IEICE TRANS. INF. & SYST., VOL.E97–D, NO.6 JUNE 2014

LETTER

Utilizing Global Syntactic Tree Features for Phrase Reordering

Yeon-Soo LEE[†], **Hyoung-Gyu LEE**[†], ***Nonmembers***, **Hae-Chang RIM**[†a)], ***Member***, ***and*** **Young-Sook HWANG**[††b)], ***Nonmember***

SUMMARY In phrase-based statistical machine translation, long distance reordering problem is one of the most challenging issues when translating syntactically distant language pairs. In this paper, we propose a novel reordering model to solve this problem. In our model, reordering is affected by the overall structures of sentences such as listings, reduplications, and modifications as well as the relationships of adjacent phrases. To this end, we reflect global syntactic contexts including the parts that are not yet translated during the decoding process.

key words: *phrase reordering model, global syntactic tree features, phrase-based statistical machine translation*

1. Introduction

Since phrase-based statistical machine translation (PBSMT) is relatively simple, it can be easily applied to many language pairs. Also, its lexical translation coverage is very high because the translation unit can be any length of "phrase." However, in recent years, PBSMT has been also applied to grammatically distant language pairs such as English-Korean, English-Japanese, or English-Turkish, thus making its reordering problem more prominent. Original PBSMT reordering model is based on the distribution of distance from a previously aligned phrase. However, the distance-based model has limits in terms of accurate reordering discrimination. Many studies have been conducted to improve the distance-based model; however, long sentences such as newspaper articles are still often not properly translated, mainly due to reordering failures. The phrases in long sentences should be reordered as complex types or long distance movements. If long distance reordering fails, translation performance barely changes even if several local reorderings succeed.

Long distance movement is caused by the differences between languages in the way they syntactically organize sentences such as subject-object-verb (SOV)/subject-verb-object (SVO), head-initial/head-final, and special grammatical conditions. Therefore, recent studies attempt to reflect syntax information in PBSMT reordering. First, there are preprocessing methods that change source words into a target word order prior to the decoding step [1], [2]. For this purpose, manually or automatically established grammar conversion rules are used. The issues in these methods are related to automatic acquisition, coverage, and ambiguity of rules. Second, post-processing methods are presented in [3], [4]. These methods re-rank the n-best translation results by using syntactic features, or they re-organize the phrases of the result sentence. Third, there are soft-constraint methods that engage in reordering of decoding step [5]–[8]. It is difficult to integrate syntactic model into the PBSMT framework, since the syntactic boundary and phrase boundary of the PBSMT do not match. However, the methods have more potential to improve due to its ability to control the decoding process. The "Lexicalized reordering model" the most widely used method exploit lexical features between two phrases [5], [6]. In [7], they takes advantage of syntactic cohesion characteristics. In recent years, reordering may be regarded as a classification problem and implemented as a discriminative model. In the [8]'s research, the source's syntactic tree information is reflected in the word lattice in order to decide the visiting order of words. However, they still considered the reordering problem as a distance based movement of word and used the tree features insufficiently to resolve the long distance reordering problem. Nevertheless, since grammatical differences better explain the reordering phenomena, the performance was improved to a certain extent. Our method belongs to these soft-constraint approaches.

Util now, little attention has been given to questions such as "What range of syntactic context should be considered?" or "How is syntax and long distance reordering related?" If there is ambiguity in discriminating between local reordering and long distance reordering, then incorrect hypotheses remain in the hypothesis space even though syntactic information is used. If the incorrect hypothesis has a high language model score, finally translation errors are generated. It is worthwhile to exam the reordering problem more closely. In this paper, we focus on long distance reordering as well as local reordering. We refer to the ambiguity of using syntactic information, investigate global syntactic features, and propose a new reordering model to support these features.

Manuscript received September 30, 2013.
Manuscript revised January 27, 2014.

[†]The authors are with the Dept. of Computer and Radio Communications Engineering, Korea University, Seoul, Korea.
[††]The author is with Platform R&D Office, SK Planet Co. Ltd., Seoul, Korea.
a) E-mail: rim@nlp.korea.ac.kr (Corresponding author)
b) E-mail: youngsook.hwang@sk.com

