Practical Approach to Syntax-based Statistical Machine Translation

Kenji Imamura, Hideo Okuma and **Eiichiro Sumita**
ATR Spoken Language Communication Research Laboratories
2-2-2 Hikari-dai, "Keihanna Science City," Kyoto, 619-0288, Japan
{kenji.imamura, hideo.okuma, eiichiro.sumita}@atr.jp

Abstract

This paper presents a practical approach to statistical machine translation (SMT) based on syntactic transfer. Conventionally, phrase-based SMT generates an output sentence by combining phrase (multi-word sequence) translation and phrase reordering without syntax. On the other hand, SMT based on tree-to-tree mapping, which involves syntactic information, is theoretical, so its features remain unclear from the viewpoint of a practical system. The SMT proposed in this paper translates phrases with hierarchical reordering based on the bilingual parse tree. In our experiments, the best translation was obtained when both phrases and syntactic information were used for the translation process.

1 Introduction

Statistical machine translation (SMT), originally proposed by Brown et al. (1993), has evolved from word-level translation to phrase-level (multi-word, i.e., flat phrases in this paper) translation (Koehn et al., 2003; Vogel et al., 2003; Zens and Ney, 2004). In phrase-based SMT, the cost of reordering words is reduced because the word order in a phrase is locally changed before translation. However, reordering phrases is also necessary for accurate translation. Most phrase-based SMT systems reorder phrases on a flat structure.

Another approach, statistical machine translation based on tree-to-tree mapping, explicitly involves syntactic information and hierarchically reordered words (Graehl and Knight, 2004; Melamed, 2004). However, these proposals are theoretical, and thus their features on a practical system remain unclear.

This paper presents a practical method of statistical MT based on syntactic transfer, which is a kind of tree-to-tree mapping. Syntactic transfer has been widely used in machine translation, and it is suitable for a language pair whose respective structures are different (e.g., languages formed by SVO and SOV). An advantage of our method is that not only hierarchical reordering but also the flat phrases handled in phrase-based SMT can be directly applied to the translation.

The rest of this paper is organized as follows. Section 2 briefly describes syntactic-transfer-based MT and its features. Section 3 introduces statistical models for MT using syntactic transfer. Sections 4 and 5 explain the training and decoding methods, respectively. Section 6 discusses features of this method by referring to experiments, and Section 7 discusses related work.

2 Overview of Syntactic-transfer-based MT

2.1 Syntactic-transfer-based MT

Syntactic-transfer-based MT generates translation by parsing an input sentence and mapping the input parse tree to output parse trees. Figure 1 shows an example of the Japanese-to-English translation process by syntactic transfer.

On a practical level, the mapping of parse trees is carried out node by node, since it is infeasible to directly map the entire source tree to the target tree. For instance, we need the following three transfer rules in order to translate the Japanese phrase "*12 ji*" to the English phrase "*12 o'clock*."

- '*12*/NUM' $\Longleftrightarrow$ '*12*/CD'
- '*ji*/NOUN' $\Longleftrightarrow$ '*o'clock*/NN'
- (NP $\rightarrow$ NUM NOUN) $\Longleftrightarrow$ (NP $\rightarrow$ CD NN)

Statistical machine translation that employs IBM models (Brown et al., 1993) inserts or deletes words by using a fertility model and a NULL model. In syntactic-transfer-based MT, the insertion or deletion of words is automatically carried out by mapping of syntactic nodes. For example, when (NP $\rightarrow$ *basu*) is transferred to the NP node in Figure 1, the determiner '*the*' is automatically inserted and (NP $\rightarrow$ *the bus*) is generated. In addition, word reordering is represented by the order of child nodes. Each child node dominates multiple words, so the phrase order is hierarchically modified by the syntactic transfer.

