

Imposing Constraints from the Source Tree on ITG Constraints for SMT

Hirofumi Yamamoto^{†,††,†††}

Hideo Okuma^{†,††}

Eiichiro Sumita^{†,††}

[†]National Institute of Information and Communications Technology

/ 2-2-2 Hikaridai Seika-cho Soraku-gun Kyoto Japan

^{††}ATR Spoken Language Communication Research Labs.

^{†††}Kinki University School of Science and Engineering Department of Information

{hirofumi.yamamoto, hideo.okuma, eichiro.sumita}@nict.go.jp

Abstract

In current statistical machine translation (SMT), erroneous word reordering is one of the most serious problems. To resolve this problem, many word-reordering constraint techniques have been proposed. The inversion transduction grammar (ITG) is one of these constraints. In ITG constraints, target-side word order is obtained by rotating nodes of the source-side binary tree. In these node rotations, the source binary tree instance is not considered. Therefore, stronger constraints for word reordering can be obtained by imposing further constraints derived from the source tree on the ITG constraints. For example, for the source word sequence { a b c d }, ITG constraints allow a total of twenty-two target word orderings. However, when the source binary tree instance ((a b) (c d)) is given, our proposed "imposing source tree on ITG" (IST-ITG) constraints allow only eight word orderings. The reduction in the number of word-order permutations by our proposed stronger constraints efficiently suppresses erroneous word orderings. In our experiments with IST-ITG using the NIST MT08 English-to-Chinese translation track's data, the proposed method resulted in a 1.8-points improvement in character BLEU-4 (35.2 to 37.0) and a 6.2% lower CER (74.1 to 67.9%) compared with our baseline condition.

1 Introduction

Statistical methods are widely used for machine translation. One of the popular statistical machine

translation paradigms is the phrase-based model (PBSMT) (Marcu et al., 2002; Koehn et al., 2003; Och et al., 2004). In PBSMT, errors in word reordering, especially in global reordering, are one of the most serious problems. Approaches used to resolve this problem are categorized into two types. The first type is linguistically syntax-based. In this approach, source (Quirk et al., 2005; Liu et al., 2006; Huang et al., 2006), target (Yamada et al., 2000; Galley et al., 2006; Marcu et al., 2006), or both side (Melamed 2004; Ding et al., 2005) tree structures are used for model training. The second type is formal constraints on word permutations. IBM constraints (Berger et al., 1996), lexical word reordering model (Tillmann, 2004), and inversion transduction grammar (ITG) constraints (Wu, 1995; Wu, 1997) belong to this type of approach. Our approach is an extension of ITG constraints and is a hybrid of the first and second type of approach.

We propose "imposing source tree on ITG" (IST-ITG) constraints for directly introducing source sentence structure into our set of constraints. In IST-ITG, ITG constraints under the given source sentence tree structure are used as stronger constraints than the original ITG. For example, IST-ITG allows only eight word orderings for a four-word sentence, even though twenty-two word orderings are possible with respect of in the original ITG constraints.

In Section 2, we present the proposed IST-ITG for word-based translation. In Section 3, the proposed method is extended to phrase-based translation. In Section 4, we present a real-time decoding algorithm for IST-ITG constraints. In Section 5, we give details of the experiments and present the results. Finally, in Section 6, we offer a summary and some concluding remarks.

2 Imposing the Source Tree on ITG Constraints

First, we introduce three previous studies on word reordering constraints: IBM constraints; lexical reordering model; and ITG constraints. Here, we consider one-to-one word-aligned source and target language sentence pairs as the simplest cases.

2.1 IBM constraints

In this constraint, a distortion penalty is given in accordance with the gap between the previously and the currently translated words, which is represented as the following equation.

$$p_D = \exp(-\sum_i d_i) \quad (1)$$

where d_i for each i is defined as:

$$d_i = \text{abs}(\text{position}(e_{i-1}) + 1 - \text{position}(e_i)) \quad (2)$$

where e_i represents the translated word from the i th source word f_i , $\text{position}(w)$ represents the position of the word w . Sometimes, a limit is set for d_i for similar language pairs such as French and English. However, for dissimilar language pairs, such as Japanese and English or Chinese and English, limiting d_i is not beneficial.

