR-9. The vertical axis is the rate of results used to produce final translations and the horizontal axis is the ranking of the ten-best parsing results.

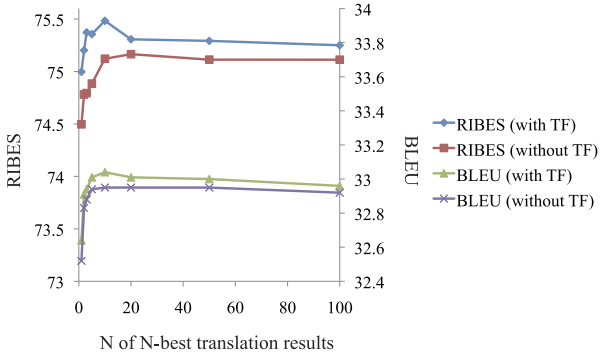

Fig. 8. Different beam widths N of the N-best translation results for NTCIR-9.

both RIBES and BLEU scores that were better than those of the baseline systems of HPBMT and PBMT. In contrast, both the RIBES and BLEU scores for our method were higher than the baseline systems of HPBMT and PBMT. A detailed comparison with pre-ordering methods is our future work.

5.4. Results and Discussion Focusing on Reordering

In order to investigate the effects of our post-ordering method more thoroughly, we conducted an "HFE-to-English reordering" experiment which focuses on the effects of word reordering for the post-ordering framework. This experiment confirms the main contribution of our post-ordering method in the framework of post-ordering SMT, as compared with Sudoh et al. [2011b]. In this experiment, we changed the word order of the oracle-HFE sentences made from reference sentences into English using reordering models. This is the same way as in Table 4 in Sudoh et al. [2011b].

Only the test data (input data) differs from the experiment in the previous section. All other settings are the same. In the experiment in Section 5.3, Japanese sentences were used for the input data. On the other hand, in the experiment in this section, oracle-HFE sentences were used for the input data. The oracle-HFE sentences were produced by (1) parsing the reference English sentences using the Enju parser and (2) applying the head finalization rules [Isozaki et al. 2010b] to the parsing results.

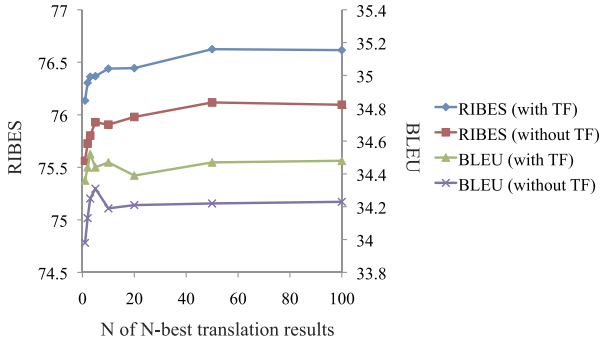

Fig. 9. Different beam widths N of the N-best translation results for NTCIR-8.

Table II. Evaluation Results Focusing on Post-Ordering

oracle-HFE-to-English	NTCIR-9		NTCIR-8	
	RIBES	BLEU	RIBES	BLEU
Proposed (without T_F)	**95.33**	**82.58**	**95.59**	**82.78**
PO-PBMT (dev1)	74.89	57.60	75.75	59.03
PO-PBMT (dev1-oracle)	77.79	60.92	78.76	62.33
PO-PBMT (dev2-oracle)	77.34	62.24	78.14	63.14
PO-HPBMT (dev1)	85.26	65.92	86.54	67.40
PO-HPBMT (dev1-oracle)	85.36	66.13	87.07	67.59
PO-HPBMT (dev2-oracle)	84.76	65.28	85.75	66.88
PO-SBMT (dev1)	87.45	69.28	89.98	73.73
PO-SBMT (dev1-oracle)	88.25	69.91	90.99	74.28
PO-SBMT (dev2-oracle)	87.96	68.75	90.52	72.72

Note that since the oracle-HFE sentences were not produced from Japanese sentences, we only used the proposed method without T_F.