DOI: 10.1587/transinf.E97.D.1694

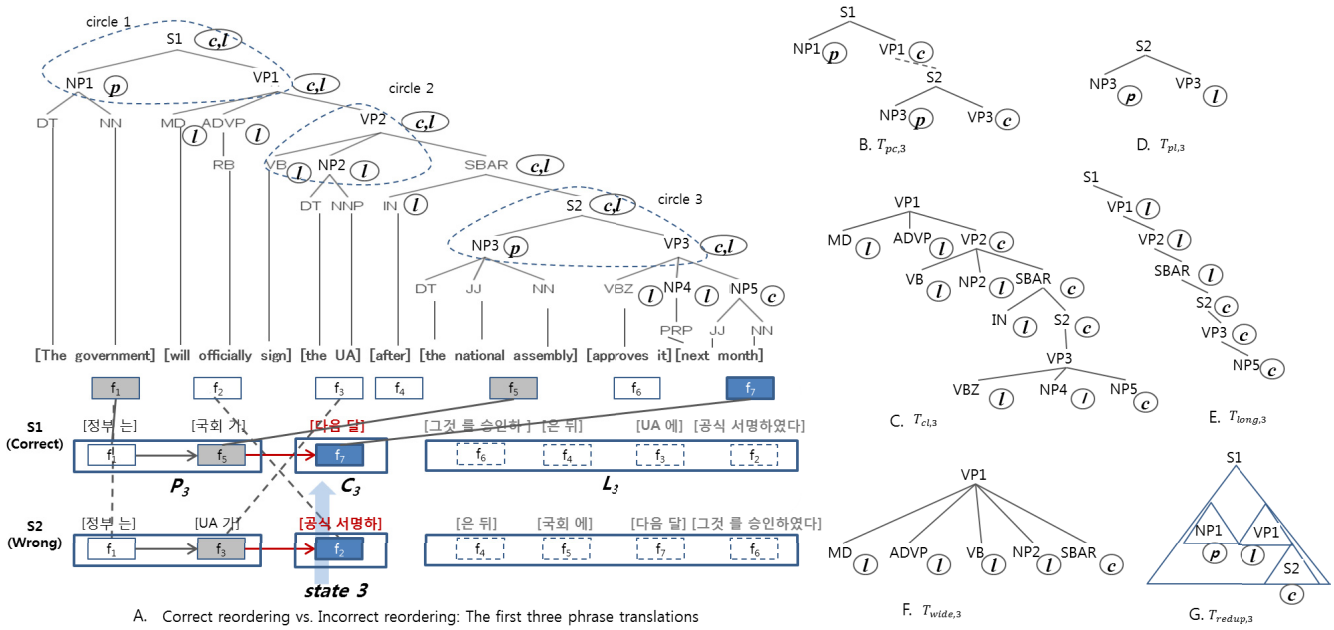

Fig. 1 Long distance reordering and global syntactic context.

2. Syntactically Featured Phrase Reordering Model

2.1 Long Distance Reordering & Syntax

First, we introduce an English-Korean translation example with complex reordering in Fig. 1. S_1 and S_2 are two of the several hypotheses in the decoding process. The decoder segments the source sentence f into phrases f_1, f_2, $f_3 \ldots$, and reorders them. Hypothesis S_1 is correct but S_2 is not. S_2 seems to be correct in terms of lexical translation; however, the meaning is completely different from the original meaning because it is translated in the wrong order, It is translated as, "The government approves it after the UA officially sign the national assembly next month." Most of the previous studies analyzed a junction point in the parse tree between a previous phrase and a current phrase. For example, when S_1 translates f_5 after f_1, the junction point on the parse tree is circle 1 the treelet (S1, NP1, VP1). Furthermore, at selection (hereafter referred to as "state") 3 of S_1, f_7 meets f_5 at circle 3. NP3 precedes VP3 under S2 at the target sentence. At state 3 of S_2, f_3 meets f_2 at circle 2. Under VP2, NP2 is translated before the phrase that includes VB. If the SOV characteristic of the target language is considered, all the above interpretations are correct. In other words, during the first three states, if we consider the local space (the junction points) on the parse tree, we cannot distinguish correct long distance movement from incorrect local reordering. Syntactic knowledge states, "The modification clause should be translated before the main verb and the subject is the first in that clause according to the target syntax." If this is reflected, S_2 receives a very low score and disappears early from the hypothesis stack. However, this information is scattered on the parts that are not yet translated but related to the current phrase and is based on the structural complexity of the sentence. To eliminate the S_2 in a limited search space, the information should be detected and reflected early.