2.2 Lexical Reordering Model

In the lexical reordering model, reordering probabilities are assigned to each word pair $\{f_i, e_i\}$. Reordering positions are categorized into three types, monotone, swap, and discontinuous. The probability is assigned to left and right sides as $p_s(t|f_i, e_i)$, where, s is left (l) or right (r), t is monotone (m), swap (s), or discontinuous (d). Therefore, a total of six probabilities are assigned to each word pair. For the source word sub-sequence f_{i-1}, f_i , probabilities of target sub-sequences are calculated as follows:

- $p(e_{i-1}, e_i) = p_r(m|f_{i-1}, e_{i-1})p_l(m|f_i, e_i)$
- $p(e_i, e_{i-1}) = p_r(s|f_{i-1}, e_{i-1})p_l(s|f_i, e_i)$
- $p(\text{otherwise}) = p_r(d|f_{i-1}, e_{i-1})p_l(d|f_i, e_i)$

2.3 ITG Constraints

In one-to-one word-alignment, the source word f_i is translated into the target word e_i . The source sentence $[f_1, f_2, \dots, f_N]$ is translated into a reordered sequence of word $[e_1, e_2, \dots, e_N]$. The number of reorderings is $N!$. When ITG is introduced, this combination $N!$ can be reduced in accordance with the following constraints.

- All possible binary tree structures are generated from the source word sequence.
- The target sentence is obtained by rotating any node of the binary trees.

When $N = 4$, the ITG constraints can reduce the number of combinations from $4! = 24$ to 22 by rejecting combinations $[e_3, e_1, e_4, e_2]$ and $[e_2, e_4, e_1, e_3]$. For a 4-word sentence, the search space is reduced to 92% ($22/24$), but for 10-word sentence, the search space is only 6% ($206,098/3,628,800$) of the original full space.

2.4 Imposing Source Tree Constraints

In ITG constraints, the source-side binary tree instance is not considered. Therefore, if the source sentence binary tree is utilized, stronger constraints than the original ITG can be created. By parsing the source sentence, a parse tree is obtained. After parsing, a bracketed sentence is obtained by removing the node labels, and this bracketed sentence can be converted to a binary tree. For example, the parse tree, (S1 (S (NP (DT This)) (VP (AUX is) (NP (DT a) (NN pen))))), is obtained from the source sentence "This is a pen". By removing the node labels, a bracketed sentence ((This) ((is) ((a) (pen)))) is obtained. Such a bracketed sentence (equivalent to a binary tree) can be used to produce constraints. If IST-ITG is applied, the number of word orderings in $N = 4$ is reduced to 8, down from 22 with ITG. For example, for the source-side bracketed tree $((f_1 f_2)(f_3 f_4))$, eight target sequences $[e_1, e_2, e_3, e_4]$, $[e_2, e_1, e_3, e_4]$, $[e_1, e_2, e_4, e_3]$, $[e_2, e_1, e_4, e_3]$, $[e_3, e_4, e_1, e_2]$, $[e_3, e_4, e_2, e_1]$, $[e_4, e_3, e_1, e_2]$, and $[e_4, e_3, e_2, e_1]$ are accepted. For the source-side bracketed tree $((((f_1 f_2)f_3)f_4)$, eight sequences $[e_1, e_2, e_3, e_4]$, $[e_2, e_1, e_3, e_4]$, $[e_3, e_1, e_2, e_4]$, $[e_3, e_1, e_2, e_4]$, $[e_4, e_1, e_2, e_3]$, $[e_4, e_2, e_1, e_3]$, $[e_4, e_3, e_1, e_2]$, and

$[e_4, e_3, e_2, e_1]$ are accepted. Generally, the number of word orderings is reduced to 2^{N-1} . Table 1 shows the number of word orderings in a target word sequence for each N with ITG, IST-ITG, and no constraints.

