Neural Machine Translation with Reordering Embeddings

Kehai Chen, Rui Wang, Masao Utiyama, and Eiichiro Sumita
National Institute of Information and Communications Technology (NICT), Kyoto, Japan
{khchen, wangrui, mutiyama, eiichiro.sumita}@nict.go.jp

Abstract

The reordering model plays an important role in phrase-based statistical machine translation. However, there are few works that exploit the reordering information in neural machine translation. In this paper, we propose a reordering mechanism to learn the reordering embedding of a word based on its contextual information. These reordering embeddings are stacked together with self-attention networks to learn sentence representation for machine translation. The reordering mechanism can be easily integrated into both the encoder and the decoder in the Transformer translation system. Experimental results on WMT'14 English-to-German, NIST Chinese-to-English, and WAT ASPEC Japanese-to-English translation tasks demonstrate that the proposed methods can significantly improve the performance of the Transformer translation system.

1 Introduction

The reordering model plays an important role in phrase-based statistical machine translation (PBSMT), especially for translation between distant language pairs with large differences in word order, such as Chinese-to-English and Japanese-to-English translations (Galley and Manning, 2008; Goto et al., 2013). Typically, the traditional PBSMT learns large-scale reordering rules from parallel bilingual sentence pairs in advance to form a reordering model. This reordering model is then integrated into the translation decoding process to ensure a reasonable order of translations of the source words (Chiang, 2005; Xiong et al., 2006; Galley and Manning, 2008). In contrast to the explicit reordering model for PBSMT, the RNN-based NMT (Sutskever et al., 2014; Bahdanau et al., 2015) depends on neural networks to implicitly encode order dependencies between words in a sentence to generate a fluent translation. Inspired by a distortion method originating in SMT (Brown et al., 1993; Koehn et al., 2003; Al-Onaizan and Papineni, 2006), there is a quite recent preliminary exploration work for NMT (Zhang et al., 2017). They distorted the existing content-based attention by an additional position-based attention inside the fixed-size window, and reported a considerable improvement on the classical RNN-based NMT. This means that the word reordering information is also beneficial to the NMT.

The Transformer (Vaswani et al., 2017) translation system relies on self-attention networks (**SANs**), and has attracted growing interesting in the machine translation community. The Transformer generates an ordered sequence of positional embeddings by a positional encoding mechanism (Gehring et al., 2017a) to explicitly encode the order of dependencies between words in a sentence. The Transformer is adept at parallelizing of performing (multi-head) and stacking (multi-layer) SANs to learn the sentence representation to predict translation, and has delivered state-of-the-art performance on various translation tasks (Bojar et al., 2018; Marie et al., 2018). However, these positional embeddings focus on sequentially encoding order relations between words, and does not explicitly consider reordering information in a sentence, which may degrade the performance of Transformer translation systems. Thus, the reordering problem in NMT has not been studied extensively, especially in Transformer.

In this paper, we propose a reordering mechanism for the Transformer translation system. We dynamically penalize the given positional embedding of a word depending on its contextual information, thus generating a reordering embedding for each word. The reordering mechanism

* Corresponding author

Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 1787–1799
Florence, Italy, July 28 - August 2, 2019.

is then stacked together with the existing SANs to learn the final sentence representation with word reordering information. The proposed method can be easily integrated into both the encoder and the decoder in the Transformer. Experimental results on the WMT14 English-to-German, NIST Chinese-to-English, and WAT ASPEC Japanese-to-English translation tasks verify the effectiveness and universality of the proposed approach. This paper primarily makes the following contributions:

- We propose a reordering mechanism to learn the reordering embedding of a word based on its contextual information, and thus these learned reordering embeddings are added to the sentence representation for archiving reordering of words. To the best of our knowledge, this is the first work to introduce the reordering information to the Transformer translation system.

- The proposed reordering mechanism can be easily integrated into the Transformer to learn reordering-aware sentence representation for machine translation. The proposed translation models outperform the state-of-the-art NMT baselines systems with a similar number of parameters and achieve comparable results compared to NMT systems with much more parameters.

2 Related Work

2.1 Reordering Model for PBSMT

In PBSMT, there has been a substantial amount of research works about reordering model, which was used as a key component to ensure the generation of fluent target translation. Bisazza and Federico (2016) divided these reordering models into four groups:

Phrase orientation models (Tillman, 2004; Collins et al., 2005; Nagata et al., 2006; Zens and Ney, 2006; Galley and Manning, 2008; Cherry, 2013), simply known as lexicalized reordering models, predict whether the next translated source span should be placed on the right (monotone), the left (swap), or anywhere else (discontinuous) of the last translated one.

Jump models (Al-Onaizan and Papineni, 2006; Green et al., 2010) predict the direction and length of the jump that is performed between consecutively translated words or phrases, with the goal of better handling long-range reordering.

Source decoding sequence models (Feng et al., 2010, 2013) address this issue by directly modeling the reordered sequence of input words, as opposed to the reordering operations that generated it.

Operation sequence models are n-gram models that include lexical translation operations and reordering operations in a single generative story, thereby combining elements from the previous three model families (Durrani et al., 2011, 2013, 2014). Their method were further extended by source syntax information (Chen et al., 2017c, 2018b) to improve the performance of SMT.

Moreover, to address data sparsity (Guta et al., 2015) caused by a mass of reordering rules, Li et al. (2013, 2014) modeled ITG-based reordering rules in the translation by using neural networks. In particular, the NN-based reordering models can not only capture semantic similarity but also ITG reordering constraints (Wu, 1996, 1997) in the translation context. This neural network modeling method is further applied to capture reordering information and syntactic coherence.

2.2 Modeling Ordering for NMT

The attention-based NMT focused on neural networks themselves to implicitly capture order dependencies between words (Sutskever et al., 2014; Bahdanau et al., 2015; Wang et al., 2017a,b, 2018; Zhang et al., 2018). Coverage model can partially model the word order information (Tu et al., 2016; Mi et al., 2016). Inspired by a distortion method (Brown et al., 1993; Koehn et al., 2003; Al-Onaizan and Papineni, 2006) originated from SMT, Zhang et al. (2017) proposed an additional position-based attention to enable the existing content-based attention to attend to the source words regarding both semantic requirement and the word reordering penalty.