2.2 Discriminative Phrase Precedence Model

Before we use the parse tree context, we first define a discriminative reordering model that estimates the precedence of phrases. The source sentence f consists of N phrases $\langle f_1, f_2, \ldots, f_N \rangle$ and the reordered sequence at target is $O = \langle o_1, o_2, \ldots, o_N \rangle$. Then f_{o_j} represents the jth translated source phrase. The source phrases can be divided into three groups when f_{o_j} is selected at the jth state. - P_j is a set of previously translated phrases, C_j is a current phrase, and L_j is a set of phrases that have not yet been translated. For example, in Fig. 1, when f_7 is translated after f_1 and f_5, P_j is $\{f_1, f_5\}$, C_j is $\{f_7\}$ and L_j is $\{f_2, f_3, f_4, f_6\}$. Our model estimates the probability of the precedence relationship at state j, given the source tree T_f.

$$p(C_j, L_j | T_f, P_j),$$

$$P_j = \left\{ f_{o_1}, f_{o_2} \ldots, f_{o_{j-1}} \right\}, C_j = \left\{ f_{o_j} \right\}, L_j = \{ f_{o_m} | j < m \leq N \} \tag{1}$$

The sentence level reordering probability is represented by Eq. (2) and it can be rewritten as Eq. (3) in maximum entropy style.

$$\max \prod_{j=1}^{N} p(C_j, L_j | T_f, P_j) \tag{2}$$

$$= \max \prod_{j=1}^{N} \frac{1}{Z(T_f, P_j)} exp\Big(\sum_k \lambda_k \phi_k (P_j, C_j, L_j, T_f) \Big) \tag{3}$$

Table 1 Feature category.

local	T_{pc}-the relationship between the previously translated part and the current part
uncovered	T_{cl}-the relationship between the current translated part and the part that will be translated in the future T_{pl}-the relationship between the previously translated part and the part that will be translated in the future
global	T_{long} - The longest path in the parse tree T_{wide} - The widest siblings T_{redup} - The nested sentence structure

Here, ϕ_k is the binary valued feature function, λ_k is the weight of the ϕ_k, and $Z(T_f, P_j)$ represents the normalization factor.

2.3 Syntactic Mapping

In order to analyze the phrase precedence relationship in terms of syntax, the relationship is mapped to the source parse tree. We can attach one of the following three tags to each node in the tree: *p*, *c*, or *l*.

- *p*: all the words dominated by the node are translated
- *c*: one of the words dominated by the node belongs to C_j
- *l*: one of the words dominated by the node belongs to L_j

The tag indicates whether each node is associated with any of the three sets that were mentioned above: P_j, C_j, and L_j. Figure 1 shows an example of the state 3 of S_1. For example, tag *p* is attached to NP1 by which all the dominated nodes are translated and all the ancestors of the current phrase "the month" have tag *c*. Also, tag *l* is attached to all the ancestors of phrases that are not translated. We extract the sub-trees that represent the global context, and they are classified in Table 1. In Fig. 1, when f_7 is selected and translated, B, C and D represent the sub-tree features T_{pc}, T_{cl}, and T_{pl}, respectively. Also shown in Fig. 1, the global sub-tree features T_{long}, T_{wide} and T_{redup} are represented by E, F, and G respectively. T_{long} features indicate which part of the longest path in the parse tree is currently translated. For the T_{long} and T_{wide} features, if a parent node has any child node attached to tag *p* or *c*, the parent node is also attached to tag *p* or *c* respectively. To extract T_{wide}, we flatten the source parse tree; in other words, we integrate the parent and grand-parent that have the same head word. On the flattened tree, the parent-child treelet with the most siblings is T_{wide}. The T_{redup} features indicate the nested structure and the current part in the structure.

In order to use the sub-trees as the maximum entropy features, the trees are divided into parent-child treelets. Also we assume independence between the treelets. The above sub-tree features can be represented as a set of treelet *t*:

$$T_{u,j} = \begin{cases} \{t \mid t = (u, pr, c_1, c_2, tag_{c_1}, tag_{c_2}, w)\} \\ \qquad\qquad \text{if } u \in \{pc, cl, pl\} \\ \{t \mid t = (u, n_1, n_2, \ldots tag_{n_1}, tag_{n_2}, \ldots)\} \\ \qquad\qquad \text{if } u \in \{long, wide, redup\} \end{cases}$$

where *u* is one of the sub-tree types. *pr*, c_1, and c_2 are parent, left child, and right child, respectively. The *tag* represents one of the following tags (described above): *p*, *c*, and *l*. The number of words dominated by the parent is represented by *w*. We differentiate the impact on the reordering according to the number of dominating words, even if the sub-tree shape is the same. *n* is a node in the parse tree. In practice, to reduce the number of features and to avoid over-fitting, similar kinds of phrase tags are grouped and the weight is divided into three types: *heavy*, *middle*, and *light*. As a result, Eq. (3) is rewritten as follows:

$$\max \prod_{j=1}^{n} \frac{1}{Z(T_f, P_j)} exp\Big(\sum_u \sum_{t \in T_{u,j}} \lambda_t \phi_t(P_j, C_j, L_j, T_f) \Big)$$

Table 2 The representation of sub-tree features.