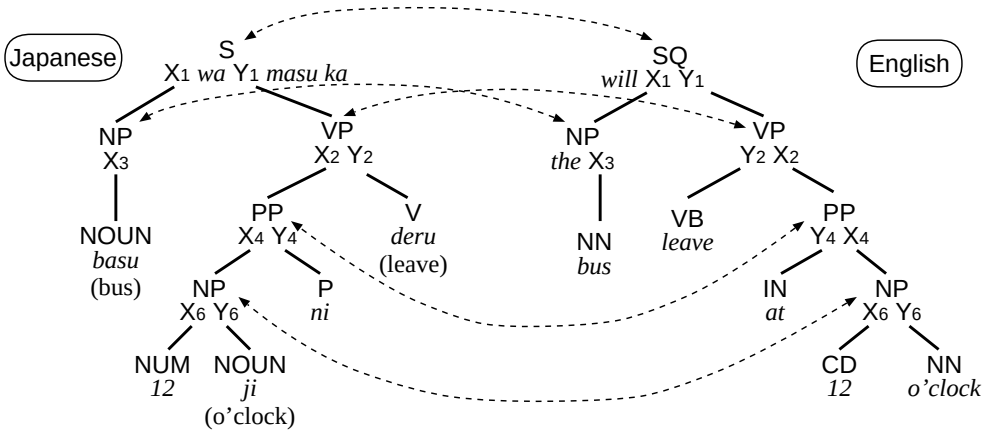

Figure 1: Example of Translation Process (Japanese-to-English) by Syntactic Transfer

Source Tree Table

Name	Parent	Prob.	Children
θ^1	S	$\overset{2.8e-1}{\rightarrow}$	$X_{\texttt{NP}}$ *wa* $Y_{\texttt{VP}}$ *masu ka*
θ^2	VP	$\overset{3.8e-1}{\rightarrow}$	$X_{\texttt{PP}}$ $Y_{\texttt{V}}$
θ^3	VP	$\overset{4.8e-2}{\rightarrow}$	$X_{\texttt{NP}}$ *ni* $Y_{\texttt{V}}$
θ^4	NP	$\overset{5.3e-1}{\rightarrow}$	$X_{\texttt{NN}}$
θ^5	NP	$\overset{8.3e-3}{\rightarrow}$	$X_{\texttt{CD}}$ $Y_{\texttt{NN}}$
		:	

Tree-mapping Table

Source	Probs. Backward	Probs. Forward	Target
θ^1	$\overset{2.0e-1}{\Leftarrow}$	$\overset{1.2e-1}{\Rightarrow}$	π^1
θ^2	$\overset{1.2e-1}{\Leftarrow}$	$\overset{1.9e-1}{\Rightarrow}$	π^2
θ^2	$\overset{4.3e-3}{\Leftarrow}$	$\overset{3.2e-2}{\Rightarrow}$	π^3
:	:		:

Target Tree Table

Name	Parent	Prob.	Children
π^1	SQ	$\overset{3.2e-1}{\rightarrow}$	*will* $X_{\texttt{NP}}$ $Y_{\texttt{VP}}$
π^2	VP	$\overset{4.3e-2}{\rightarrow}$	$Y_{\texttt{VP}}$ $X_{\texttt{NP}}$
π^3	VP	$\overset{1.0e-2}{\rightarrow}$	$Y_{\texttt{VP}}$ $X_{\texttt{PP}}$
π^4	NP	$\overset{1.2e-2}{\rightarrow}$	*the* $X_{\texttt{NN}}$
π^5	NP	$\overset{8.4e-4}{\rightarrow}$	$X_{\texttt{CD}}$ $Y_{\texttt{NN}}$
		:	

Figure 2: Example of Tables for Syntactic-transfer-based Translation Model

We assume that the syntactic transfer is context-free. Namely, each sub-tree is independently transferred without any influence from outside of the sub-tree.

2.2 Flat Phrases

In the syntactic transfer method, flat phrases are regarded as parse trees and handled as ambiguities during translation. In other words, parsing hypotheses, which are flat phrases and words connected by context-free grammar rules, are created. The best hypothesis is selected based on the scoring of SMT. Note that flat phrases require syntactic labels in advance in order to create hypotheses (c.f. Sections 4 and 5).

3 Syntactic-transfer-based Statistical MT

In this section, we explain syntactic-transfer-based MT by using the formalisms of statistical MT.

3.1 Models

Statistical machine translation searches for the best output word sequence $\mathbf{e}$ that maximizes the conditional probability given the input word sequence $\mathbf{f}$. The following equation is used for the search.

$$\begin{aligned}\hat{\mathbf{e}} &= \operatorname*{argmax}_{\mathbf{e}} P(\mathbf{e}|\mathbf{f}) \\ &= \operatorname*{argmax}_{\mathbf{e}} P(\mathbf{e})P(\mathbf{f}|\mathbf{e}). \end{aligned} \quad (1)$$

$P(\mathbf{e})$ is called the language model, and $P(\mathbf{f}|\mathbf{e})$ is called the translation model. Syntactic-transfer-based SMT assumes source and target parse trees as the hidden variables in the translation model, and translation is done by mapping between the parse trees.

$$\begin{aligned}P(\mathbf{f}|\mathbf{e}) &= \sum_{\mathcal{F},\mathcal{E}} P(\mathbf{f},\mathcal{F},\mathcal{E}|\mathbf{e}) \\ &\approx \sum_{\mathcal{F},\mathcal{E}} P(\mathbf{f}|\mathcal{F})P(\mathcal{F}|\mathcal{E})P(\mathcal{E}|\mathbf{e}), \end{aligned} \quad (2)$$

where $\mathcal{F}$ and $\mathcal{E}$ denote the source and target parse trees and they yield $\mathbf{f}$ and $\mathbf{e}$, respectively.

