# Reverse Pre-reordering for SMT from Head-final to Head-initial Languages: A Case Study on Japanese-to-Khmer

Chenchen Ding[†], Vichet Chea[‡], Masao Utiyama[†], Eiichiro Sumita[†]
[†]Multilingual Translation Laboratory, National Institute of Information and Communications Technology, Japan
[†]Email: {chenchen.ding, mutiyama, eiichiro.sumita}@nict.go.jp
[‡]National Institute of Posts, Telecommunication and Information Communication Technology, Cambodia
[‡]Email: vichet.chea@niptict.edu.kh

***Abstract*—We conduct a case study on applying an approach called *reverse pre-reordering* (REV-REO) to Japanese-to-Khmer (JKm) statistical machine translation (SMT). The REV-REO is a simple but efficient approach originally designed for Japanese-to-English SMT to resolve the *word ordering* problem in translation. Theoretically, REV-REO can be applied on translation tasks from a head-final language to a head-initial language. As Khmer is a more typical head-initial language than English is, REV-REO should have effects on JKm SMT. We conducted experiments of JKm SMT using a standard phrase-based SMT system with a Japanese-Khmer parallel corpus of more than $100$ thousand sentence pairs. The performance of REV-REO is evaluated and investigated by three automatic measures. The experimental results suggest that established natural language processing techniques can work well on the understudied Khmer language.**

## I. Introduction

Statistical machine translation (SMT) is one of the most important research fields in natural language processing (NLP). SMT techniques have been well developed and widely applied in practice. Linguistic knowledge-free SMT frameworks, such as phrase-based (PB) SMT [1] and hierarchical phrase-based SMT (HIERO) [2], handle many translation tasks efficiently as long as sufficient training data prepared.

In this paper, we conduct a case study on applying an approach called *reverse pre-reordering* (REV-REO) on Japanese-to-Khmer (JKm) SMT. Our intention is to investigate whether an established, efficient technique performs well on SMT of Khmer, with a consideration of linguistic features of the Khmer language. The REV-REO is a simple rule-based approach originally designed for Japanese-to-English SMT. Because it is an rule-base approach, no training data are needed, which is especially suitable for an understudied language without enough resource for model training. Theoretically, REV-REO can be applied on translation tasks from a head-final language (i.e., Japanese, Korean) to a head-initial language (i.e., English, French). Because Khmer is a typical head-initial language, REV-REO should have effects on JKm SMT and the experimental results demonstrated it works.

More generally, the research on Khmer language processing has not been studied deeply, despite there have been many mature theories and efficient techniques developed in the NLP field nowadays. This paper, although just a case study of an simple approach, suggests that well-developed techniques can be directly used to process the Khmer language as long as considering the linguistic features of Khmer.

The paper is organized as follows. Section II contains related work on pre-reordering in SMT. In section III, we give a brief introduction and comparison of the linguistic *head-directionality*, i.e., the difference between head-final and head-initial languages. In section IV, we give the introduction of the approach of REV-REO. In section V, we conduct experiments and give discussions based on the experimental results. Section VI is the conclusion.

## II. Related Work on Pre-reordering in SMT

Word reordering is a problematic issue in SMT, when translating those language pairs with significantly different word orders. Among different lines of researches, pre-reordering has been widely applied in practice and still studied in recent researches [3], [4]. The process of a pre-reordering approach tries to arrange the word order on the source side into a target-like order before a standard SMT system applied.

Different pre-reordering approaches have been proposed. Language-independent pre-reordering approaches typically utilize designed statistical models to learn and conduct the reordering (e.g., [5]); language-dependent approaches typically utilize a trained parser with reordering rules, which can be either manually designed (e.g., [6]) or automatically extracted (e.g., [7] and [8]). Generally, a rule-based pre-reordering approach offers fast and crisp reordering, but parsing errors and a lack of robustness are drawbacks. In contrast, a statistical pre-reordering approach is more robust but also slower.

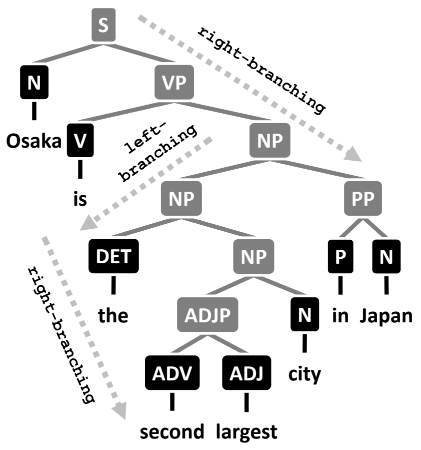

Fig. 1. An English syntactic tree.