The results are shown in Table II. This results show that our post-ordering method is more effective than PO-PBMT, PO-HPBMT, and PO-SBMT. Since RIBES is based on the rank order correlation coefficient, these results show that the proposed method correctly recovered the word order of the English sentences. These high scores also indicate that the parsing results for high quality HFE are fairly trustworthy.

The causes of reordering errors are classified into distinguishing errors between "_ST" and "_SW" and parsing errors. We investigated how often distinguishing errors occurred. We checked the agreement rate of suffixes ("_ST" or "_SW") between the parsing results by the ITG parsing model (parsed trees) and the tree structures of the test data (oracle trees) for the labels with the following conditions: (1) labels that had suffixes ("_ST" or "_SW"); (2) the subtree spans of the labels are the same in the parsed trees and the oracle trees; and (3) labels without suffixes are the same in the parsed trees and the oracle trees. The agreement rate of suffixes was 99.3% for the NTCIR-9 dataset. We checked the number of hidden states learned for the ITG parsing model. The top three labels are VP_ST (61), VP_SW (56), and NP_ST (53). The number in the parenthesis represents the number of hidden states. Some other major labels are PP_ST (43), S_SW (33), PP_SW (32), S_ST (32), and NP_SW (25). From the high agreement rate, these numbers of hidden states are thought to be enough for learning the distinction between "_ST" and "_SW", and the main cause of errors is thought to be

parsing errors. To improve parsing, techniques for parsing such as these of Petrov [2010] will be useful.

Since there are large differences between the values in Table I and Table II, problems in post-ordering are not entirely solved by improving the reordering accuracy of oracle-HFE. Noise may be included during Japanese-HFE monotone translation. Errors such as word selection errors or lack of translation at the Japanese-HFE monotone translation step cannot be recovered at the reordering step. Using the N-best results for Japanese-HFE monotone translation reduces the effects of these errors compared with using the 1-best result for Japanese-HFE monotone translation. However, this cannot solve the problem perfectly. Word selection is not the only cause of problems. It is rare, but there are word orders in Japanese that cannot be covered by ITG between HFE and English. For example, the fundamental word order of Japanese is SOV, but a word order of OSV is also acceptable in Japanese. An HFE sentence in an OSV word order monotonously translated from a Japanese sentence in an OSV word order cannot be transferred into (S (V O)) by ITG because O and V are not continuous. In this case, it is necessary to convert a Japanese sentence in an OSV word order into a Japanese sentence in an SOV word order at preprocessing.

6. RELATED WORK

This section describes related research other than the aforementioned post-ordering [Matusov et al. 2005; Sudoh et al. 2011b]. Features of our method are as follows.

— Monotonously translated sentences are parsed for reordering in the post-ordering framework.
— Word reordering is done by syntactic transfer based on an ITG model merged with a parsing model.

The post-ordering method splits the word selection and reordering processes. There are many pre-ordering methods that also split the word selection and reordering processes.

Some pre-ordering methods use parsers and manually defined rules for translating different languages. These languages include German to English [Collins et al. 2005], Chinese to English [Wang et al. 2007], English to Hindi [Ramanathan et al. 2008], English to Arabic [Badr et al. 2009], English to Japanese [Isozaki et al. 2010b], and English to five SOV languages (Korean, Japanese, Hindi, Urdu, and Turkish) [Xu et al. 2009]. In English-to-Japanese translation, a pre-ordering method using head finalization rules [Isozaki et al. 2010b], which are used in our post-ordering method, achieved the best quality measured by both RIBES and BLEU, and by the human evaluations which were conducted for the NTCIR-9 patent machine translation task [Goto et al. 2011; Sudoh et al. 2011a]. The reason why this method worked out well is that Japanese is a head-final language, so estimating a Japanese word order based on English is not difficult. On the other hand, English is not a head final language, which makes pre-ordering for Japanese to English more difficult than pre-ordering for the opposite direction, and the pre-ordering method using the head finalization rules cannot be applied. Pre-ordering methods for Japanese to English estimate an English word order based on Japanese. In contrast, the post-ordering methods estimate an English word order based on HFE, which consists of English words. Estimating an English word order based on English words (HFE) is more tractable than estimating an English word order based on Japanese words. This is an advantage of post-ordering methods over pre-ordering methods for Japanese to English translation.