Pre-reordering, a pre-processing to make the source-side word orders close to those of the target side, has been proven very helpful for the SMT in improving translation quality. Moreover, neural networks were used to pre-reorder the source-side word orders close to those of the target side (Du and Way, 2017; Zhao et al., 2018b; Kawara et al., 2018), and thus were input to the existing RNN-based NMT for improving the performance of translations. Du and Way (2017)

and Kawara et al. (2018) reported that the pre-reordering method had an negative impact on the NMT for the ASPEC JA-EN translation task. In particular, Kawara et al. (2018) assumed that one reason is the isolation between pre-ordering and NMT models, where both models are trained using independent optimization functions.

In addition, several research works have been proposed to explicitly introduce syntax structure into the RNN-based NMT for encoding syntax ordering dependencies into sentence representations (Eriguchi et al., 2016; Li et al., 2017; Chen et al., 2017a,b; Wang et al., 2017b; Chen et al., 2018a). Recently, the neural Transformer translation system (Vaswani et al., 2017), which relies solely on self-attention networks, used a fixed order sequence of positional embeddings to encode order dependencies between words in a sentence.

3 Background

3.1 Positional Encoding Mechanism

Transformer (Vaswani et al., 2017) typically uses a positional encoding mechanism to encode order dependencies between words in a sentence. Formally, given a embedding sequence of source sentence of length J, $\mathbf{X}$=$\{\mathbf{x}_1, \cdots, \mathbf{x}_J\}$, the positional embedding is computed based on the position of each word by Eq.(1):

$$\begin{aligned} \mathbf{pe}_{(j,2i)} &= \sin(j/10000^{2i/d_{model}}), \\ \mathbf{pe}_{(j,2i+1)} &= \cos(j/10000^{2i/d_{model}}), \end{aligned} \quad (1)$$

where j is the word's position index in the sentence and i is the number of dimensions of the position index. As a result, there is a sequence of positional embeddings:

$$\mathbf{PE} = \{\mathbf{pe}_1, \cdots, \mathbf{pe}_J\}. \quad (2)$$

Each $\mathbf{pe}_j$ is then added to the corresponding word embedding $\mathbf{x}_j$ as an combined embedding $\mathbf{v}_j$:

$$\mathbf{v}_j = \mathbf{x}_j + \mathbf{pe}_j. \quad (3)$$

Finally, a sequence of embeddings $\{\mathbf{v}_1, \cdots, \mathbf{v}_J\}$ is the initialized sentence representation $\mathbf{H}^0$. Later, $\mathbf{H}^0$ will be input to the self-attention layer to learn the sentence representation.

3.2 Self-Attention Mechanism

Following the positional embedding layer, self-attention mechanism is used to learn sentence representation over the $\mathbf{H}^0$ obtained in the previous section. Generally, the self-attention mechanism is a stack of N identical layers in the Transformer architecture. Each identical layer consists of two sub-layers: self-attention network, and position-wise fully connected feed-forward network. A residual connection (He et al., 2016) is employed around each of two sub-layers, followed by layer normalization (Ba et al., 2016). Formally, the stack of learning the final sentence representation is organized as follows:

$$\begin{bmatrix} \overline{\mathbf{H}}^n = \text{LN}(\text{SelfAtt}^n(\mathbf{H}^{n-1}) + \mathbf{H}^{n-1}) \\ \mathbf{H}^n = \text{LN}(\text{FFN}^n(\overline{\mathbf{H}}^n) + \overline{\mathbf{H}}^n) \end{bmatrix}_N, \quad (4)$$

where $\text{SelfAtt}^n(\cdot)$, $\text{LN}(\cdot)$, and $\text{FFN}^n(\cdot)$ are self-attention network, layer normalization, and feed-forward network for the n-th identical layer, respectively. $[\cdots]_N$ denotes the stack of N identical layer. In the encoder and decoder of Transformer, $\text{SelfAtt}_n(\cdot)$ computes attention over the output $\mathbf{H}^{n-1}$ of the n-1 layer:

$$\text{SelfAtt}^n(\mathbf{H}^{n-1}) = \text{softmax}(\frac{\boldsymbol{QK}^\top}{\sqrt{d_k}})\mathbf{V}. \quad (5)$$

where $\{\mathbf{Q}, \mathbf{K}, \mathbf{V}\}$ are query, key and value vectors that are transformed from the input representations $\mathbf{H}^{n-1}$. d_k is the dimension size of the query and key vectors. As a result, the output of the N-th layer $\mathbf{H}^N$ is the final sentence representation for machine translation.

4 Reordering Mechanism

Intuitively, when a human translates a sentence, he or she often adjusts word orders based on the global meaning of the original sentence or its context, thus gaining one synonymous sentence which is easier to be understood and translated. It is thus clear that the reordering of a given word relies heavily on the global or contextual meaning of the sentence. Motivated by this, we use the word and its global contextual information of the sentence to gain a *Reordering Embedding* for each word (as shown in Figure 1), thus modeling the above human reordering process. The reordering mechanism is then stacked with the SAN layer to learn a reordering-aware sentence representation.

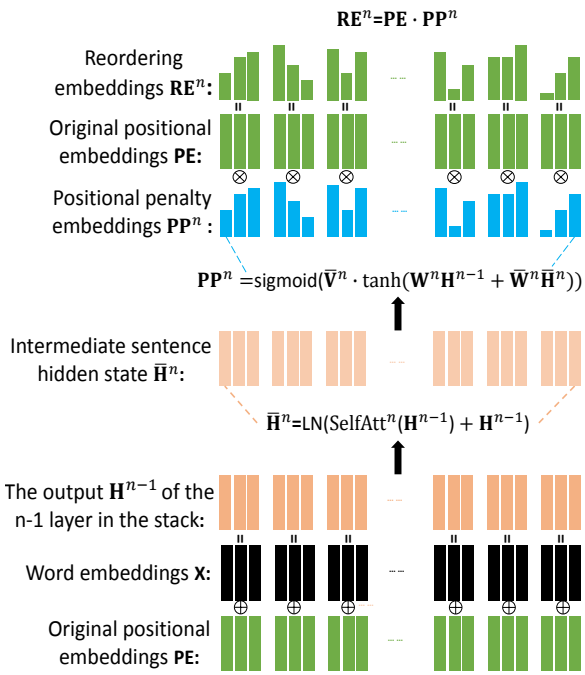

Figure 1: Learning reordering embeddings for the n-th layer in the stack.