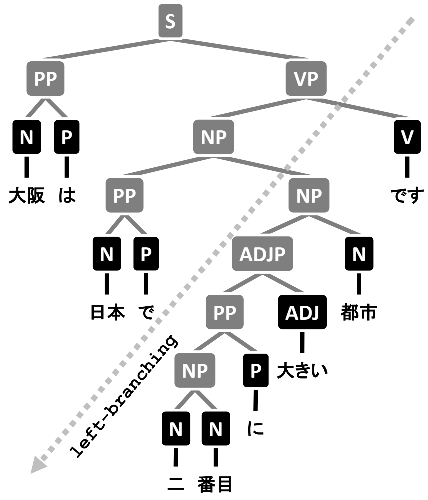

Fig. 2. A Japanese syntactic tree.

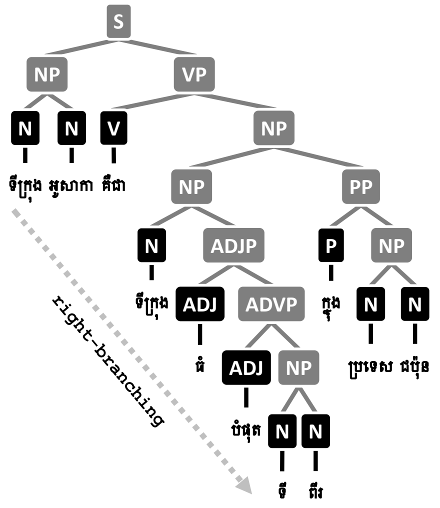

Fig. 3. A Khmer syntactic tree.

## III. Head-Final and Head-Initial Languages

In linguistics, the *head* of a phrase is the word that determines the syntactic type of the phrase it belongs to. For example, the phrase "*good day*" is a noun phrase because it has the noun *day* as its head; further, the phrase "*have a good day*" is a verb phrase because it has the verb *have* as its head.

Heads are crucial to establishing the direction of branching in syntactic structure of specific languages, i.e., head-initial phrases are right-branching and head-final phrases are left-branching. Some languages are consistently head-initial or head-final at all phrasal levels. English is considered to be strongly (not consistently) head-initial, while Japanese is an example of a language that is consistently head-final.

As to the Khmer language we focus on in this paper, it is primarily an analytic language, with no inflection, but using some derivation by means of prefixes. Since Khmer is an analytic language, word order is relatively fixed, as changes in word order are very likely to affect meaning. Khmer is generally a subject-verb-object language, using prepositions (as used in English) rather than postpositions (as used in Japanese). Correspondingly, Khmer is overwhelmingly head-initial, that is, modifiers follow the head they modify with in phrases. Topicalization is also typical in Khmer (as Japanese), whereby the topic of the sentence is placed at its beginning, and the rest of the sentence serves as a comment on the topic.

To offer an intuitive comparison, we show syntactic tree examples of English, Japanese and Khmer in Fig. 1, fig. 2, and Fig. 3, respectively. It can be observed that English has a head-initial (right-branching) tendency but not strictly. While Japanese is typical head-final (left-branching) and Khmer typical head-initial (right-branching).

Generally, the SMT between a head-final language and a head-initial language always suffers the word reordering problem because of the extremely different word orders. However, for language pairs with *consistent* head-final / head-initial properties, *consistent (and thus simple)* reordering approaches should handle the problem efficiently. So we consider the translation from Japanese, which is typically head-final, to Khmer, which is typically head-initial, will be an interesting task on which we test the simple but efficient REV-REO approach.

## IV. Reverse Pre-reordering

The REV-REO is originally proposed by [9] for the NTCIR-7 Japanese-to-English Patent MT translation task. The approach has been described in detail in the original papers, we here give an example in Fig. 4 to illustrate and explain the approach.