Some pre-ordering methods use parsers and automatically constructed rules [Dyer and Resnik 2010; Ge 2010; Genzel 2010; Habash 2007; Li et al. 2007; Visweswariah et al. 2010; Wu et al. 2011a, 2011b; Xia and McCord 2004]. Li et al. [2007] used

N-best parsing results. Habash [2007] used labeled dependency structures. Dyer and Resnik [2010] used forests based on parsers. Ge [2010] used a manually-aligned corpus to build a pre-ordering model. Genzel [2010] used a dependency parser and tested English into seven languages, including Japanese, and German into English. Wu et al. [2011a] investigated the automatic acquisition of Japanese to English pre-ordering rules using bilingual Japanese and English parsing trees. Wu et al. [2011b] used predicate-argument structures to extract pre-ordering rules and tested English to Japanese.

Some pre-ordering methods do not use supervised parsers. Rottmann and Vogel [2007] proposed a pre-ordering method based on POS. Tromble and Eisner [2009] used ITG constraints to reduce computational costs. DeNero and Uszkoreit [2011] and Neubig et al. [2012] proposed methods for inducing binary tree structures automatically from a parallel corpus with high-quality word alignments and using these structures to preorder source sentences based on ITG. They tested English to Japanese, and Neubig et al. [2012] also tested Japanese to English. Visweswariah et al. [2011] trained a model that used pairwise costs of a word by using a small parallel corpus with high-quality word alignments. They tested Hindi to English, Urdu to English, and English to Hindi.

These are all pre-ordering methods, not post-ordering modes, and thus are different from our method.

The post-edit methods also use a two-step translation process that translates first using a rule-based MT system then post-edits the outputs of the rule-based MT using a phrase-based SMT system [Aikawa and Ruopp 2009; Dugast et al. 2007; Ehara 2007; Simard et al. 2007], or translates first using a syntax-based SMT system then post-edits the outputs of the syntax-based SMT using a phrase-based SMT system [Aikawa and Ruopp 2009]. For Japanese-English translation, the first process changes the word order of Japanese into an English word order and translates, then the post-edit process corrects word selection errors from the first process. This method is similar to pre-ordering methods because the first process mainly decides word order and the second process mainly decides word selection. Thus, these post-edit methods are different from our method.

Our method learns the ITG model [Wu 1997] for reordering. There has also been work done using the ITG model in SMT for joint word selection and reordering. These methods include grammar induction methods from a parallel corpus [Blunsom et al. 2009; Cherry and Lin 2007; Neubig et al. 2011; Zhang et al. 2008]; hierarchical phrase-based SMT [Chiang 2007], which is an extension of ITG; reordering models using ITG [Chen et al. 2009; He et al. 2010]; and ITG constraint for reordering in SMT [Petrov et al. 2008; Zens et al. 2004; Zhang and Gildea 2008]. Note that the aforementioned methods of DeNero and Uszkoreit [2011] and Neubig et al. [2012] also use ITG for training pre-ordering model. However, none of these methods using the ITG model are post-ordering methods.

Our method uses linguistic syntactic structures for reordering. Linguistic syntactic structures have also been used in various works. There are methods that use target language syntactic structures (string-to-tree) [Galley et al. 2004; Shen et al. 2008; Yamada and Knight 2002], methods that use source language syntactic structures (tree-to-string) [Huang et al. 2006; Liu et al. 2006; Quirk et al. 2005], and methods that use both the source and the target language syntactic structures (tree-to-tree) [Chiang 2010; Ding and Palmer 2005; Liu et al. 2009]. These methods do word selection and reordering simultaneously. In contrast, our method does word selection and reordering separately.

Our method is related to tree-to-tree translation methods using syntactic transfer for word reordering. Since Japanese words and English words do not always

correspond one to one, there are large differences between Japanese and English syntactic structures. This makes it difficult to learn syntactic transfer for word reordering. On the other hand, since HFE words and English words always correspond one to one, the difference between HFE and English syntactic structures are smaller than that of Japanese and English. This makes it easier to learn syntactic transfer for word reordering. From these, our method can be regarded to treat a task that learns word reordering based on syntactic transfer for Japanese to English as a more tractable task.