In this paper, we call $P(\mathcal{E}|\mathbf{e})$ the target tree model and $P(\mathcal{F}|\mathcal{E})$ the tree-mapping model from target to source. Models are assumed to be independent of each other. The probability $P(\mathbf{f}|\mathcal{F})$ is 1.0 by its definition, so Equation 2 is modified as follows.

$$P(\mathbf{f}|\mathbf{e}) \approx \sum_{\mathcal{F},\mathcal{E}} P(\mathcal{F}|\mathcal{E})P(\mathcal{E}|\mathbf{e}). \quad (3)$$

3.2 Bidirectional Translation Model

The target word sequence $\hat{\mathbf{e}}$ is the same if the probability of Equation 1 is multiplied by itself. Therefore, we can obtain a bidirectional translation model by the following modification.

$$\begin{aligned}
&\operatorname*{argmax}_{\mathbf{e}} P(\mathbf{e}|\mathbf{f}) \\
&= \operatorname*{argmax}_{\mathbf{e}} P(\mathbf{e}|\mathbf{f})^2 \\
&= \operatorname*{argmax}_{\mathbf{e}} P(\mathbf{e})P(\mathbf{f}|\mathbf{e})P(\mathbf{e}|\mathbf{f}). \qquad (4)
\end{aligned}$$

According to the above representation, the final score of the translation model is given by multiplying the translation model from source to target ($P(\mathbf{e}|\mathbf{f})$, forward translation model) and the translation model from target to source ($P(\mathbf{f}|\mathbf{e})$, backward translation model). Then, we obtain the score of the bidirectional translation model by applying Equation 3 to these models as follows.

$$\begin{aligned}
&P(\mathbf{f}|\mathbf{e})P(\mathbf{e}|\mathbf{f}) \\
&= \sum_{\mathcal{F}',\mathcal{E}'} P(\mathcal{F}'|\mathcal{E}')P(\mathcal{E}'|\mathbf{e}) \\
&\quad \cdot \sum_{\mathcal{E}'',\mathcal{F}''} P(\mathcal{E}''|\mathcal{F}'')P(\mathcal{F}''|\mathbf{f}) \\
&\approx \sum_{\mathcal{F},\mathcal{E}} P(\mathcal{F}|\mathbf{f})P(\mathcal{E}|\mathcal{F})P(\mathcal{F}|\mathcal{E})P(\mathcal{E}|\mathbf{e}). \qquad (5)
\end{aligned}$$

Equation 5 is regarded as the score that the source sentence is translated to a target sentence and then the target is translated to the same source again. In other words, the bidirectional translation model includes backward translation probability. Backward translation is used for checking the correctness of MT results (Yasuda et al., 2003), utilizing a feature that the most incorrect translation cannot restore the original source sentence. The bidirectional translation model involves the above feature, so we can expect the following effects.

- Incorrect translation caused by incorrect parameters in the models is reduced. [1]
- Since the model includes the source tree model, a correct parse tree of the source sentence is derived from the viewpoint of the source language.

[1] If the models were perfect, the bidirectional translation model would not be necessary. However, our models contain many incorrect parameters due to inperfect training or rough approximation, so we employed this approach.

Equation 5 is nearly equal to the log linear model (Och and Ney, 2002), in which the feature functions are probabilities of source/target tree models and tree mapping models, and the weights of the models are uniform.

3.3 Inside Probability

The source and target tree models can be regarded as probabilistic context-free grammar (PCFG). Namely, nodes in the tree are generated independently of each other, and the probability of the tree is computed by the product of the probabilities of a parent node generating a child node sequence (i.e., the inside probability).

$$P(\mathcal{F}|\mathbf{f}) = \prod_{\theta:\theta\in\mathcal{F}} P(\theta), \qquad (6)$$

$$P(\mathcal{E}|\mathbf{e}) = \prod_{\pi:\pi\in\mathcal{E}} P(\pi). \qquad (7)$$

Here, θ and π denote context-free grammar rules that construct $\mathcal{F}$ and $\mathcal{E}$, respectively.