	S_1 (Correct)	S_2 (Incorrect)
T_{pc}	(S,NP,VP,P,C,15) (S,NP,VP,P,C,7)	(S,NP,VP,P,C,15) (VP,MD,VP,C,P,12) (VP,ADVP,VP,C,P,1)
T_{cl}	(VP,MD,VP,L,C,13) (VP,ADVP,VP,L,C,1) (VP,VB,SBAR,L,C,9) ... (VP,NP,NP,L,C,3)	(VP,VB,SBAR,C,L,9)
T_{pl}	(S,NP,VP,P,L,15)	(VP,NP,SBAR,P,L,10)
T_{long}	(S,VP,VP,SBAR,S,VP,NP,L,L,L,C,C,C)	(S,VP,VP,SBAR,S,VP,NP,C,P,L,L,L,L)
T_{wide}	(VP,ADVP,VB,NP,SBAR,L,L,L,L,C)	(VP,ADVP,VB,NP,SBAR,C,C,C,P,L)
T_{redup}	(NP,VP,S,P,L,C)	(NP,VP,S,P,C,L)

Table 2 shows the sub-tree features in the *t* format. The "correct" column shows the extracted features at the state 3 of S_1, and the "incorrect" column shows the extracted features at the state 3 of S_2. In the incorrect case, we know through the trees T_{cl} and T_{pl} that the modified clause "will officially sign the UA" is translated before the modifying clause "after ..."; however, this order is wrong. Moreover, long distance reordering is needed because there are a large number of words in the modifying clause.

To train the above model, we must construct the correct positive and negative examples for the phrase reordering. Since this requires large costs, we construct a pseudo-answer set in the following way: First, we run the word alignment using GIZA++. From the word aligned parallel sentence, the phrases which are consecutively aligned words, can be C_j. With C_j as the midpoint, P_j is the part that is aligned forward than C_j and L_j is the part that is aligned backward. The negative example is generated by exchanging the P_j and the L_j for the same C_j. This method may not be accurate, but it has the advantage of generating a large amount of the training set. In this way, the positive set and the negative set each have an amount of 100,000 examples[†].

[†]To measure the accuracy of the pseudo-answer set, we sampled 100 examples from the training set (which contained 100,000 examples). For each reordering example, the correctness was evaluated manually. As a result, the accuracy of the positive set is 82% and the accuracy of the negative set is 87%.

3. Experiments

We tested our method on three different language pairs: English-to-Korean(E2K), Chinese-to-Korean(C2K), and English-to-Chinese(E2C). The first two language pairs are SVO-SOV pairs which require a great deal of long distance movements. The last pair is an SVO-SVO pair with the same word order; however, it also requires long distance reorderings. The corpus[†] for training and testing is shown in Table 3. For a language model, we used the SRI Language Modeling Toolkit to train a 5-gram language model on all the training (target) sentences. For parsing the input source sentences, we used the Stanford parser for English and Chinese. We used the SMT system, Moses [9], with default options for the baseline and implemented the proposed model by modifying the decoding module. BLEU [10] metric is used to measure the translation performance.

We measured the reordering accuracy using the following steps. First, we manually constructed word aligned data from 400 parallel sentences. Second, we automatically generated the positive and negative reordering hypothesis from the data in the same way we constructed the training set. Then we randomly selected 1,500 examples at three times from the total of 24,957 examples. We measured the accuracy of the three hypothesis sets by using the previously trained model and averaged the results. The reordering accuracy was evaluated only in the English-Korean set. Table 4 shows the results of distortion distance variation and the distance represents the minimum word level distance at target sentence between the previously aligned phrase and the current phrase. The baseline results "T_{pc} only" is produced by using only the precedence relationship with previously translated phrases. We can see that for long distance reordering, it is important to consider the uncovered part in advance. Although T_{long}, T_{wide}, and T_{redup} do not affect the short movement, they contribute to the accuracy of the long movements. Also, this result shows that considering only the relationship between the adjacent phrases (e.g., "T_{pc} only"), in the long distortion set, is insufficient.

Next, we evaluate the translation performance. The "Lexicalized" model uses lexical features. On the other hand, the "HPMT (Hierarchical reordering model)", a basic syntax-based model, is known for its good performance on syntactically different language pairs. Table 5 shows the results of the comparative experiment on sentence length and features in English-Korean translation. These results show that the proposed features are effective, especially for the long sentence translation. Since the distribution of the NIST12 set is uneven, we do not conduct the length variation experiment; however, we can see that the overall performance is improved by using our proposed global features. Table 6 shows the performance of translations of language pairs. In the case of English-Korean and Chinese-Korean pairs, the proposed method performed better than HPMT. As for the English-Chinese pair, the performance of our method is comparable to that of HPMT.