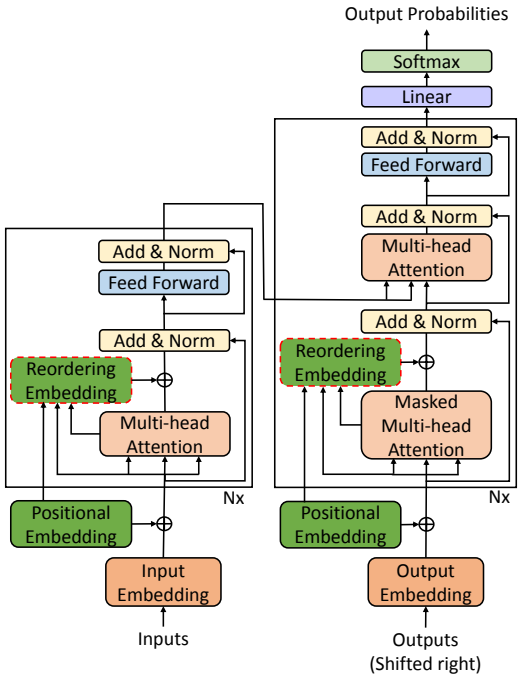

Figure 2: The architecture of Transformer with reordering embeddings.

4.1 Reordering Embeddings

To capture reordering information, we first learn a positional penalty vector based on the given word and its global context of the sentence. The positional penalty vector is then used to penalize the given positional embedding of the word to generate a new, reordering embedding. Finally, these reordering embeddings are added to the intermediate sentence representation to achieve the reordering of words. We divide the process into the following three steps:

Positional Penalty Vectors: The self-attention mechanism focuses on global dependencies between words to learn an intermediate sentence representation $\overline{\mathbf{H}}^n$, which is regarded as the expected global context of the sentence as reordered by a human translator. Therefore, given a sentence of J words, we use the output $\mathbf{H}^{n-1}$ of the previous layer in the stack together with the new intermediate global context representation $\overline{\mathbf{H}}^n$ to learn positional penalty vectors $\mathbf{PP}^n$ for the n-th layer of the stack $[\cdots]_N$:

$$\mathbf{PP}^n = \text{sigmoid}(\overline{\mathbf{V}}^n \cdot \tanh(\mathbf{W}^n \cdot \mathbf{H}^{n-1} + \overline{\mathbf{W}}^n \cdot \overline{\mathbf{H}}^n)), \quad (6)$$

where $\mathbf{W}^n \in R^{d_{model} \times d_{model}}$, $\overline{\mathbf{W}}^n \in R^{d_{model} \times d_{model}}$, and $\overline{\mathbf{V}}^n \in R^{d_{model} \times d_{model}}$ are the parameters of model. d_{model} is the dimension of the model. Each element of $\mathbf{PP}^n \in R^{J \times d_{model}}$ is a real value between zero and one.

Reordering Embeddings: $\mathbf{PP}^n$ is used to penalize the original positional embeddings **PE**:

$$\mathbf{RE}^n = \mathbf{PE} \cdot \mathbf{PP}^n, \quad (7)$$

where $\mathbf{RE}^n$ is called reordered embedding (**RE**) because each element of **PE** is multiplied by a probability between zero and one.

Achieving Reordering: The learned $\mathbf{RE}^n$ is further added to $\overline{\mathbf{H}}^n$ to achieve reordering operations for the current sentence hidden state $\overline{\mathbf{H}}^n$:

$$\mathbf{C}^n = \text{LN}(\overline{\mathbf{H}}^n + \mathbf{RE}^n), \quad (8)$$

where LN is a layer normalization. As a result, there is a reordering-aware sentence hidden state representation $\mathbf{C}^n$.

4.2 Stacking SANs with Reordering Embeddings

The original positional embeddings of a sentence allow the Transformer to avoid having to recurrently capture the order of dependencies between words, thus relying entirely on the stacked SANs to parallel learn sentence representations. The learned REs are similar to the original positional embeddings. This means that these learned reordering embeddings can be also easily stacked together with the existing SANs to learn

the final reordering-aware sentence representation for machine translation. According to Eq.(4), stacking SANs with reordering embeddings is formalized as the following Eq.(9):

$$\left[\begin{array}{c} \overline{\mathbf{H}}^n = \mathrm{LN}(\mathrm{SelfAtt}^n(\mathbf{H}^{n-1}) + \mathbf{H}^{n-1}) \\ \mathbf{PP}^n = \mathrm{sigmoid}(\overline{\mathbf{V}}^n \cdot \tanh(\mathbf{W}^n \cdot \mathbf{H}^{n-1} + \overline{\mathbf{W}}^n \cdot \overline{\mathbf{H}}^n)) \\ \mathbf{C}^n = \mathrm{LN}(\overline{\mathbf{H}}^n + \mathbf{PE} \cdot \mathbf{PP}^n) \\ \mathbf{H}^n = \mathrm{LN}(\mathrm{FFN}^n(\mathbf{C}^n) + \overline{\mathbf{H}}^n), \end{array}\right]_N \quad (9)$$

where $\mathbf{H}^0$ is the initialized sentence representation as in the Section 3.1. Finally, there is a reordering-aware sentence representation $\mathbf{H}^N$ for predicting translations.

5 Neural Machine Translation with Reordering Mechanism

Based on the proposed approach to learning sentence representation, we design three Transformer translation models: **Encoder_REs**, **Decoder_REs**, and **Both_REs**, all of which enable reordering knowledge to improve the translation performance of Transformer.