The probability of the forward and backward tree-mapping model is computed by the following equation in the same manner as the inside probability. Figure 2 shows an example of these models.

$$P(\mathcal{E}|\mathcal{F}) = \prod_{\substack{\theta:\theta\in\mathcal{F},\\ \pi:\pi\in\mathcal{E}}} P(\pi|\theta), \qquad (8)$$

$$P(\mathcal{F}|\mathcal{E}) = \prod_{\substack{\theta:\theta\in\mathcal{F},\\ \pi:\pi\in\mathcal{E}}} P(\theta|\pi). \qquad (9)$$

By using the inside probability, the score of the bidirectional translation model can be computed recursively from that of child sub-trees.

For example, supposing that a bilingual node N^i directly contains K child sub-trees (i.e., N_f^i is the top node of $\mathcal{F}^i$ that yields $\mathbf{f}^i$, and θ^i is the CFG rule of $N_f^i \rightarrow N_f^{i+1} \cdots N_f^{i+K}$. N_e^i, $\mathcal{E}^i$, and π^i are those of the target side), the score is computed by the following equation.

$$\begin{aligned}
&P(\mathcal{F}^i|\mathbf{f}^i)P(\mathcal{E}^i|\mathcal{F}^i)P(\mathcal{F}^i|\mathcal{E}^i)P(\mathcal{E}^i|\mathbf{e}^i) \\
&= \prod_{\substack{\theta:\theta\in\mathcal{F}^i\\ \pi:\pi\in\mathcal{E}^i}} P(\theta)P(\pi|\theta)P(\theta|\pi)P(\pi) \\
&= P(\theta^i)P(\pi^i|\theta^i)P(\theta^i|\pi^i)P(\pi^i) \\
&\quad \cdot \prod_{j=1}^{K} \left\{ \begin{array}{l} P(\mathcal{F}^{i+j}|\mathbf{f}^{i+j})P(\mathcal{E}^{i+j}|\mathcal{F}^{i+j}) \\ \cdot P(\mathcal{F}^{i+j}|\mathcal{E}^{i+j})P(\mathcal{E}^{i+j}|\mathbf{e}^{i+j}) \end{array} \right\}. \qquad (10)
\end{aligned}$$

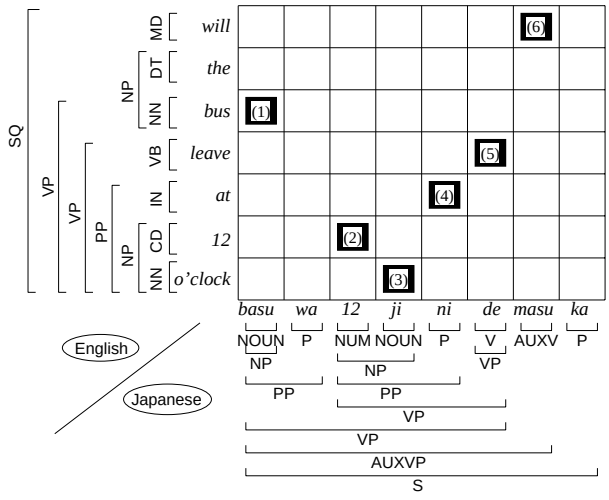

Figure 3: Example of Phrase Alignment

Therefore, the score of the bidirectional translation model can be computed in a bottom-up manner, so it can also naturally be applied to a bottom-up parser that parses two languages.

4 Training

In the training phase, we assume that parse trees are given in advance in order to reduce complexity. Therefore, the problems in the training phase are (1) extracting corresponding nodes between bilingual trees (phrase alignment) and (2) estimating probabilities of the source and target tree models and the tree-mapping model.

4.1 Phrase Alignment

The phrase alignment used here is a similar approach to acquisition of alignment templates (Och and Ney, 2004), which extracts phrases based on the continuity of word alignment. The phrase alignment in this paper uses not only the continuity of word alignment but also the constraints of parse trees. Namely, only phrases that are a part of the source and target trees are extracted.

Figure 3 shows an example of this phrase alignment. The phrases are extracted as follows.

1. First, perform word alignment in both directions (source to target and target to source). We use Viterbi alignment of IBM model 4 learned by GIZA++ (Och and Ney, 2003).

2. Next, extract sure alignments, i.e., those that agree with Viterbi alignments in both directions. The alignments that do not agree in both directions are regarded as possible alignments.

3. For each combination of the sure alignments, extract correspondences of syntactic nodes that only contain the combination and exclude the other sure alignments. If correspondence is ambiguous, the correspondence that contains the most possible alignments is selected.