7. CONCLUSION

This article has described a new post-ordering method. Our reordering model consists of a parsing model based on ITG. The proposed method parses sentences that consist of target language words in a source language word order, and does reordering by transferring the syntactic structure similar to the source language syntactic structure into the target language syntactic structure based on ITG. We conducted experiments using Japanese-to-English patent translation. In the experiments, our method outperformed phrase-based SMT, hierarchical phrase-based SMT, string-to-tree syntax-based SMT, and post-ordering methods based on SMT for both RIBES and BLEU. Since RIBES is sensitive to global word order and BLEU is sensitive to local word order, we concluded that the proposed method was better than the compared methods at global word ordering and local word ordering. We also conducted experiments focusing on reordering. These experiments confirmed that our method was able to correctly recover an English word order for high-quality HFE.

REFERENCES

Takako Aikawa and Achim Ruopp. 2009. Chained system: A linear combination of different types of statistical machine translation systems. In *Proceedings of the 12th Machine Translation Summit*. International Association for Machine Translation.

Ibrahim Badr, Rabih Zbib, and James Glass. 2009. Syntactic phrase reordering for English-to-Arabic statistical machine translation. In *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)*. Association for Computational Linguistics, 86–93. http://www.aclweb.org/anthology/E09-1011.

Phil Blunsom, Trevor Cohn, Chris Dyer, and Miles Osborne. 2009. A Gibbs sampler for phrasal synchronous grammar induction. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*. Association for Computational Linguistics, 82–790. http://www.aclweb.org/anthology/P/P09/P09-1088.

Han-Bin Chen, Jian-Cheng Wu, and Jason S. Chang. 2009. Learning bilingual linguistic reordering model for statistical machine translation. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics, 254–262. http://www.aclweb.org/anthology/N/N09/N09-1029.

Stanley F. Chen and Joshua T. Goodman. 1998. An Empirical Study of Smoothing Techniques for Language Modeling. Tech. rep. TR-10-98, Computer Science Group, Harvard University, Cambridge, MA.

Colin Cherry and Dekang Lin. 2007. Inversion transduction grammar for joint phrasal translation modeling. In *Proceedings of the AMTA Workshop on Syntax and Structure in Statistical Translation*. Association for Computational Linguistics, 17–24. http://www.aclweb.org/anthology/W/W07/W07-0403.

David Chiang. 2007. Hierarchical phrase-based translation. *Computat. Linguist. 33*, 2, 201–228.

David Chiang. 2010. Learning to translate with source and target syntax. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 1443–1452. http://www.aclweb.org/anthology/P10-1146.

Michael Collins, Philipp Koehn, and Ivona Kucerova. 2005. Clause restructuring for statistical machine translation. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*. Association for Computational Linguistics, 531–540. http://dx.doi.org/10.3115/1219840.1219906.

John DeNero and Jakob Uszkoreit. 2011. Inducing sentence structure from parallel corpora for reordering. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 193–203. http://www.aclweb.org/anthology/D11-1018.

Yuan Ding and Martha Palmer. 2005. Machine translation using probabilistic synchronous dependency insertion grammars. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*. Association for Computational Linguistics, 541–548. http://dx.doi.org/10.3115/1219840.1219907.

Loïc Dugast, Jean Senellart, and Philipp Koehn. 2007. Statistical post-editing on SYSTRAN's rule-based translation system. In *Proceedings of the 2nd Workshop on Statistical Machine Translation*. Association for Computational Linguistics, 220–223. http://www.aclweb.org/anthology/W/W07/W07-0732.

Chris Dyer and Philip Resnik. 2010. Context-free reordering, finite-state translation. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, 858–866. http://www.aclweb.org/anthology/N10-1128.

Terumasa Ehara. 2007. Rule based machine translation combined with statistical post editor for Japanese to English patent translation. In *Proceedings of the MT Summit XI Workshop on Patent Translation*. International Association for Machine Translation, 13–18.