Table 1: Number of word orderings in each type of constraint

N	IST-ITG	ITG	No Constraint
1	1	1	1
2	2	2	2
3	4	6	6
4	8	22	24
5	16	90	120
6	32	394	720
7	64	1806	5040
8	128	8558	40320
9	256	41586	362880
10	512	206098	3628800
15	16384	745387038	1307674368000

2.5 Extension to Non-binary Tree

In the above subsection, a source binary tree was assumed in order to perform IST-ITG. However, parsing results sometimes are not binary trees. In this case, some tree nodes have more than two branches. For a non-binary node, any reordering of branches is allowed. In a non-binary tree $(f_1(f_2 f_3 f_4))$, twelve target-side sequences $[e_1, e_2, e_3, e_4]$, $[e_1, e_2, e_4, e_3]$, $[e_1, e_3, e_2, e_4]$, $[e_1, e_3, e_4, e_2]$, $[e_1, e_4, e_2, e_3]$, $[e_1, e_4, e_3, e_2]$, $[e_2, e_3, e_4, e_1]$, $[e_2, e_4, e_3, e_1]$, $[e_3, e_2, e_4, e_1]$, $[e_3, e_4, e_2, e_1]$, $[e_4, e_2, e_3, e_1]$, and $[e_4, e_3, e_2, e_1]$ are allowed. For nodes that have more than three branches, the original ITG constraints are locally applied. Therefore, for a non-binary tree $(f_1(f_2 f_3 f_4 f_5))$, $22 \times 2 = 44$ word orderings are allowed in the target-side and represented by the following formula.

$$\prod_{i=1}^n (S_{Bi}) \quad (3)$$

where S_k represents the number of combinations from the original ITG constraints for $N = k$ and B_i represents the number of branches at the i th node.

3 IST-ITG in Phrase-based SMT

In the above section, we described each constraint in the case of a one-to-one word-alignment. In this section, we consider phrase-based models. When a phrase-based model is used, each constraint must be extended. For IBM constraints, equation (2) is rewritten using phrase Pe_n instead of word e_n as follows:

$$d_i = \text{abs}(\text{last_position}(Pe_{i-1}) + 1 - \text{first_position}(Pe_i)) \quad (4)$$

where $\text{last_position}(Pe_n)$ represents the position of the last word in n th phrase, and $\text{first_position}(Pe_n)$ represents the position of the first word in n th phrase. The lexical reordering model and ITG constraints can be extended by changing the model (or constraint) unit from "word" to "phrase". However, in IST-ITG, "word" must be used for the constraint unit since the parse (bracketed tree) unit is in "words". To absorb different units between translation models and IST-ITG constraints, we investigated a new limitation for word ordering as follows.

- Word ordering that destroys a phrase is not allowed.

When this limitation is applied, the translated word ordering is obtained from the bracketed source sentence tree by reordering the nodes in the tree, the same as for one-to-one word-alignment. According to this limitation, the following nodes cannot be reordered. If a sub-tree with root node X includes part of a phrase ph , node X cannot be reordered. Consider the source bracketed source tree $((e_a e_b e_c) ((e_d e_e) (e_f e_g)))$, in which $e_b e_c$, and e_d form a phrase e_{ph} as in Figure 1. Node 1 cannot be reordered since part of the phrase $e_b e_c$ is included in node 1's sub-tree. For the same reason, node 2 and 4 cannot be reordered. Node 3 can be reordered since the sub-tree does not include the phrase (target sequence $[f_a f_{ph} f_e f_g f_f]$ is obtained by rotating node 3). Node 5 also can be reordered since it includes the whole phrase (target sequence $[f_g f_f f_e f_{ph} f_a]$ is obtained by rotating node 5). If node 2 is reordered, phrase ph is split into two parts, and translated in two parts in the target sentence. It is inconsistent

with the condition that phrase-to-phrase alignment is one-to-one. As a result, only the target sequences $[f_a f_{ph} f_e f_f f_g]$, $[f_a f_{ph} f_e f_g f_f]$, $[f_g f_f f_e f_{ph} f_a]$, and $[f_f f_g f_e f_{ph} f_a]$ are allowed. Here, f_{ph} represents an equivalent phrase in the translation for e_{ph} .