For example, by focusing on the sure alignments (2) and (3) in Figure 3, the pair (NP → *12 o'clock*) and (NP → *12 ji*) is extracted as a bilingual phrase because it only contains (2) and (3) (i.e., it does not contain the sure alignments (1), (4), (5), and (6)). However, focusing on the sure alignments (4) and (5), there are no sub-trees that contain only them and thus do not contain (1), (2), (3), and (6). Therefore, the English word sequence "*leave at*" and the Japanese word sequence "*ni de*" are not extracted as a bilingual phrase. Table 1 shows a list of the extracted phrases from Figure 3.

The bilingual phrases extracted here are regarded as the flat phrases. They can be directly applied to syntactic parsing because they have syntactic labels.

Moreover, the results of phrase alignment maintain hierarchy. Using this information, context-free grammar rules of source and target language are generated. For example, the bilingual phrase 9 in Table 1 dominates phrases 8 and 7 in its children. If phrases 8 and 7 are generalized as non-terminal symbols, an English grammar rule VP → VB PP and a Japanese rule VP → PP V are acquired as the tree-mapping rule.

4.2 Parameter Estimation

In this paper, all probabilities of models are estimated from relative frequencies. For example, a probability in the source tree model is estimated by the following equation.

$$\begin{aligned} P(\theta^i) &= P(N_f^{i+1} \ldots N_f^{i+K} | N_f^i) \\ &= \frac{count(N_f^{i+1} \ldots N_f^{i+K}, N_f^i)}{count(N_f^i)}, \end{aligned} \quad (11)$$

where $count(N)$ denotes the frequency of the syntactic node N that appears in a training corpus. A probability in the target tree model is estimated in the same way as done for the source tree model.

A probability in the forward tree-mapping model is estimated from the following equation. The backward tree-mapping model is estimated in the same way.

$$P(\pi^i | \theta^i) = \frac{count(\pi^i, \theta^i)}{count(\theta^i)}. \quad (12)$$

No.	Japanese			English		
1	NOUN	→	*basu*	NN	→	*bus*
2	NP	→	*basu*	NP	→	*the bus*
3	NUM	→	*12*	CD	→	*12*
4	NOUN	→	*ji*	NN	→	*o'clock*
5	NP	→	*12 ji*	NP	→	*12 o'clock*
6	P	→	*ni*	IN	→	*at*
7	PP	→	*12 ji ni*	PP	→	*at 12 o'clock*
8	V	→	*de*	VB	→	*leave*
9	VP	→	*12 ji ni de*	VP	→	*leave at 12 o'clock*
10	S	→	*basu wa 12 ji ni de masu ka*	SQ	→	*will the bus leave at 12 o'clock*

Table 1: Extracted Phrases from Figure 3

Our method does not distinguish between words and phrases, so the translation probabilities of words are re-estimated by relative frequencies. No smoothing is performed in the experiments of this paper.

5 Decoding

Our method first parses the input sentence. Therefore, the decoder is realized by a CFG parser supplemented with transfer and generation modules. In this paper, we utilize a bottom-up chart parser. The process of decoding is as follows.

1. First, parse the input sentence using the source tree model in a bottom-up manner.
2. When a sub-tree of the input sentence is built, refer to the tree-mapping model and the target tree model and then construct sub-trees of the output sentence.
3. For each output sub-tree, serialize it and generate word sequences of the output. The probability of the output word sequence is given by the product of Equation 10 and the language model probability.
4. Merge the listed output word sequences so that the input word sequence and the syntactic labels of the trees are identical. Then, the top n sequences of the highest scores are kept as the translation results.
5. Repeat Steps 1 to 4 until the entire input sentence is parsed.

Figure 4 shows an example in which the Japanese partial sentence "*12 ji ni de*" is translated into English. If the parsing result of the input sentence is ambiguous, or there are multiple mappings, each input tree is mapped one by one, and the listed output (partial) sentences are merged in Step 4. Through this process, not only the output sequence but also the syntactic labels of the input and output are acquired, so the decoder can parse and transfer the higher structure.

If the parsing of the input (or output) sentence fails, the decoder extracts partial translations from its agenda and sequentially outputs the translations whose products of the probabilities are the highest.

6 Experiments

We evaluate the proposed method through Japanese to English translation.

6.1 Experimental Settings

Corpora: The corpus used here is the Basic Travel Expression Corpus (BTEC) (Takezawa et al., 2002; Kikui et al., 2003). This is a collection of Japanese sentences and their English translations based on expressions that are usually found in phrasebooks for foreign tourists. The corpus size is shown in Table 2. IWSLT in Table 2 is a corpus used in the evaluation campaign of the International Workshop on Spoken Language Translation (Akiba et al., 2004), which is a subset of BTEC. The test set is the same as that of IWSLT. [2]

Training: Word alignment was acquired from the Viterbi alignment of IBM model 4 using GIZA++ (Och and Ney, 2003). Charniak (2000)'s parser was used for English parsing, and a rule-based phrase structure parser developed in-house was used for Japanese parsing in the training phase.