Encoder_REs: The proposed reordering mechanism is only applied to the encoder of Transformer to learn the representation of the source sentence, as shown in the Encoder of Figure 2.

Decoder_REs: Similarly, the proposed reordering mechanism is only introduced into the SAN layer of Transformer related to the representation of the target sentence, as shown in the Decoder of Figure 2.

Both_REs: To further enhance translation performance, we simultaneously apply the proposed method to the source and target sentences to learn their sentence representations, as shown in Figure 2.

Note that the reordering model in PBSMT is an independent model and therefore needs to consider information concerning both the source and target. In NMT, the reordering embedding is jointly trained with the entire NMT model. Although it is only applied to the encoder (or decoder), it can still obtain information about the target (or source) from the decoder (or encoder) by neural network feedback. Therefore, the proposed reordering mechanism makes use of information concerning both the source and the target.

6 Experiments

6.1 Datasets

The proposed method was evaluated on three tasks from the WMT14 English-to-German (EN-DE), NIST Chinese-to-English (ZH-EN), and WAT ASPEC Japanese-to-English (JA-EN) benchmarks.

1) For the EN-DE translation task, 4.43 million bilingual sentence pairs of the WMT14 dataset were used as training data, including Common Crawl, News Commentary, and Europarl v7. The newstest2013 and newstest2014 datasets were used as the dev set and test set, respectively.

2) For the ZH-EN translation task, the training dataset consisted of 1.28 million bilingual sentence pairs from LDC corpus consisting of LDC2002E18, LDC2003E07, LDC2003E14, and Hansard's portions of LDC2004T07, LDC2004T08, and LDC2005T06. The MT06 and the MT02/MT03/MT04/MT05/MT08 datasets were used as the dev set and test set, respectively.

3) For the JA-EN translation task, the training dataset consisted of two million bilingual sentence pairs from the ASPEC corpus (Nakazawa et al., 2016). The dev set consisted of 1,790 sentence pairs and the test set of 1,812 sentence pairs.

6.2 Baseline Systems

These baseline systems included:

Transformer: a vanilla Transformer with absolute positional embedding (Vaswani et al., 2017), for example Transformer (base) and Transformer (big) models.

Relative PE (Shaw et al., 2018): incorporates relative positional embeddings into the self-attention mechanism of Transformer.

Additional PE (control experiment): uses original absolute positional embeddings to enhance the position information of each SAN layer instead of the proposed reordering embeddings.

Pre-reordering: a pre-ordering method (Goto et al., 2013) for JA-EN translation task was used to adjust the order of Japanese words in both the training, dev, and test datasets, and thus reordered each source sentence into the similar order as its target sentence.

6.3 System Setting

For all models (base), the byte pair encoding algorithm (Sennrich et al., 2016) was adopted and the size of the vocabulary was set to 32,000. The number of dimensions of all input and output

System	Architecture	newstest2014	#Speed1	#Speed2	#Params
Existing NMT systems					
Wu et al. (2016)	GNMT	26.3	N/A	N/A	N/A
Gehring et al. (2017b)	CONVS2S	26.36	N/A	N/A	N/A
Vaswani et al. (2017)	Transformer (base)	27.3	N/A	N/A	65.0M
Vaswani et al. (2017)	Transformer (big)	28.4	N/A	N/A	213.0M
Our NMT systems					
this work	Transformer (base)	27.24	9910	181	97.6M
	+Additional PEs	27.10	9202	179	97.6M
	+Relative PEs	27.63	4418	146	97.6M
	+Encoder_REs	28.03++	8816	179	102.1M
	+Decoder_REs	27.61+	9101	175	102.1M
	+Both_REs	28.22++	8605	174	106.8M
	Transformer (big)	28.34	4345	154	272.8M
	+Both_REs	29.11++	3434	146	308.2M

Table 1: Comparison with existing NMT systems on WMT14 EN-DE Translation Task. "#Speed1" and "#Speed2" denote the training and decoding speed measured in source tokens per second, respectively. In Table 1, 2 and 3, "++/+" after score indicate that the proposed method was significantly better than the corresponding baseline Transformer (base or big) at significance level p<0.01/0.05.

layers was set to 512, and that of the inner feed-forward neural network layer was set to 2048. The heads of all multi-head modules were set to eight in both encoder and decoder layers. In each training batch, a set of sentence pairs contained approximately 4096×4 source tokens and 4096×4 target tokens. During training, the value of label smoothing was set to 0.1, and the attention dropout and residual dropout were $p = 0.1$. The Adam optimizer (Kingma and Ba, 2014) was used to tune the parameters of the model. The learning rate was varied under a warm-up strategy with warmup steps of 8,000. For evaluation, we validated the model with an interval of 1,000 batches on the dev set. Following the training of 200,000 batches, the model with the highest BLEU score of the dev set was selected to evaluate on the test sets. During the decoding, the beam size was set to four. All models were trained and evaluated on a single P100 GPU. SacreBELU (Post, 2018) was used as the evaluation metric of EN-DE, and the multi-bleu.perl[1] was used the evaluation metric of ZH-EN and JA-EN tasks. The signtest (Collins et al., 2005) was as statistical significance test. We re-implemented all methods ("this work" in the tables) on the OpenNMT toolkit (Klein et al., 2017).

[1]https://github.com/moses-smt/mosesdecoder/tree/RELEASE-4.0/scripts/generic/multi-bleu.perl

6.4 Main Results

To validate the effectiveness of our methods, the proposed models were first evaluated on the WMT14 EN-DE translation task as in the original Transformer translation system (Vaswani et al., 2017). The main results of the translation are shown in Tables 1. We made the following observations:

1) The baseline Transformer (base) in this work outperformed GNMT, CONVS2S, and Transformer (base)+Relative PEs, and achieved performance comparable to the original Transformer (base). This indicates that it is a strong baseline NMT system.

2) The three proposed models significantly outperformed the baseline Transformer (base). This indicates that the learned reordering embeddings were beneficial for the Transformer. Meanwhile, our models outperformed the comparison system +Additional PEs (control experiment), which means that these improvements in translation derived from the learned REs instead of the original PEs. +Encoder_REs and +Both_REs were superior to +Relative PEs, which means that the REs better captured reordering information than +Relative PEs.