REV-REO focuses on only one specific Japanese morphemes: the topic-marker. Within the processing, the sequence before and after the topic-marker are totally reversed for source-side Japanese sentences to match up the head-initial target-side word order (i.e., Khmer in this paper). The word alignments of between original / reordered Japanese sentence and Khmer sentence are also shown in Fig. 4. Because the alignments are generated by the automatic word aligner `GIZA++`, there are several errors; however, the performance of REV-REO is impressive, that the Khmer and reordered Japanese sentences nearly have similar word orders after the REV-REO process.

For the implementation of REV-REO, an important point is to **avoid the reordering across punctuations**, otherwise the reordering will become excessive. We used four marks to

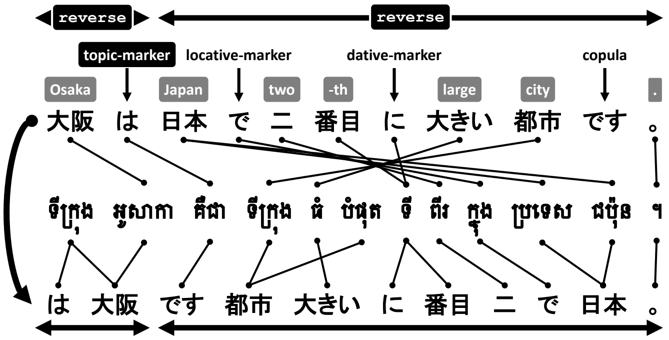
reverse
reverse
topic-marker
locative-marker
dative-marker
copula
Osaka
Japan
two
-th
large
city
.
大阪 は 日本 で 二 番目 に 大きい 都市 です 。
ទីក្រុង អូសាកា គឺជា ទីក្រុង ធំ បំផុត ទី ពីរ ក្នុង ប្រទេស ជប៉ុន ។
は 大阪 です 都市 大きい に 番目 二 で 日本 。

Fig. 4. REV-REO example for a Japanese sentence "*Osaka is the second largest city in Japan.*" The glosses and explanations of Japanese morphemes are illustrated over the original Japanese sentence in the top rank. The reordered Japanese sentence is placed in the bottom rank, where the parts before / after the topic-marker are reversed. The automatically generated word alignments of between the original / reordered Japanese sentence, and Khmer translation are also shown, form which the effect of REV-REO is obvious.

compose the punctuation set: `U+002C`[1], `U+FF0C`[2], `U+3001`[3], and `U+3002`[4]. For the Japanese topic-marker, which plays the key role of the approach, we did not judge it only by its surface form, but also referred to its part-of-speech tag.

## V. Experiment

We conducted translation experiment using the Japanese-Khmer version of *Basic Travel Expression Corpus* (BTEC), which is translated form the original English-Japanese version [10]. As the data mainly contain short, daily expressions, we select relately long sentences as test set for evluation. we used `MeCab`[5] (IPA dictionary) for Japanese word segmentation and an indoor, conditional random field based tool for Khmer word segmentation [11][6]. The details of the data sets we used are listed in Table I.

We used PB SMT and in `Moses`[7] [12] as a baseline SMT system. For word aligner, we used `GIZA++`[8] [13] with the default setting of `Moses`, and the *grow-diag-final-and* symmetrization heuristics [1]. The *max-phrase-length* was 7 and the reordering model was trained with using the *msd-bidirectional-fe* option. In decoding, we used the default setting of `Moses`' decoder, with different distortion-limit (DL) for comparison. The language model used in decoding was an interpolated modified Kneser-Ney discounting 5-gram model, trained on the Khmer side of the training set by `SRILM`[9] [14]. MERT [15] was used to tune the feature weights for the development sets to optimize the BLEU score [16] and the translation performance was evaluated on the test sets with the tuned weights. The decoding settings were identical for the development and test sets in our experiments. The REV-REO was applied consistently on the training, development, and test sets before the baseline PB SMT applied, in training and decoding phases.

TABLE I
Data sets used in experiments.

| Set | Sentences | J-Words | Km-Words |
|---:|---:|---:|---:|
| Training | $122,269$ | $1,139,317$ | $1,044,552$ |
| Development | $500$ | $16,117$ | $13,229$ |
| Test | $500$ | $16,059$ | $13,285$ |

We used three automatic measures to evaluate the performance of REV-REO on JKm SMT: Kendall's $\tau$ [6] on training set, BLEU score and RIBES [17] on test set. Kendall's $\tau$ measures the reordering performance on training set; BLEU score measures the precision of the N-gram matching rate between the SMT system output and corresponding reference translations on test set; and RIBES evaluates the accordance of word order on test set.