Atsushi Fujii, Masao Utiyama, Mikio Yamamoto, Takehito Utsuro, Terumasa Ehara, Hiroshi Echizen-ya, and Sayori Shimohata. 2010. Overview of the patent translation task at the NTCIR-8 workshop. In *Proceedings of the 8th NTCIR Workshop Meeting on Evaluation of Information Access Technologies: Informational Retrieval, Question Answering and Cross-Lingual Information Access (NTCIR-8)*. 371–376.

Michel Galley, Mark Hopkins, Kevin Knight, and Daniel Marcu. 2004. What's in a translation rule? In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*. Daniel Marcu, Susan Dumais, and Salim Roukos Eds., Association for Computational Linguistics, 273–280.

Niyu Ge. 2010. A direct syntax-driven reordering model for phrase-based machine translation. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*. Association for Computational Linguistics, 849–857. http://www.aclweb.org/anthology/N10-1127.

Dmitriy Genzel. 2010. Automatically learning source-side reordering rules for large scale machine translation. In *Proceedings of the 23rd International Conference on Computational Linguistics (COLING'10)*. 376–384. http://www.aclweb.org/anthology/C10-1043.

Isao Goto, Bin Lu, Ka Po Chow, Eiichiro Sumita, and Benjamin K. Tsou. 2011. Overview of the patent machine translation task at the NTCIR-9 workshop. In *Proceedings of the 9th NTCIR Workshop Meeting on Evaluation of Information Access Technologies: Informational Retrieval, Question Answering and Cross-Lingual Information Access (NTCIR-9)*. 559–578.

Isao Goto, Masao Utiyama, and Eiichiro Sumita. 2012. Post-ordering by parsing for Japanese-English statistical machine translation. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Association for Computational Linguistics, 311–316. http://www.aclweb.org/anthology/P12-2061.

Nizar Habash. 2007. Syntactic preprocessing for statistical machine translation. In *Proceedings of the Machine Translation Summit XI*. 215–222.

Yanqing He, Yu Zhou, Chengqing Zong, and Huilin Wang. 2010. A novel reordering model based on multilayer phrase for statistical machine translation. In *Proceedings of the 23rd International Conference on Computational Linguistics (COLING'10)*. 447–455. http://www.aclweb.org/anthology/C10-1051.

Hieu Hoang, Philipp Koehn, and Adam Lopez. 2009. A unified framework for phrase-based, hierarchical, and syntax-based statistical machine translation. In *Proceedings of the International Workshop on Spoken Language Translation*. 152–159.

Liang Huang, Kevin Knight, and Aravind Joshi. 2006. Statistical syntax-directed translation with extended domain of locality. In *Proceedings of the 7th Conference of the Association for Machine Translation of the Americas*. 66–73.

Hideki Isozaki, Tsutomu Hirao, Kevin Duh, Katsuhito Sudoh, and Hajime Tsukada. 2010a. Automatic evaluation of translation quality for distant language pairs. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. 944–952.

Hideki Isozaki, Katsuhito Sudoh, Hajime Tsukada, and Kevin Duh. 2010b. Head finalization: A simple reordering rule for SOV languages. In *Proceedings of the Joint 5th Workshop on Statistical Machine Translation and MetricsMATR*. Association for Computational Linguistics, 244–251. http://www.aclweb.org/anthology/W10-1736.

Philipp Koehn. 2004. Statistical significance tests for machine translation evaluation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP'04)*. Dekang Lin and Dekai Wu Eds., Association for Computational Linguistics, 388–395.

Philipp Koehn, Franz J. Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*. 48–54.

Philipp Koehn, Amittai Axelrod, Alexandra Birch Mayne, Chris Callison-Burch, Miles Osborne, and David Talbot. 2005. Edinburgh system description for the 2005 IWSLT speech translation evaluation. In *Proceedings of the International Workshop on Spoken Language Translation*.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*. Association for Computational Linguistics, 177–180. http://www.aclweb.org/anthology/P07-2045.

Shuhei Kondo, Mamoru Komachi, Yuji Matsumoto, Katsuhito Sudoh, Kevin Duh, and Hajime Tsukada. 2011. Learning of linear ordering problems and its application to J-E patent translation in NTCIR-9 PatentMT. In *Proceedings of the 9th NTCIR Workshop Meeting on Evaluation of Information Access Technologies: Informational Retrieval, Question Answering and Cross-Lingual Information Access (NTCIR-9)*. 641–645.