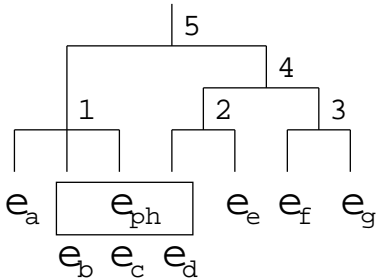


Figure 1: Example sentence tree with a phrase

4 Decoding with IST-ITG Constraints

In this section, we describe a one-pass decoding algorithm that uses IST-ITG constraints in the decoder. The translation target sentence is sequentially generated from left (sentence head) to right (sentence tail). To introduce the IST-ITG constraints into a decoder, the target candidate must be checked whether it satisfies the IST-ITG constraints or not whenever a new phrase is selected to extend a target candidate.

To explain this checking algorithm, we categorized source sub-trees into four types **UNTRANSLATED**, **TRANSLATED**, **TRANSLATING**, and **NG** (no good) as follows:

- If a sub-tree consists of only leaf word nodes, and all leaf words are not yet translated, this sub-tree is defined as **UNTRANSLATED**.
- If a sub-tree consists of only **UNTRANSLATED** sub-trees, this sub-tree is also **UNTRANSLATED**.
- If a sub-tree consists of only leaf word nodes, and all leaf words are already translated, this sub-tree is defined as **TRANSLATED**.
- If a sub-tree consists of only **TRANSLATED** sub-trees, this sub-tree is also **TRANSLATED**.
- If a sub-tree consists of only leaf word nodes with both translated and untranslated words, this sub-tree is defined as **TRANSLATING**.
- If a sub-tree consists of both **TRANSLATED** and **UNTRANSLATED** sub-trees, this sub-tree is **TRANSLATING**.
- If a sub-tree includes only one **TRANSLATING** sub-tree and any number (including zero) of **TRANSLATED** and **UNTRANSLATED** sub-trees, this sub-tree is **TRANSLATING**.
- If a sub-tree includes more than one **TRANSLATING** sub-tree, this sub-tree is **NG**.
- If a sub-tree includes **NG** sub-tree, this sub-tree is also **NG**.

If a translation candidate includes **TRANSLATING** sub-tree t , t must become **TRANSLATED** before anything else can happen. Given sub-tree $((ab)c)$, a is translated, b and c are not yet translated. In this case, b must be translated before c . If c is translated before b , the target word order becomes ACB . This word order does not satisfy the IST-ITG constraints. For the same reason, a candidate that includes an **NG** sub-tree does not satisfy the IST-ITG constraints. The checking algorithm for IST-ITG constraints is as follows.

1. For old translation candidates, the smallest **TRANSLATING** sub-tree t and its untranslated part u are calculated.
2. When a new target phrase f_{ph} is generated, the source phrase e_{ph} and untranslated part u calculated in above step are compared. If e_{ph} does not include and is not included in u , the new candidate is rejected. For example, in Figure 1, only source word e_a is already translated. The smallest **TRANSLATING** sub-tree is 1 and its untranslated part u is $[e_b e_c]$. In this case, phrases containing $[e_b]$, $[e_c]$, or $[e_b e_c]$ are accepted since these are included in u . Phrases $[e_b e_c e_d]$ or $[e_b e_c e_d e_e]$ are also accepted since these include u .
3. If a new candidate includes **NG** sub-trees, this candidate is rejected.

5 Experiments

5.1 Evaluation Measures

We evaluated the proposed method using four evaluation measures, BLEU (Papineni et al., 2002), NIST (Doddington 2002), WER(word error rate), and PER(position independent word error rate). Before discussing the evaluation, the characteristics of each one are analyzed.

- BLEU: This evaluation measure takes into account middle range word order, but does not take into account global word order. When the translation result is $[w_1, w_2, \dots, w_{j-1}, X, w_{j+1}, \dots, w_n]$ for reference translation $[w_1, w_2, \dots, w_n]$, both WER and BLEU scores will be high. For a translation result $[w_{j+1}, \dots, w_n, X, w_1, w_2, \dots, w_{j-1}]$, the BLEU score will be the same as the previous result since BLEU only takes into account 4grams. However, the WER score will be zero since global word positions are taken into account. Therefore, the effectiveness of the proposed method using BLEU is less than that of using WER.
- NIST: This evaluation measure only takes into account n-grams like BLEU. However, importance of higher order n-grams are less than BLEU. Therefore, the effectiveness of the proposed method using NIST will be less than that of using BLEU.
- WER: This evaluation measure takes into account not only local but also global word order, and is the most suitable for evaluating our method.
- PER: With this evaluation measure, we are almost incapable of considering word order. Therefore, our proposed method would seem to offer no improvement in this evaluation measure.