3) Of the proposed models, +Encoder_REs

System	Architecture	Test Sets					#Param
		MT02	MT03	MT04	MT05	MT08	
Existing NMT systems							
Vaswani et al. (2017)	Transformer	N/A	N/A	N/A	N/A	N/A	N/A
Zhang et al. (2017)	RNNsearch+Distortion	N/A	38.33	40.40	36.81	N/A	N/A
Meng and Zhang (2018)	DTMT#1	46.90	45.85	46.78	45.96	36.58	170.5M
Meng and Zhang (2018)	DTMT#4	47.03	46.34	47.52	46.70	37.61	208.4M
Kong et al. (2018)	RNN-based NMT	N/A	38.62	41.98	37.42	N/A	87.9M
Zhao et al. (2018a)	RNN-based NMT+MEM	N/A	44.98	45.51	43.95	33.33	N/A
Our NMT systems							
this work	Transformer (base)	46.45	45.33	45.82	45.57	35.57	78.3M
	+Additional PEs	46.66	45.35	46.11	45.40	35.75	78.3M
	+Relative PEs	46.41	45.94	46.54	46.21	36.14	78.3M
	+Encoder_REs	47.47++	45.87++	46.82++	46.58++	36.42++	83.0M
	+Decoder_REs	46.80	45.43	46.23++	46.11++	36.02+	83.0M
	+Both_REs	47.54++	46.56++	47.27++	46.88++	36.77++	87.6M
	Transformer (Big)	47.76	46.66	47.51	47.71	37.73	244.7M
	+Both_REs	48.42++	47.32++	48.22++	48.56++	38.19+	269.7M

Table 2: Results on NIST ZH-EN Translation Task.

performed slightly better than +Decoder_REs. This indicates that the reordering information of the source sentence was slightly more useful than that of the target sentence. +Both_REs which combined reordering information for both source and target further improved performance and were significantly better than +Encoder_REs and +Decoder_REs. This indicates that the reordering information of source and target can be used together to improve predicted translation.

4) We also evaluated the best performing method (+Both_REs) in big Transformer model settings (Vaswani et al., 2017). Compared with Transformer (base), Transformer (big) contains approximately three times parameters and obtained one BLEU score improvement. The Transformer (big)+Both_REs further achieved 0.77 BLEU score improvement.

5) The proposed models contains approximately 5%∼10% additional parameters and decreased 10%∼15% training speed, compared to the corresponding baselines. Transformer (base)+Both_REs achieved comparable results compared to Transformer (big) which has much more parameters. This indicates that the improvement of the proposed methods is not from more parameters.

6) In Table 3, the +Pre-ordering performed worse that the baseline Transformer (base) for the WAT JA-EN translation task. We assume that the simple +pre-ordering strategy has negative impact on the translation performance of NMT model, which is in line with the functional similarity findings in (Du and Way, 2017; Kawara et al., 2018). Conversely, the proposed methods performed better than the Transformer (base), especially the +pre-ordering. This means that because this pre-ordering operation is isolated with the existing NMT, these generated pre-ordered data are not conducive to model source translation knowledge for the NMT framework.

Systems	testset	#Param
Transformer (base)	30.33	73.9M
+Pre-Reordering	28.93	73.9M
+Additional PEs	30.16	73.9M
+Relative PEs	30.42	73.9M
+Encoder_REs	31.12++	78.6M
+Decoder_REs	30.78+	78.6M
+Both_REs	31.41++	84.4M
Transformer (big)	31.21	234.6M
+Both_REs	31.93++	273.7M

Table 3: Results for WAT JA-EN Translation Task.

In addition, Tables 2 and 3 show that the proposed models yielded similar improvements over the baseline system and the compared methods on the NIST ZH-EN and WAT JA-EN translation tasks. These results indicate that our method can effectively improve the NIST ZH-EN and WAT JA-EN translation tasks. In other words, our approach is a universal method for improving the translation of other language pairs.

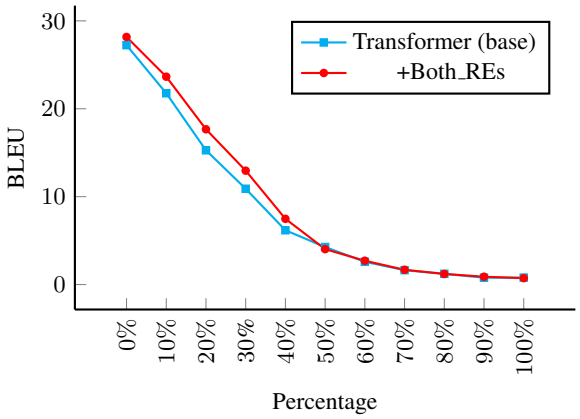

Figure 3: The effect of reordering in the test set where the word orders are partially wrong for test set of EN-DE. "Percentage" denotes that there is percentage of swapped words in one source sentence.

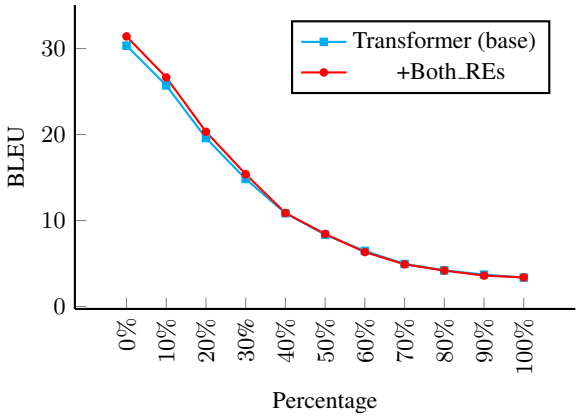

Figure 4: The effect of reordering in the test set where the word orders are partially wrong for test set of JA-EN.