Chi-Ho Li, Minghui Li, Dongdong Zhang, Mu Li, Ming Zhou, and Yi Guan. 2007. A probabilistic approach to syntax-based reordering for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*. Association for Computational Linguistics, 720–727. http://www.aclweb.org/anthology/P07-1091.

Yang Liu, Qun Liu, and Shouxun Lin. 2006. Tree-to-string alignment template for statistical machine translation. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 609–616. http://dx.doi.org/10.3115/1220175.1220252.

Yang Liu, Yajuan Lü, and Qun Liu. 2009. Improving tree-to-tree translation with packed forests. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*. Association for Computational Linguistics, 558–566. http://www.aclweb.org/anthology/P/P09/P09-1063.

E. Matusov, S. Kanthak, and Hermann Ney. 2005. On the integration of speech recognition and statistical machine translation. In *Proceedings of the Annual Conference of the International Speech Communication Association (INTERSPEECH)*. 3177–3180.

Yusuke Miyao and Jun'ichi Tsujii. 2008. Feature forest models for probabilistic HPSG parsing. *Computat. Linguist. 34*, 1, 81–88.

Graham Neubig, Taro Watanabe, Eiichiro Sumita, Shinsuke Mori, and Tatsuya Kawahara. 2011. An unsupervised model for joint phrase alignment and extraction. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, 632–641. http://www.aclweb.org/anthology/P11-1064.

Graham Neubig, Taro Watanabe, and Shinsuke Mori. 2012. Inducing a discriminative parser to optimize machine translation reordering. In *Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*. Association for Computational Linguistics, 843–853. http://www.aclweb.org/anthology/D12-1077.

Franz Josef Och. 2003. Minimum error rate training in statistical machine translation. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 160–167. http://dx.doi.org/10.3115/1075096.1075117.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: A method for automatic evaluation of machine translation. In *Proceedings of 40th Annual Meeting of the Association for Computational Linguistics*. 311–318.

Slav Petrov. 2010. Products of random latent variable grammars. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*. Association for Computational Linguistics, 19–27. http://www.aclweb.org/anthology/N10-1003.

Slav Petrov, Leon Barrett, Romain Thibaux, and Dan Klein. 2006. Learning accurate, compact, and interpretable tree annotation. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 433–440. http://dx.doi.org/10.3115/1220175.1220230.

Slav Petrov, Aria Haghighi, and Dan Klein. 2008. Coarse-to-fine syntactic machine translation using language projections. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 108–116. http://www.aclweb.org/anthology/D08-1012.

Chris Quirk, Arul Menezes, and Colin Cherry. 2005. Dependency treelet translation: Syntactically informed phrasal SMT. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*. Association for Computational Linguistics, 271–279. http://dx.doi.org/10.3115/1219840.1219874.

Ananthakrishnan Ramanathan, Hegde, Jayprasad, Ritesh M. Shah, Pushpak Bhattacharyya, and Sasikumar M. 2008. Simple syntactic and morphological processing can help English-Hindi statistical machine translation. In *Proceedings of the 3rd International Joint Conference on Natural Language Processing*. 171–180.

Kay Rottmann and Stephan Vogel. 2007. Word reordering in statistical machine translation with a POS-based distortion model. In *Proceedings of the 11th Theoretical and Methodological Issues in Machine Translation (TMI)*. 171–180.

Libin Shen, Jinxi Xu, and Ralph Weischedel. 2008. A new string-to-dependency machine translation algorithm with a target dependency language model. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-08: HLT)*. Association for Computational Linguistics, 577–585. http://www.aclweb.org/anthology/P/P08/P08-1066.

Michel Simard, Cyril Goutte, and Pierre Isabelle. 2007. Statistical phrase-based post-editing. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*. Association for Computational Linguistics, 508–515. http://www.aclweb.org/anthology/N/N07/N07-1064.

Andreas Stolcke, Jing Zheng, Wen Wang, and Victor Abrash. 2011. SRILM at sixteen: Update and outlook. In *Proceedings of the IEEE Automatic Speech Recognition and Understanding Workshop*.