5.2 English and Japanese Patent Corpus Experiments

First, we conducted experiments on English and Japanese patent translations. Details of the experimental corpus are shown in Table 2. This corpus is

created by automatic sentence alignment (Uchiyama 2003). The first nine hundred sentence pairs with the best alignment scores were used as the evaluation data (single reference) and the next thousand sentence pairs were used as the development data. This corpus is a subset of the training corpus that will be used in the NTCIR-7 Workshop patent translation track.

Table 2: E-J patent corpus

	# of sent.	Total words	# of entries
E/J Train	1.8M	60M/64M	188K/118K
E/J Dev	916	30K/32K	4,072/3,646
E/J Eval	899	29K/32K	3,967/3,682

5.2.1 English-to-Japanese Translation

The translation direction of the first experiment was English-to-Japanese (E-J). For phrase-based translation model training, we used the GIZA++ toolkit (Och et al., 2003). For language model training, the SRI language model tool kit (Stolcke 2002) was used. The language model type was word 5-gram smoothed by Kneser-Ney discounting (Kneser 1995). For tuning of decoder parameters, we conducted minimum error training (Och 2003) with respect to the BLEU score using 916 development sentence pairs. For extraction of source sentence tree structure, we used the Charniak parser (Charniak 2000). We used Chasen for segmentation of the Japanese. The numbers of entries in the language models were 0.1 M, 2.1 M, 4.3 M, 6.2 M, and 6.9 M for 1, 2, 3, 4, and 5grams respectively. The number of entries in the phrase-table was 76 M. For decoding, we used an in-house decoder that is a close relative to the Moses decoder. The performance of this decoder was configured to be the same as Moses. Another conditions are the same as the default conditions of Moses decoder.

In the previous work (Zens et al., 2003, 2004), an IBM constraints and an ITG constraints are compared. In these experiments, a lexical reordering model, the proposed IST-ITC, and combinations of these are added as comparison targets. The combination of constraints in these experiments is as follows.

1. Monotone: Monotone translation (no reordering).
2. No constraints: There were no constraints for word reordering. Any word order was allowed without penalty.
3. IBM: IBM constraints without distortion limit.
4. ITG: ITG constraints.
5. IBM+ITG: Both IBM and ITG constraints were used at the same time.
6. IBM+LR: Both IBM constraints and lexical reordering model.
7. IST: Only the proposed IST-ITC constraints.
8. IBM+IST: Both IBM and IST-ITC constraints.
9. IBM+LR+IST: IBM constraints, Lexical reordering model, and IST-ITG constraints were used at the same time.

Table 3 shows the following experimental results. In comparing the original ITG constraints (ITG) with the proposed IST-ITG (IST) method, the improvement in BLEU was 2.67 points, and in WER was 5.39%. WER had the largest improvement, next was BLEU. This particular improvement order was the same as in the previous subsection. The large improvement of WER helped us confirm the effectiveness of the proposed method for global word ordering. When IBM constraints were used at the same time (IBM+ITG and IBM+IST), the BLEU score improved by 1.57 points and WER improved by 4.63%. When the lexical reordering model was used at the same time (IBM+LR and IBM+LR+IST), BLEU improved by 1.03 points and WER improved by 5.12%. The lexical reordering model fixed phrase position for the monotone and swap categories, but did not fix phrase position for the discontinuous category. IST-ITG fixed phrase position for the discontinuous category, even though it did not assign a probability. Combinations of the lexical reordering model and IST-ITG resulted in a better WER than with both IBM+LR and IBM+IST since both position and probability could be assigned for the discontinuous category.