6.5 Effect of Reordering Embeddings

Unlike the reordering model in PBSMT, which can be illustrated explicitly, it is challenging to explicitly show the effect of reordering embedding. To further analyze this effect, we simulated a scenario where the word order of a sentence was partially incorrect and reordering was needed for NMT. We randomly swapped words of a source sentence in the test set according to different percentages of incorrectly swapped words in a sentence. For example, "10%" indicates that there were 10% randomly swapped words for each source sentence in the test set. We evaluated Transformer (base) and +Both_REs (base) on these test set for three translation tasks and the results are as shown in Figure 3, 4, and 5.

1) We observed that when the ratio of swapped words gradually increased, the performances

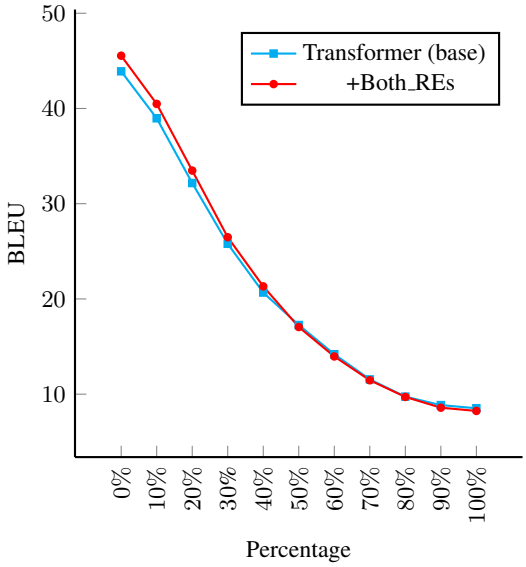

Figure 5: The effect of reordering in the test set where the word orders are partially wrong for test set of ZH-EN.

of Transformer (base) and +Both_REs (base) significantly degraded. This indicates that correct ordering information has an important effect on the Transformer system.

2) When the percentage of swapped words was less than 40%, the NMT systems still delivered reasonable performance. The gap between +Both_REs (base) and Transformer (base) was approximately 2-3 BLEU scores. This indicates that +Both_REs (base) dealt better than the vanilla baseline with this scenario. In other words, the learned REs retrained part of reordering information in a sentence.

3) When the percentage of swapped words was greater than 40%, Transformer (base) and +Both_REs (base) yielded poor performance on translation. We infer that excessive exchanges of word order may increase the ambiguity of the source sentence such that Transformer (base) and +Both_REs (base) struggled to convert the original meaning of the source sentence into the target translation.

6.6 Cases Analysis

Figure 6 shows two translation examples, which were generated by Transformer (base) model and +Both_REs (base) model, respectively.

For the first sample, +Both_REs (base) translated the Chinese phrase "继续[continue] 改革[reform] 的[to] 努力[efforts]" into the "the efforts to continue the reform" while Transformer

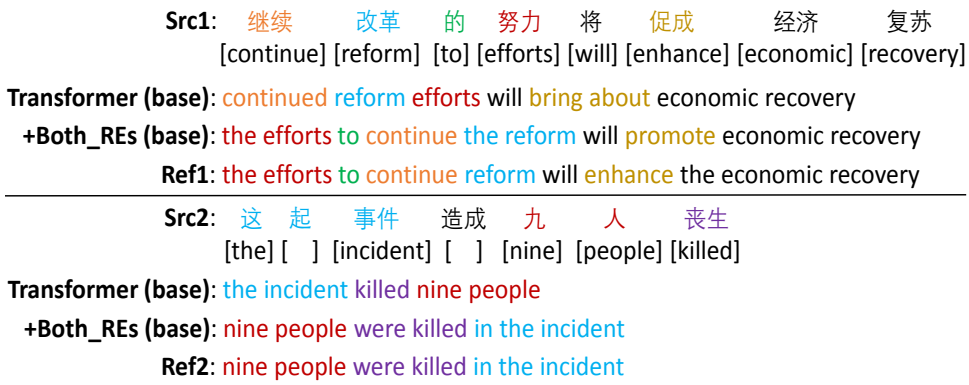

Figure 6: Two translation examples for ZH-EN task. In each example, the English phrases in color indicate they are translations from the corresponding Chinese phrase with the same color.

(base) translated the Chinese phrase into "continued reform efforts". Although both of them covered the meanings of main words, the order of the former translation is closer to the natural English word order.

For the second sample, Transformer (base) generated a puzzling translation "the incident killed nine people". It seems to be an English sentence in Chinese word order. In comparison, the +Both_REs (base) translated it into "nine people were killed in the incident" which is the same as the reference.

These two examples show that the proposed model with reordering embeddings was conducive to generating a translation in line with the target language word order.

7 Conclusion and Future Work

Word ordering is an important issue in translation. However, it has not been extensively studied in NMT. In this paper, we proposed a reordering mechanism to capture knowledge of reordering. A reordering embedding was learned by considering the relationship between the positional embedding of a word and that of the entire sentence. The proposed reordering embedding can be easily introduced to the existing Transformer translation system to predict translations. Experiments showed that our method can significantly improve the performance of Transformer.

In future work, we will further explore the effectiveness of the reordering mechanism and apply it to other natural language processing tasks, such dependency parsing (Zhang et al., 2016; Li et al., 2018), and semantic role labeling (He et al., 2018; Li et al., 2019).

Acknowledgments

We are grateful to the anonymous reviewers and the area chair for their insightful comments and suggestions. This work was partially conducted under the program "Promotion of Global Communications Plan: Research, Development, and Social Demonstration of Multilingual Speech Translation Technology" of the Ministry of Internal Affairs and Communications (MIC), Japan. Rui Wang was partially supported by JSPS grant-in-aid for early-career scientists (19K20354): "Unsupervised Neural Machine Translation in Universal Scenarios" and NICT tenure-track researcher startup fund "Toward Intelligent Machine Translation".

References

Yaser Al-Onaizan and Kishore Papineni. 2006. Distortion models for statistical machine translation. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 529–536, Sydney, Australia. Association for Computational Linguistics.

Lei Jimmy Ba, Ryan Kiros, and Geoffrey E. Hinton. 2016. Layer normalization. *CoRR*, abs/1607.06450.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *Proceedings of the 3rd International Conference on Learning Representations*, San Diego, CA.