The Kendall's $\tau$ is calculated according to

$$\tau = 2\,\frac{\#\textit{increasing pairs}}{\#\textit{all pairs}} - 1, \qquad (1)$$

where the *pairs* here are the pairs of aligned words[10]. The range of this measure is in $[-1.0, +1.0]$. A larger value suggests the the source and target languages tend to have similar word order and a smaller valuer suggests different order. Specifically, $\tau = +1.0$ means the source and target languages have a totally identical word order and $\tau = -1.0$ means they have a totally reversed order. Fig. 5 gives four examples of the calculation of Kendall's $\tau$.

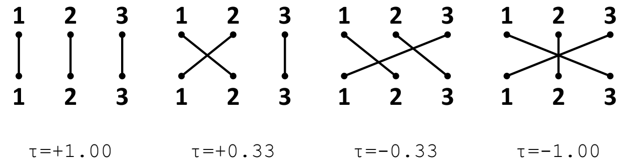

Fig. 5. Examples of Kendall's $\tau$. For all four cases, the #*all pairs* in Exp. 1 is three ($C_3^2$), from left to right the #*increasing pairs* in Exp. 1 is three, two ([1-2, 3-3] and [2-1, 3-3] are increasing), one (only [1-2, 2-3] is increasing), and zero, respectively.

[1] i.e., the ordinary comma.

[2] "fullwidth comma", the Chinese comma.

[3] "ideographic comma", the Japanese *tōten*.

[4] "ideographic full stop", the Japanese *kuten*.

[5] http://taku910.github.io/mecab/

[6] The referenced paper is expected to be accepted and appear in the the Second Annual Conference on Khmer Natural language Processing (KNLP 2015).

[7] http://www.statmt.org/moses/

[8] http://www.statmt.org/moses/giza/GIZA++.html

[9] http://www.speech.sri.com/projects/srilm/

[10] More detailed description and example can be found in the paper of [6]. In the calculations for Kendall's $\tau$, we used only the one-to-one aligned words from the GIZA++ alignment file as used in [6].

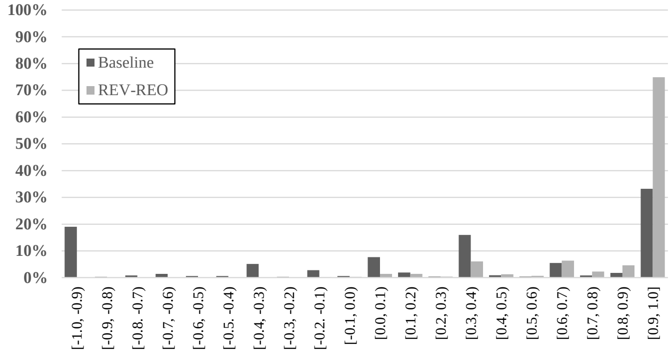

Fig. 6. Distribution of Kendall's $\tau$ on training set.

TABLE II
TRAINING SET AVERAGE KENDELL'S $\tau$.

| Baseline | REV-REO |
|---|---|
| 0.22 | 0.88 |

The distribution of Kendall's $\tau$ [6] on training set is shown in Fig. 6 and the averages are list in Table II, for baseline PB SMT and for REV-REO, respectively. The improvement on Kendall's $\tau$ brought by REV-REO is extremely obvious, that there are originally only around $30\%$ sentences in the training data have a $\tau$ over $0.9$ but REV-REO can increase it to over $70\%$. Consequently, the average value is improved greatly. The results suggest that the REV-REO actually reorder the Japanese sentences to Khmer-like word orders.

The evaluation results of the translation performance are list in Table III and Table IV, with different settings of distortion-limit. The REV-REO can give better results than the baseline system on both BLEU score and RIBES. As mentioned, the translation between head-final and head-initial languages requires very heavy reordering, so the best performance of the baseline PB SMT system can only be reached by a relative large distortion-limit (DL $= 15$), which means the reordering range covers $15$ words in translation. However, the translation performance can reach a mediocre BLEU score even with a small distortion-limit and can be further improved with the increasing of the distortion-limit setting. On the other hand, REV-REO have already reached its best RIBES using a small distortion-limit (DL $= 6$), which cannot be improved by larger distortion-limit settings. From the phenomena, the good reordering ability of the REV-REO can be observed.