Table 3: Evaluation results in E-J patent translation

	BLEU	NIST	WER	PER
Monotone	24.91	6.95	79.97	42.02
No constraint	26.83	7.19	81.10	39.52
IBM	28.35	7.29	78.35	39.25
ITG	27.59	7.26	80.29	39.15
IBM+ITG	28.50	7.30	78.01	39.29
IBM+LR	31.17	7.50	76.30	38.61
IST	30.26	7.41	74.90	38.93
IBM+IST	30.07	7.41	73.38	39.05
IBM+LR+IST	32.20	7.61	71.18	38.15

5.2.2 Japanese-to-English Translation

Next, we conducted J-E translation experiments using the same corpus. The numbers of entries in the language models were 0.2 M, 3.1 M, 4.1 M, 5.7 M, and 5.9 M for 1, 2, 3, 4, and 5grams. The improvement. The number of entries in the phrase-table was 76 M. For parsing of Japanese, we used the dependency structure analyzer CaboCha. From the dependency structure, Japanese bracketed trees were generated. The combination of constraints in these experiments was the same as those of the E-J translation experiments.

Table 4 shows the translation results of sentence evaluation with the top five alignment scores. In comparing the original ITG constraints (ITG) with the proposed IST-ITG (IST), BLEU was improved by 1.21 points, and by 3.81% in WER. The largest improvement was in WER, and BLEU had the next largest. This particular improvement order of these evaluation measures was the same as that of the E-J translation experiments. When IBM constraints were used at the same time (IBM+ITG and IBM+IST), there was no improvement in BLEU, but WER improved by 3.89%. When the lexical reordering model was used at the same time (IBM+LR and IBM+LR+IST), there was also no improvement in BLEU, but WER improved by 4.47%. One possible reason for the small (or no) improvement in BLEU is the lower parsing accuracy of Japanese compared with that of the English. However, better the WER figure indicates that using IST-ITC constraints leads to better word order. In the Appendix, differences in the translation results for the first five

evaluation sentences between IBM+LR (Baseline:) and IBM+LR+IST (Proposed:) are shown.

Table 4: Evaluation results in J-E patent translation

	BLEU	NIST	WER	PER
Monotone	26.29	7.25	76.42	40.85
No constraint	26.20	7.18	81.41	40.76
IBM	27.87	7.34	78.16	39.94
ITG	27.01	7.24	80.43	40.50
IBM+ITG	28.16	7.35	78.04	40.07
IBM+LR	29.93	7.54	77.27	39.12
IST	28.32	7.31	76.62	40.67
IBM+IST	28.14	7.32	74.13	40.40
IBM+LR+IST	29.77	7.50	72.80	39.73

5.3 NIST MT08 English-to-Chinese Translation Experiments

Next, we conducted English-to-Chinese (E-C) newspaper translation experiments for different language pairs. The training and evaluation corpora were used in the NIST MT08 evaluation campaign English-to-Chinese translation track. For the translation model training, we used 6.2M bilingual sentences. For the language model training, we used 20.1M sentences. A development set with 1,664 sentences was used as evaluation data in the Chinese-to-English translation track in the NIST MT07 evaluation campaign. A single reference was used in the development set. The evaluation set with 1,859 sentences is the same as MT08’s evaluation data, with 4 references. Model training and decoding conditions were the same as those in the E-J experiments. In both baseline and proposed condition, IBM constraints and lexical reordering model were used at the same time. Therefore, the baseline conditions correspond to the IBM+LR condition in the J-E experiments, the proposed conditions correspond to the IBM+LR+IST in the J-E experiments.

The evaluation unit was both the Chinese character and word as defined by the PKU corpus. As in the E-J experiments, the improvements in WER and CER (character error rate) were large. The improvements in WER, CER, word BLEU, and character BLEU were 5.3% (from 75.0% to 69.7%), 6.2% (from 74.1% to 67.9%), 2.2-points (from 21.0 to

23.2), and 1.8-points (from 35.2 to 37.0) respectively. We again demonstrated that the proposed method is effective (especially in WER) for multiple language pairs.

6 Conclusion

We proposed new word reordering constraints for PBSMT using source tree structure. The proposed IST-ITG constraints are extensions of the ITG constraints. In ITG constraints, the instance of the source-side tree is not taken into account. On the other hand, in IST-ITG constraints, the tree that is obtained by source sentence parsing is imposed on the decoding process. Therefore, IST-ITG constraints are stronger than those of the original ITG. For example, for four-word source sentences, IST-ITG constraints allow eight word orderings in a target sentence compared with twenty-two orderings under the original ITG constraints. IST-ITG constraints can be applied to a common decoder to determine a target sentence from one-pass without re-scoring. In our E-J patent translation experiments, the proposed method resulted in a 2.7-point improvement in BLEU and a 5.7% improvement in WER compared with those of the original ITG constraints. In this paper we have argued the WER is the most appropriate measure to gauge the effectiveness of our approach since it gives importance to the global word order. Our approach gave rise to considerable gains in term of WER in all of our experiments, indicating that a respectable improvement in global word order was achieved. The improvement could clearly be seen from visual inspection of the output, a few examples of which are presented in the following Appendix.