[†]They are provided by SK Planet Co., Ltd. (http://www.skplanet.com/) only for research purposes. They are crawled over the various Korean on-line news sites and refined semi-automatically.

Table 3 Training and testing corpus (number of sentences).

	Training	Testing
E2K	SKP 491K[†], KUNLP 400K [11]	SKP 1,000[†], NIST12 3,074
C2K	SKP 477K[†]	SKP 1,000[†]
E2C	Hong Kong Parallel Text 1.8K, Chinese English News Magazine Parallel Text 166K	NIST08 1,859

Table 4 Reordering accuracy with distortion distance.

Features	♯ of features	Accuracy			
		~ 3	4 ~ 6	6 ~	All
T_{pc} only	6,532	84.24	79.26	73.39	76.68
T_{pc}+T_{cl}+T_{pl}	18,562	85.03	80.36	76.25	78.15
+T_{long}	12,258	85.03	80.94	78.10	80.11
+T_{wide}	28,473	85.75	81.17	78.78	81.17
+T_{redup}	2,965	85.78	81.42	81.99	82.34

Table 5 BLEU score with sentence length.

Feature	~ 5	6 ~ 14	15 ~	All	NIST12
Lexicalized	41.85	32.29	24.26	27.51	26.26
T_{pc} only	42.31	32.51	24.31	27.59	26.61
T_{pc}+T_{cl}+T_{pl}	42.54	33.83	25.69	28.96	27.48
+T_{long} +T_{wide}+T_{redup}	42.95	35.79	27.17	31.87	28.50

Table 6 BLEU score with language variation.

	English-Korean	English-Chinese	Chinese-Korean
Lexicalized	28.29	31.38	40.06
Hierarchical	31.58	32.29	41.53
Proposed	33.94	32.87	41.93

4. Conclusion

We explore more discriminative features by analyzing long distance reordering in the context of global structures. Also, we propose a new reordering model and a learning method to reflect the global structure context. Through this work, we found that to solve long distance reordering problems in structurally different language pairs such as English-Korean, it is necessary to reflect the high level context of syntax. In our future work, we will use the information of target sentences and integrate it into our model.

Acknowledgements

This research was supported by Basic Science Research Program and Next-Generation Information Computing Development Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT & Future Plannig (No.2011-0016878, No.2012M3C4A7033344).

References

[1] F. Xia and M. McCord, "Improving a statistical mt system with au-

tomatically learned rewrite patterns," Proc. Coling'04, p.508, 2004.
[2] H. Isozaki, K. Sudoh, H. Tsukada, and K. Duh, "Head finalization: A simple reordering rule for sov languages," Proc. Joint Fifth Workshop on Statistical Machine Translation and MetricsMATR, pp.244–251, 2010.
[3] F.J. Och, D. Gildea, S. Khudanpur, A. Sarkar, K. Yamada, A. Fraser, S. Kumar, L. Shen, D. Smith, K. Eng, et al., "A smorgasbord of features for statistical machine translation," Proc. HLT/NAACL'04, pp.161–168, 2004.
[4] K. Sudoh, X. Wu, K. Duh, H. Tsukada, and M. Nagata, "Post-ordering in statistical machine translation," Proc. MT Summit, 2011.
[5] C. Tillman, "A unigram orientation model for statistical machine translation," Proc. HLT/NAACL'04, 2004.
[6] P. Koehn, A. Axelrod, and A.B. Mayne, "Chris system description for the 2005 iwslt speech translation evaluation," Proc. IWSLT'05, 2005.
[7] C. Cherry, "Cohesive phrase-based decoding for statistical machine translation," Proc. ACL'08, 2008.
[8] N. Ge, "A direct syntax-driven reordering model for phrase-based machine translation," Proc. HLT/NAACL'10, 2010.
[9] P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, C. Dyer, O. Bojar, A. Constantin, and E. Herbst, "Moses: open source toolkit for statistical machine translation," Proc. ACL'07, 2007.
[10] K. Papineni, S. Roukos, T. Ward, and W.J. Zhu, "Bleu: a method for automatic evaluation of machine translation," Proc. ACL'02, 2002.
[11] G. Hong, C.H. Li, M. Zhou, and H.C. Rim, "An empirical study on web mining of parallel data," Proc. Coling'10, pp.474–482, 2010.