Word bigram and trigram models learned by CMU-Cambridge Statistical Language Modeling Toolkit (Clarkson and Rosenfeld, 1997) formed the language model.

Evaluation Metrics: We used BLEU (Papineni et al., 2002), NIST (Doddington, 2002), and mWER (multiple Word Error Rate, (Nießen et al., 2000))

[2] In these experiments, we arranged numerical words into numbers (e.g., "fifty/CD one/CD" → '51/CD'), so the number of words is different from that of IWSLT.

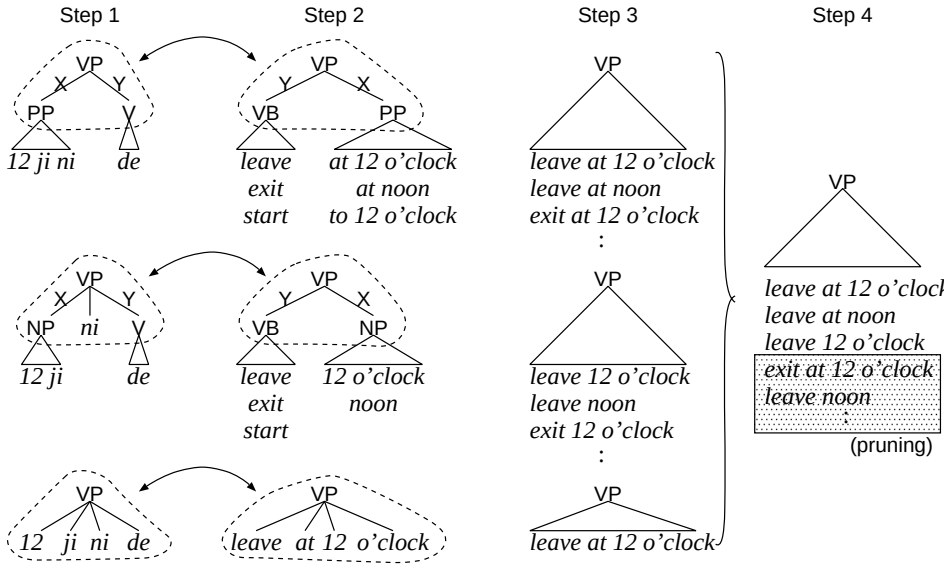

Figure 4: Example of Decoding

Set Name	Item	Japanese	English
BTEC	# of Sentences	152,170	
(Training)	# of Words	1,178,419	1,103,600
	# of Diff. Words	16,686	10,669
IWSLT	# of Sentences	20,000	
(Training)	# of Words	188,533	182,018
	# of Diff. Words	8,652	6,133
Test	# of Sentences	500	—
	# of Words	4,018	—
	# of Diff. Words	888	—

Table 2: Corpus Size

metrics for the automatic evaluation. For the subjective evaluation, an English native classified the translations into the four ranks of A: Perfect, B: Fair, C: Acceptable, and D: Nonsense (Sumita et al., 1999). Note that a lower score denotes a better translation in the mWER metric.

6.2 Results

6.2.1 Translation Quality

First, we measured the translation quality of the proposed method. The results are shown in Table 3. In order to measure the effect of syntactic transfer and flat phrases independently, two alternative methods were applied:

- Flat phrases were excluded from the translation model (w/o phrases). Only the most primitive rules were applied to decoding.
- Decoding was performed without syntactic information. We used the phrase-based beam search decoder, Pharaoh, developed by USC ISI (Koehn et al., 2003) [3]. The same phrase set as used by the proposed method was used for decoding. [4]

In comparing the methods, the best result was obtained by the proposed method, which uses syntax and flat phrases for decoding, for both BTEC and IWSLT by all metrics. The quality of the proposed method without flat phrases is better than that of Pharaoh. We suppose that this is due to using the bidirectional translation model. However, in syntactic-transfer-based MT, we can simultaneously utilize flat phrases and syntax, so both approaches should be used to improve translation quality.

6.2.2 Parsing Failure

Since syntactic-transfer-based MT performs parsing, a higher structure could not be built if the parsing failed to build the lower structure due to the lack of grammar rules. In this experiment, 42 sentences (8.4%) showed failed parsing in BTEC, and 110 sentences (22%) had failed parsing in IWSLT.