A Samples from the Translation of Japanese Patent into English

A.1 Sentence 1

Source: そして、ロータ 16 とステータ 15 の間に充填した液体の運動エネルギーが熱エネルギーに変換されて制動トルクを発生する。

Reference: and, the kinetic energy of the liquid filled between the rotor 16 and stator 15 is converted into thermal energy to thereby produce a brake torque.

Baseline: then, the rotor 16 and the kinetic energy is converted to thermal energy braking torque is gener-

ated between the liquid filled in the stator 15.

Proposed: then, the rotor 16 and between the stator 15, the liquid filling the kinetic energy is converted to thermal energy braking torque is generated.

A.2 Sentence 2

Source: 7はシール材であり、後述の加工ガス9のシールと移動ホルダ3のガイドを兼ねたものである。

Reference: a sealant 7, which serves as a seal for cutting gas 9, also serves as a guide for the moving holder 3.

Baseline: the seal and movement of the holder 3 also serves as a guide for the seal member 7 is a work gas 9.

Proposed: 7 denotes a seal material, which also serves as a guide for the working gas 9 described later seal and movement of the holder 3.

A.3 Sentence 3

Source: 次に、車両前方の交差点の信号機が赤であり、運転者は信号機が赤であることを認知しており、停止しようとして、アクセルからブレーキに踏み換えようとしている場合を例として説明する。

Reference: suppose that the red signal light of a traffic signal installed at a crossing situated ahead is on, the driver has recognized the red signal light, and the driver's foot is about to shift from the accelerator pedal to the brake pedal to stop the vehicle.

Baseline: next, the tread brake by the driver, the accelerator to be stopped from the traffic of recognizing traffic signals is "red" and the intersection ahead of the vehicle is red, it is described as an example.

Proposed: next, a case will be exemplified below so as to tread brakes from the accelerator to be stopped, and of recognizing traffic signals of red, the driver is "red" and is traffic light ahead of the vehicle.

A.4 Sentence 4

Source: さらに、被塗装物の表面を予め洗浄し、塗膜が付着しやすいように前処理、乾燥等の工程を必要とし、経済的でない。

Reference: in addition, this method is not economical because it requires special steps such as pre-washing of the substrate surface, pre-treatments for providing the substrate with adherability to a coating, a drying step and the like.

Baseline: further, the coating film is apt to be deposited on the surface of the object to be coated by washing and drying process is required, and the preliminary process advance not economical.

Proposed: further, to clean the surface of the object to be coated beforehand so as to facilitate the adhesion of the coating film preprocessing and drying process is required, and not economical.

A.5 Sentence 5

Source: 4は送油路を示し、本体1の中空部をもって構成してある。

Reference: an oil passage 4 is formed as a hollow portion in the main body 1.

Baseline: 4 is a hollow portion of the body 1 with an oil supply passage is shown.

Proposed: 4 is an oil supply passage, with a hollow portion of the main body 1.