TABLE III
TEST SET BLEU SCORES.

| DL | Baseline | REV-REO |
|---|---|---|
| 6 | 14.5 | 15.8 |
| 9 | 16.2 | 16.0 |
| 12 | 15.9 | 16.3 |
| 15 | **16.2** | **16.5** |
| 18 | **16.2** | 15.4 |
| $\infty$ | 16.0 | 15.9 |

TABLE IV
TEST SET RIBES.

| DL | Baseline | REV-REO |
|---|---|---|
| 6 | .626 | **.655** |
| 9 | .639 | .639 |
| 12 | .633 | .645 |
| 15 | **.638** | .635 |
| 18 | .623 | .619 |
| $\infty$ | .564 | .580 |

## VI. CONCLUSION

In this paper, we have applied the REV-REO, an approach originally designed for Japanese-to-English SMT to Japanese-to-Khmer SMT. In linguistic theories, the specific technique should work and our experimental results on three different automatic evaluation measures have given positive confirmations.

Although the NLP is an active research field nowadays, the research on Khmer language processing has not been studied systematically and insightfully. In this paper, we prove that well-developed techniques can be directly used to process the Khmer language as long as considering the typical head-initial feature of Khmer.

## REFERENCES

[1] P. Koehn, F. J. Och, and D. Marcu, "Statistical phrase-based translation," in *Proc. of HTL-NAACL*, 2003, pp. 48–54.

[2] D. Chiang, "Hierarchical phrase-based translation," *Computational Linguistics*, vol. 33, no. 2, pp. 201–228, 2007.

[3] A. de Gispert, G. Iglesias, and B. Byrne, "Fast and accurate preordering for SMT using neural networks," in *Proc. of NAACL-HLT*, 2015, pp. 1012–1017.

[4] S. Hoshino, Y. Miyao, K. Sudoh, K. Hayashi, and M. Nagata, "Discriminative preordering meets Kendall's Tau maximization," in *Proc. of ACL (Short Papers)*, 2015, pp. 139–144.

[5] G. Neubig, T. Watanabe, and S. Mori, "Inducing a discriminative parser to optimize machine translation reordering," in *Proc. of EMNLP-CoNLL*, 2012, pp. 843–853.

[6] H. Isozaki, K. Sudoh, H. Tsukada, and K. Duh, "HPSG-based preprocessing for English-to-Japanese translation," *ACM Transactions on Asian Language Information Processing*, vol. 11, no. 3, p. 8, 2012.

[7] D. Genzel, "Automatically learning source-side reordering rules for large scale machine translation," in *Proc. of COLING*, 2010, pp. 376–384.

[8] X. Wu, K. Sudoh, K. Duh, H. Tsukada, and M. Nagata, "Extracting preordering rules from predicate-argument structures," in *Proc. of IJCNLP*, 2011, pp. 29–37.

[9] J. Katz-Brown and M. Collins, "Syntactic reordering in preprocessing for Japanese → English translation: MIT system description for NTCIR-7 patent translation task," in *Proc. of NTCIR*, 2008, pp. 409–414.

[10] G. Kikui, "Creating corpora for speech-to-speech translation," in *Proc. of INTERSPEECH*, 2003, pp. 381–384.

[11] V. Chea, Ye Kyaw Thu, C. Ding, A. Finch, M. Utiyama, and S. Eiichiro, "Khmer word segmentation using conditional random field," in *Proc. of KNLP*, 2015.

[12] 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," in *Proc. of ACL*, 2007, pp. 177–180.

[13] F. J. Och and H. Ney, "A systematic comparison of various statistical alignment models," *Computational Linguistics*, vol. 29, no. 1, pp. 19–51, 2003.

[14] A. Stolcke, "SRILM–an extensible language modeling toolkit," in *Proc. of ICSLP 2002*, 2002, pp. 901–904.

[15] F. J. Och, "Minimum error rate training in statistical machine translation," in *Proc. of ACL*, 2003, pp. 160–167.

[16] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, “BLEU: a method for automatic evaluation of machine translation,” in *Proc. of ACL*, 2002, pp. 311–318.

[17] H. Isozaki, T. Hirao, K. Duh, K. Sudoh, and H. Tsukada, “Automatic evaluation of translation quality for distant language pairs,” in *Proc. of EMNLP*, 2010, pp. 944–952.