Figure 5 shows that the subjective quality distinguished parsing success and failure in the case of BTEC. The quality is clearly better when the parsing succeeded, and no translation became perfect when the parsing failed.

[3] http://www.isi.edu/licensed-sw/pharaoh/

[4] The weights of models for Pharaoh are experimentally determined to minimize mWER by using a development set of BTEC. Uniform weights are used for the proposed decoder, and the 20-best is set.

Training Set	System	Automatic Evaluation mWER	BLEU	NIST	Subjective Evaluation A	A+B	A+B+C
BTEC	Proposed System (w/ phrases)	**0.288**	**0.638**	**10.18**	**68.2%**	**76.6%**	**83.0%**
	Proposed System (w/o phrases)	0.334	0.566	9.20	62.0%	72.6%	78.8%
	Pharaoh (w/ phrases)	0.372	0.530	9.72	55.4%	66.0%	75.6%
IWSLT	Proposed System (w/ phrases)	**0.476**	**0.414**	**8.38**	**43.4%**	**55.0%**	**63.8%**
	Proposed System (w/o phrases)	0.478	0.387	8.01	41.6%	53.8%	63.0%
	Pharaoh (w/ phrases)	0.523	0.382	7.65	32.0%	45.8%	58.2%

Table 3: Translation Quality (Japanese-to-English)

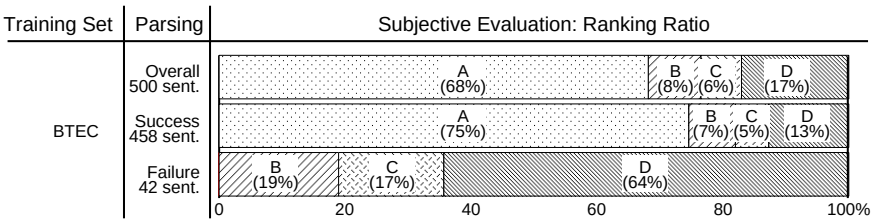

Figure 5: Subjective Evaluation According to Parsing Success/Failure (in Proposed Method with Phrases)

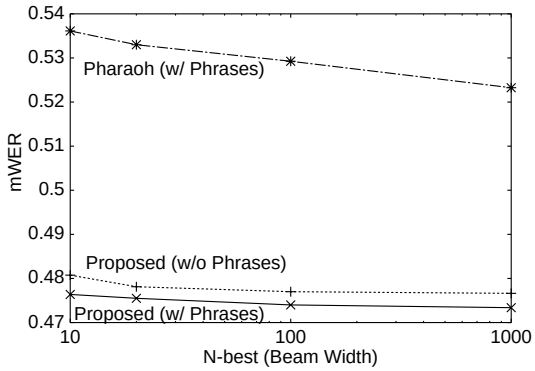

Figure 6: Changes of Translation Quality According to N-best Size (mWER)

In contrast with monolingual parsing, our method experiences parsing failure more frequently because (1) both the source and the target sentences must be parsed, and (2) only grammar rules on one side of the translation equivalence can be applied to parsing. Reduce parsing failure is a task that must be accomplished to improve translation quality.

6.2.3 Translation Quality According to N-best Size

Figure 6 shows the changes in the multiple word error rates according to the n-best size in the case of IWSLT. The translation quality by Pharaoh improved along with the expansion of the beam width. In the proposed methods, the quality was nearly fixed to the n-best size.

Generally, beam search decoders decode from the head or tail of the sentence. On the other hand, a syntactic-transfer-based decoder decodes the input sentence from the content words composing the base NP phrases, while functional words are transferred later. The translation of most functional words cannot be determined until content words are translated. We assume that this explains how good translations could be obtained even with a small n-best size.

7 Related Work

Statistical machine translation that employs syntax has been proposed as outlined below.

Yamada and Knight (2001) and Charniak et al. (2003) proposed translation and language models in which the input sentence is mapped to the output parse tree. Even though they only used parse trees for one side, while we use both sides, a syntax-based language model would improve the fluency of translation.

Graehl and Knight (2004) and Melamed (2004) proposed theoretical models that employ parse trees of source and target languages. Our proposed method is a realization of these methods.

On the other hand, Koehn et al. (2003), Vogel et al. (2003), and Zens and Ney (2004) proposed phrase-based statistical MTs that do not use syntactic information. Koehn et al. (2003) reported that the translation quality was degraded if flat phrases were constrained by parse trees in the training phase. However, as we mentioned above, syntactic information would improve translation quality if it were used in the decoding phase.