References

- Adam L. Berger, Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, Andrew S. Kehler, and Robert L. Mercer, "Language translation apparatus and method of using context-based translation models," United States patent, patent number 5510981, April, 1996.
- Cabocha
<http://chasen.org/taku/software/cabocha/>
- Eugene Charniak, "A Maximum-Entropy-Inspired Parser," Proc. NAACL-2000, Seattle, Washington, pp.132-139, 2000.
- Chasen
<http://chasen-legacy.sourceforge.jp/>
- Yuan Ding, Martha Palmer, "Machine translation using probabilistic synchronous dependency insert grammars," Proc. ACL, Ann Arbor, pp. 541-548, 2005.
- George Doddington, "Automatic evaluation of machine translation quality using n-gram co-occurrence statistics," Proc. ARPA Workshop on Human Language Technology, San Diego, CA, 2002.
- Michel Galley, Jonathan Graehl, Kevin Knight, Daniel Marcu, Steve DeNeeffe, Wei Wang, Ignacio Thayer, "Scalable Inference and Training of Context-Rich Syntactic Models," Proc. ACL-COLING, Sydney Australia, pp. 961-968, 2006.
- Liang Huang, Kevin Knight, Aravind Joshi, "Statistical Syntax-Directed Translation with Extended Domain of Locality," Proc. AMTA, Massachusetts, 2006
- Reinhard Kneser, Hermann Ney, "Improved backing-off for m-gram language model," Proceedings of the IEEE International Conference of Acoustic, Speech, and Signal processing. Vol. 1, pp. 181-184, 1995.
- Yang Liu, Qun Liu, Shouxun Lin, "Tree-to-String Alignment Template for Statistical Machine Translation," Proc. ACL-COLING, Sydney Australia, pp. 609-616, 2006.
- Daniel Marcu, William Wong, "A phrase-based, joint probability model for statistical machine translation," Proc. EMNLP-2002, Philadelphia, pp.133-139, 2002. p. 127-133, 2003.
- Daniel Marcu, Wei Wang, Abdessamad Echihabi, Kevin Knight, "SPMT: Statistical Machine Translation with Syntactified Target Language Phrases," Proc. EMNLP-2006, Sydney Australia, pp. 44-52, 2006.
- Dan Melamed, "Statistical machine translation by parsing," Proc. ACL, Barcelona, pp. 653-660, 2004.
- Moses
<http://www.statmt.org/moses/>
- NIST MT08
<http://www.nist.gov/speech/tests/mt/2008/>
- NTCIR-7
<http://ntcir.nii.ac.jp/>
- Franz Josef Och, Hermann Ney, "A Systematic Comparison of Various Statistical Alignment Models," Computational Linguistics, No. 1, Vol. 29, pp. 19-51, 2003.
- Franz Josef Och, "Minimum error rate training for statistical machine translation," Proc. ACL, Sapporo Japan, pp. 160-167, 2003.
- Franz Josef Och, Hermann Ney, "The alignment template approach to statistical machine translation, Computational Linguistics, 30(4), pp417-449, 2004.
- Kishore Papineni, Salim Roukos, Todd Ward, Wei-Jing Zhu, "Bleu: a method for automatic evaluation of machine translation," Proc. ACL, Philadelphia PA, pp. 311-318, 2002.
- Chris Quirk, Arul Menezes, Colin Cherry, "Dependency treelet translation: Syntactically informed phrasal SMT," Proc. ACL, Ann Arbor, pp. 271-279, 2005.
- Andreas Stolcke, "SRILM - An Extensible Language Model Toolkit," Proc. ICSLP'02, Denver, pp. 901-904, 2002. <http://www.speech.sri.com/projects/srilm/>
- Christopher Tillmann, "A unigram orientation model for statistical machine translation," HLT-NAACL, Boston, pp. 101-104, 2004.
- Masao Uchiyama and Hitoshi Isahara, "Reliable Measures for Aligning Japanese-English News Articles and Sentences", Proc. ACL, Sapporo Japan, pp. 72-79, 2003.
- Dekai Wu, "Stochastic inversion transduction grammars, with application to segmentation, bracketing, and alignment of parallel corpora," In Proc. IJCAI, pp. 1328-1334, Montreal, 1995.
- Dekai Wu, "Stochastic inversion transduction grammars and bilingual parsing of parallel corpora," Computational Linguistics, 23(3), pp.377-403, 1997.
- Kenji Yamada, Kevin Knight, "A syntax-based statistical translation model," Proc. ACL, Hong Kong, pp. 523-530, 2000.
- Richard Zens, Hermann Ney, "A Comparative Study on Reordering Constraints in Statistical Machine Translation" Proc. ACL, Sapporo Japan, pp. 144-151, 2003.
- Richard Zens, Hermann Ney, Taro Watanabe, Ei-ichiro Sumita, "Reordering Constraints for Phrase-Based Statistical Machine Translation" Proc. Coling, Geneva, pp. 205-211, 2004.