Arianna Bisazza and Marcello Federico. 2016. A survey of word reordering in statistical machine translation: Computational models and language phenomena. *Computational Linguistics*, 42(2):163–205.

Ondrej Bojar, Christian Federmann, Mark Fishel, Yvette Graham, Barry Haddow, Matthias Huck, Philipp Koehn, and Christof Monz. 2018. Findings of the 2018 conference on machine translation (wmt18). In *Proceedings of the Third Conference on Machine Translation, Volume 2: Shared Task Papers*, pages 272–307, Belgium, Brussels. Association for Computational Linguistics.

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

Huadong Chen, Shujian Huang, David Chiang, and Jiajun Chen. 2017a. Improved neural machine translation with a syntax-aware encoder and decoder. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1936–1945, Vancouver, Canada. Association for Computational Linguistics.

Kehai Chen, Rui Wang, Masao Utiyama, Lemao Liu, Akihiro Tamura, Eiichiro Sumita, and Tiejun Zhao. 2017b. Neural machine translation with source dependency representation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2846–2852, Copenhagen, Denmark. Association for Computational Linguistics.

Kehai Chen, Rui Wang, Masao Utiyama, Eiichiro Sumita, and Tiejun Zhao. 2018a. Syntax-directed attention for neural machine translation. In *AAAI Conference on Artificial Intelligence*, pages 4792–4798, New Orleans, Lousiana, USA.

Kehai Chen, Tiejun Zhao, Muyun Yang, and Lemao Liu. 2017c. Translation prediction with source dependency-based context representation. In *AAAI Conference on Artificial Intelligence*, pages 3166–3172, San Francisco, California, USA.

Kehai Chen, Tiejun Zhao, Muyun Yang, Lemao Liu, Akihiro Tamura, Rui Wang, Maosao Utiyama, and Eiichro Sumita. 2018b. A neural approach to source dependence based context model for statistical machine translation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 26(2):266–280.

Colin Cherry. 2013. Improved reordering for phrase-based translation using sparse features. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 22–31, Atlanta, Georgia. Association for Computational Linguistics.

David Chiang. 2005. A hierarchical phrase-based model for statistical machine translation. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 263–270, Ann Arbor, Michigan. Association for Computational Linguistics.

Michael Collins, Philipp Koehn, and Ivona Kucerova. 2005. Clause restructuring for statistical machine translation. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics*, pages 531–540, Ann Arbor, Michigan. Association for Computational Linguistics.

Jinhua Du and Andy Way. 2017. Pre-Reordering for Neural Machine Translation: Helpful or Harmful? *The Prague Bulletin of Mathematical Linguistics*, 108:171–182.

Nadir Durrani, Alexander Fraser, Helmut Schmid, Hieu Hoang, and Philipp Koehn. 2013. Can markov models over minimal translation units help phrase-based smt? In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 399–405, Sofia, Bulgaria. Association for Computational Linguistics.

Nadir Durrani, Philipp Koehn, Helmut Schmid, and Alexander Fraser. 2014. Investigating the usefulness of generalized word representations in smt. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 421–432, Dublin, Ireland. Dublin City University and Association for Computational Linguistics.

Nadir Durrani, Helmut Schmid, and Alexander Fraser. 2011. A joint sequence translation model with integrated reordering. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 1045–1054, Portland, Oregon, USA. Association for Computational Linguistics.

Akiko Eriguchi, Kazuma Hashimoto, and Yoshimasa Tsuruoka. 2016. Tree-to-sequence attentional neural machine translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 823–833, Berlin, Germany. Association for Computational Linguistics.

Minwei Feng, Arne Mauser, and Hermann Ney. 2010. A source-side decoding sequence model for statistical machine translation. In *The Ninth Conference of the Association for Machine Translation in the Americas*, Denver, Colorado.

Minwei Feng, Jan-Thorsten Peter, and Hermann Ney. 2013. Advancements in reordering models for statistical machine translation. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 322–332, Sofia, Bulgaria. Association for Computational Linguistics.

Michel Galley and Christopher D. Manning. 2008. A simple and effective hierarchical phrase reordering model. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 848–856, Honolulu, Hawaii. Association for Computational Linguistics.

Jonas Gehring, Michael Auli, David Grangier, and Yann Dauphin. 2017a. A convolutional encoder model for neural machine translation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 123–135, Vancouver, Canada. Association for Computational Linguistics.

Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N. Dauphin. 2017b. Convolutional sequence to sequence learning. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 1243–1252, International Convention Centre, Sydney, Australia. PMLR.

Isao Goto, Masao Utiyama, and Eiichiro Sumita. 2013. Post-ordering by parsing with itg for japanese-english statistical machine translation. *ACM Transactions on Asian Language Information Processing*, 12(4):17:1–17:22.

Spence Green, Michel Galley, and Christopher D. Manning. 2010. Improved models of distortion cost for statistical machine translation. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 867–875, Los Angeles, California. Association for Computational Linguistics.

Andreas Guta, Tamer Alkhouli, Jan-Thorsten Peter, Joern Wuebker, and Hermann Ney. 2015. A comparison between count and neural network models based on joint translation and reordering sequences. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1401–1411, Lisbon, Portugal. Association for Computational Linguistics.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778.

Shexia He, Zuchao Li, Hai Zhao, and Hongxiao Bai. 2018. Syntax for semantic role labeling, to be, or not to be. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2061–2071, Melbourne, Australia.

Yuki Kawara, Chenhui Chu, and Yuki Arase. 2018. Recursive neural network based preordering for english-to-japanese machine translation. In *Proceedings of ACL 2018, Student Research Workshop*, pages 21–27, Melbourne, Australia. Association for Computational Linguistics.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980.

Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander Rush. 2017. Opennmt: Open-source toolkit for neural machine translation. In *Proceedings of ACL 2017, System Demonstrations*, pages 67–72, Vancouver, Canada. Association for Computational Linguistics.