Katsuhito Sudoh, Kevin Duh, Hajime Tsukada, Masaaki Nagata, Xianchao Wu, Takuya Matsuzaki, and Jun'ichi Tsujii. 2011a. NTT-UT statistical machine translation in NTCIR-9 PatentMT. In *Proceedings of the 9th NTCIR Workshop Meeting on Evaluation of Information Access Technologies: Informational Retrieval, Question Answering and Cross-Lingual Information Access (NTCIR-9)*. 585–592.

Katsuhito Sudoh, Xianchao Wu, Kevin Duh, Hajime Tsukada, and Masaaki Nagata. 2011b. Post-ordering in statistical machine translation. In *Proceedings of the 13th Machine Translation Summit*. 316–323.

Roy Tromble and Jason Eisner. 2009. Learning linear ordering problems for better translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 1007–1016. http://www.aclweb.org/anthology/D/D09/D09-1105.

Karthik Visweswariah, Jiri Navratil, Jeffrey Sorensen, Vijil Chenthamarakshan, and Nandakishore Kambhatla. 2010. Syntax based reordering with automatically derived rules for improved statistical machine translation. In *Proceedings of the 23rd International Conference on Computational Linguistics (COLING'10)*. 1119–1127. http://www.aclweb.org/anthology/C10-1126.

Karthik Visweswariah, Rajakrishnan Rajkumar, Ankur Gandhe, Ananthakrishnan Ramanathan, and Jiri Navratil. 2011. A word reordering model for improved machine translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 486–496. http://www.aclweb.org/anthology/D11-1045.

Chao Wang, Michael Collins, and Philipp Koehn. 2007. Chinese syntactic reordering for statistical machine translation. In *Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*. Association for Computational Linguistics, 737–745. http://www.aclweb.org/anthology/D/D07/D07-1077.

Dekai Wu. 1997. Stochastic inversion transduction grammars and bilingual parsing of parallel corpora. *Computat. Linguist. 23*, 3, 377–403.

Xianchao Wu, Katsuhito Sudoh, Kevin Duh, Hajime Tsukada, and Masaaki Nagata. 2011a. Extracting pre-ordering rules from chunk-based dependency trees for Japanese-to-English translation. In *Proceedings of the 13th Machine Translation Summit*. 300–307.

Xianchao Wu, Katsuhito Sudoh, Kevin Duh, Hajime Tsukada, and Masaaki Nagata. 2011b. Extracting pre-ordering rules from predicate-argument structures. In *Proceedings of the 5th International Joint Conference on Natural Language Processing*. Asian Federation of Natural Language Processing, 29–37. http://www.aclweb.org/anthology/I11-1004.

Fei Xia and Michael McCord. 2004. Improving a statistical MT system with automatically learned rewrite patterns. In *Proceedings of the International Conference on Computational Linguistics (COLING)*. 508–514.

Peng Xu, Jaeho Kang, Michael Ringgaard, and Franz Och. 2009. Using a dependency parser to improve SMT for subject-object-verb languages. In *Proceedings of the Annual Conference of the North*

American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT). Association for Computational Linguistics, 245–253. http://www.aclweb.org/anthology/N/N09/N09-1028.

Kenji Yamada and Kevin Knight. 2002. A decoder for syntax-based statistical MT. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 303–310. http://dx.doi.org/10.3115/1073083.1073134.

Richard Zens, Hermann Ney, Taro Watanabe, and Eiichiro Sumita. 2004. Reordering constraints for phrase-based statistical machine translation. In *Proceedings of the International Conference on Computational Linguistics (COLING)*. 205–211.

Hao Zhang and Daniel Gildea. 2008. Efficient multi-pass decoding for synchronous context free grammars. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-08: HLT)*. Association for Computational Linguistics, 209–217. http://www.aclweb.org/anthology/P/P08/P08-1025.

Hao Zhang, Chris Quirk, Robert C. Moore, and Daniel Gildea. 2008. Bayesian learning of non-compositional phrases with synchronous parsing. In *Proceedings of Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-08: HLT)*. Association for Computational Linguistics, 97–105. http://www.aclweb.org/anthology/P/P08/P08-1012.

Received October 2012; revised January 2013; accepted February 2013