8 Conclusions

This paper presented a syntactic-transfer-based method of statistical MT. The proposed method can combine the syntax and flat phrases used in phrase-based SMT. Experiments verified that translation quality improved by combining syntactic transfer and flat phrases.

Since a syntactic-transfer-based decoder decodes an input sentence from content words to functional words, high-quality translation can be obtained even if the n-best size is small.

The uniform weights of the models were used in this paper. We will attempt to optimize the weights using the log-linear model.

Acknowledgment

This research was supported in part by the National Institute of Information and Communications Technology.

References

Yasuhiro Akiba, Marcello Federico, Noriko Kando, Hiromi Nakaiwa, Michael Paul, and Jun'ichi Tsujii. 2004. Overview of the IWSLT04 evaluation campaign. In *IWSLT 2004 Proceedings*, pages 1–12.

Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. 1993. The mathematics of machine translation: Parameter estimation. *Computational Linguistics*, 19(2):263–311.

Eugene Charniak, Kevin Knight, and Kenji Yamada. 2003. Syntax-based language models for statistical machine translation. In *Proceedings of Machine Translation Summit IX*, pages 40–46.

Eugene Charniak. 2000. A maximum-entropy-inspired parser. In *Proceedings of the 1st Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-2000)*, pages 132–139.

Philip Clarkson and Ronald Rosenfeld. 1997. Statistical language modeling using the CMU-Cambridge toolkit. In *Proceedings of EuroSpeech 97*, pages 2707–2710.

George Doddington. 2002. Automatic evaluation of machine translation quality using n-gram co-occurrence statistics. In *Proceedings of the HLT Conference*.

Jonathan Graehl and Kevin Knight. 2004. Training tree transducers. In *HLT-NAACL 2004: Main Proceedings*, pages 105–112. Association for Computational Linguistics.

Genichiro Kikui, Eiichiro Sumita, Toshiyuki Takezawa, and Seiichi Yamamoto. 2003. Creating corpora for speech-to-speech translation. In *Proceedings of EuroSpeech 2003*, pages 381–384.

Philipp Koehn, Franz J. Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *HLT-NAACL 2003: Main Proceedings*, pages 127–133. Association for Computational Linguistics.

I. Dan Melamed. 2004. Statistical machine translation by parsing. In *Proceedings of the 42nd Meeting of the Association for Computational Linguistics (ACL'04), Main Volume*, pages 653–660.

Sonja Nießen, Franz Josef Och, Gregor Leusch, and Hermann Ney. 2000. An evaluation tool for machine translation: Fast evaluation for MT research. In *Proceedings of 2nd International Conference on Language Resources and Evaluation (LREC 2000)*, pages 39–46.

Franz Josef Och and Hermann Ney. 2002. Discriminative training and maximum entropy models for statistical machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 295–302.

Franz Josef Och and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51.

Franz Josef Och and Hermann Ney. 2004. The alignment template approach to statistical machine translation. *Computational Linguistics*, 30(4):417–449.

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

Eiichiro Sumita, Setsuo Yamada, Kazuhide Yamamoto, Michael Paul, Hideki Kashioka, Kai Ishikawa, and Satoshi Shirai. 1999. Solutions to problems inherent in spoken-language translation: The ATR-MATRIX approach. In *Machine Translation Summit VII*, pages 229–235.

Toshiyuki Takezawa, Eiichiro Sumita, Fumiaki Sugaya, Hirofumi Yamamoto, and Seiichi Yamamoto. 2002. Toward a broad-coverage bilingual corpus for speech translation of travel conversations in the real world. In *Proceedings of the Third International Conference on Language Resources and Evaluation (LREC 2002)*, pages 147–152.

Stephan Vogel, Ying Zhang, Fei Huang, Alicia Tribble, Ashish Venugopal, Bing Zhao, and Alex Waibel. 2003. The CMU statistical machine translation system. In *Proceedings of the 9th Machine Translation Summit (MT Summit IX)*, pages 402–409.

Kenji Yamada and Kevin Knight. 2001. A syntax-based statistical translation model. In *Proceedings of 39th Annual Meeting of the Association for Computational Linguistics*, pages 523–530.

Keiji Yasuda, Eiichiro Sumita, Genichiro Kikui, Seiichi Yamamoto, and Masuzo Yanagida. 2003. Real-time evaluation architecture for MT using multiple backward translations. In *Proceedings of Recent Advances in Natural Language Processing (RANLP-2003)*, pages 518–522.

Richard Zens and Hermann Ney. 2004. Improvements in phrase-based statistical machine translation. In *HLT-NAACL 2004: Main Proceedings*, pages 257–264. Association for Computational Linguistics.