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

Xiang Kong, Zhaopeng Tu, Shuming Shi, Eduard H. Hovy, and Tong Zhang. 2018. Neural machine translation with adequacy-oriented learning. *CoRR*, abs/1811.08541.

Junhui Li, Deyi Xiong, Zhaopeng Tu, Muhua Zhu, Min Zhang, and Guodong Zhou. 2017. Modeling source syntax for neural machine translation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 688–697, Vancouver, Canada. Association for Computational Linguistics.

Peng Li, Yang Liu, and Maosong Sun. 2013. Recursive autoencoders for itg-based translation. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 567–577, Seattle, Washington, USA. Association for Computational Linguistics.

Peng Li, Yang Liu, Maosong Sun, Tatsuya Izuha, and Dakun Zhang. 2014. A neural reordering model for phrase-based translation. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 1897–1907, Dublin, Ireland. Dublin City University and Association for Computational Linguistics.

Zuchao Li, Jiaxun Cai, Shexia He, and Hai Zhao. 2018. Seq2seq dependency parsing. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3203–3214, Santa Fe, New Mexico, USA.

Zuchao Li, Shexia He, Hai Zhao, Yiqing Zhang, Zhuosheng Zhang, Xi Zhou, and Xiang Zhou. 2019. Dependency or span, end-to-end uniform semantic role labeling. *CoRR*, abs/1901.05280.

Benjamin Marie, Rui Wang, Atsushi Fujita, Masao Utiyama, and Eiichiro Sumita. 2018. Nict's neural and statistical machine translation systems for the wmt18 news translation task. In *Proceedings of the Third Conference on Machine Translation, Volume 2: Shared Task Papers*, pages 453–459, Belgium, Brussels. Association for Computational Linguistics.

Fandong Meng and Jinchao Zhang. 2018. DTMT: A novel deep transition architecture for neural machine translation. *CoRR*, abs/1812.07807.

Haitao Mi, Baskaran Sankaran, Zhiguo Wang, and Abe Ittycheriah. 2016. Coverage embedding models for neural machine translation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 955–960, Austin, Texas. Association for Computational Linguistics.

Masaaki Nagata, Kuniko Saito, Kazuhide Yamamoto, and Kazuteru Ohashi. 2006. A clustered global phrase reordering model for statistical machine translation. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 713–720, Sydney, Australia. Association for Computational Linguistics.

Toshiaki Nakazawa, Manabu Yaguchi, Kiyotaka Uchimoto, Masao Utiyama, Eiichiro Sumita, Sadao Kurohashi, and Hitoshi Isahara. 2016. ASPEC: Asian scientific paper excerpt corpus. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, pages 2204–2208, Portorož, Slovenia. European Language Resources Association (ELRA).

Matt Post. 2018. A call for clarity in reporting BLEU scores. *CoRR*, abs/1804.08771.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Peter Shaw, Jakob Uszkoreit, and Ashish Vaswani. 2018. Self-attention with relative position representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 464–468, New Orleans, Louisiana. Association for Computational Linguistics.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112. Curran Associates, Inc.

Christoph Tillman. 2004. A unigram orientation model for statistical machine translation. In *Proceedings of HLT-NAACL 2004: Short Papers*, Stroudsburg, PA, USA.

Zhaopeng Tu, Zhengdong Lu, Yang Liu, Xiaohua Liu, and Hang Li. 2016. Modeling coverage for neural machine translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 76–85, Berlin, Germany.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 5998–6008. Curran Associates, Inc.

Rui Wang, Andrew Finch, Masao Utiyama, and Eiichiro Sumita. 2017a. Sentence embedding for neural machine translation domain adaptation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 560–566, Vancouver, Canada. Association for Computational Linguistics.

Rui Wang, Masao Utiyama, Lemao Liu, Kehai Chen, and Eiichiro Sumita. 2017b. Instance weighting for neural machine translation domain adaptation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1482–1488, Copenhagen, Denmark. Association for Computational Linguistics.

Rui Wang, Masao Utiyama, and Eiichiro Sumita. 2018. Dynamic sentence sampling for efficient training of neural machine translation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 298–304, Melbourne, Australia. Association for Computational Linguistics.

Dekai Wu. 1996. A polynomial-time algorithm for statistical machine translation. In *Proceedings of the 34th Annual Meeting on Association for Computational Linguistics*, ACL '96, pages 152–158, Santa Cruz, California. Association for Computational Linguistics.

Dekai Wu. 1997. Stochastic inversion transduction grammars and bilingual parsing of parallel corpora. *Computational Linguistics*, 23(3).

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Lukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *CoRR*, abs/1609.08144.

Deyi Xiong, Qun Liu, and Shouxun Lin. 2006. Maximum entropy based phrase reordering model for statistical machine translation. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 521–528, Sydney, Australia. Association for Computational Linguistics.

Richard Zens and Hermann Ney. 2006. Discriminative reordering models for statistical machine translation. In *Proceedings on the Workshop on Statistical Machine Translation*, pages 55–63, New York City. Association for Computational Linguistics.

Jinchao Zhang, Mingxuan Wang, Qun Liu, and Jie Zhou. 2017. Incorporating word reordering knowledge into attention-based neural machine translation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1524–1534, Vancouver, Canada. Association for Computational Linguistics.

Zhisong Zhang, Rui Wang, Masao Utiyama, Eiichiro Sumita, and Hai Zhao. 2018. Exploring recombination for efficient decoding of neural machine translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4785–4790, Brussels, Belgium. Association for Computational Linguistics.

Zhisong Zhang, Hai Zhao, and Lianhui Qin. 2016. Probabilistic graph-based dependency parsing with convolutional neural network. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1382–1392, Berlin, Germany.

Yang Zhao, Jiajun Zhang, Zhongjun He, Chengqing Zong, and Hua Wu. 2018a. Addressing troublesome words in neural machine translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 391–400, Brussels, Belgium. Association for Computational Linguistics.

Yang Zhao, Jiajun Zhang, and Chengqing Zong. 2018b. Exploiting pre-ordering for neural machine translation. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018)*, Miyazaki, Japan. European Language Resource Association.