Exploiting Pre-Ordering for Neural Machine Translation

Yang Zhao, Jiajun Zhang and Chengqing Zong
National Laboratory of Pattern Recognition, Institute of Automation, CAS
University of Chinese Academy of Sciences
{yang.zhao, jjzhang, cqzong}@nlpr.ia.ac.cn

Abstract

Neural Machine Translation (NMT) has drawn much attention due to its promising translation performance in recent years. However, the under-translation and over-translation problem still remain a big challenge. Through error analysis, we find that under-translation is much more prevalent than over-translation and the source words that need to be reordered during translation are more likely to be ignored. To address the under-translation problem, we explore the pre-ordering approach for NMT. Specifically, we pre-order the source sentences to approximate the target language word order. We then combine the pre-ordering model with position embedding to enhance the monotone translation. Finally, we augment our model with the coverage mechanism to tackle the over-translation problem. Experimental results on Chinese-to-English translation have shown that our method can significantly improve the translation quality by up to 2.43 BLEU points. Furthermore, the detailed analysis demonstrates that our approach can substantially reduce the number of under-translation cases by 30.4% (compared to 17.4% using the coverage model).

Keywords: Neural Machine Translation, pre-ordering, under translation, over translation

1. Introduction

The Past several years have witnessed a significant progress in Neural Machine Translation (NMT). Most NMT methods are based on the encoder-decoder architecture proposed by (Kalchbrenner and Blunsom, 2013; Cho et al., 2014; Sutskever et al., 2014; Bahdanau et al., 2015) and can achieve promising translation performance in a variety of language pairs (Junczys-Dowmunt et al., 2016; Wu et al., 2016).

However, previous studies have showed that NMT suffers from the problems that some source words are mistakenly translated for multiple times meanwhile some words are missed during translation (Tu et al., 2016; Tu et al., 2017; Mi et al., 2016; Feng et al., 2016), which can be called over-translation and under-translation, respectively[1].

Under-translation		Over-translation	
Times	No. Words	Times	No. Words
92	307	32	48

Table 1: Statistics on the under-translation and the over-translation in NMT.

Under-translation		
Reorder	No reorder	Sub-sentence
48	26	18

Table 2: Statistics on different kinds of the under-translation.

In order to figure out the distribution of over-translation and under-translation in NMT, we analyze 500 sentences translated by the NMT system, which is trained by 2.1M parallel Chinese-English sentences pairs. Table 1 shows the statistical results. Specifically, in 500 sentences, NMT system produces 92 under-translations and 32 over-translations. Besides that, for the under-translation, the total number of missing words is 307, while the number of over-translated words is 48. From these statistics, we can see that the under-translation in NMT is more serious than the over-translation.

Therefore, further analysis for the under-translation is made and Table 2 shows the results. In 92 under-translations, we find that the source words should to be reordered during translation are more likely to be missed by NMT and this kind of under-translation occurs 48 times. While the opposite case, i.e. source words requiring no reordering are missed by NMT, occurs 26 times. The remaining (18 times) is the case that the sub-sentences in source are totally dropped. From these statistics, we think that the first kind of under-translation, i.e. words need to be reordered are ignored, is a major problem affecting the final translation quality.

Considering the fact that source words requiring reordering during translation are more likely to be ignored by the NMT model, we propose to exploit the pre-ordering approach which is commonly used in Statistical Machine Translation (SMT). The pre-ordering can make the word order of a source sentence closer to that of a target sentence (Genzel, 2010; Hitschler et al., 2016). We first pre-order the source sentences to approximate the target language word order. We then further combine the pre-ordering model with the position embedding strategy to enhance the monotone translation. Finally, to overcome the over-translation problem, we augment our model with the coverage mechanism.

In this paper, we make the following contributions:

1) Through error analysis, we find that under-translation occurs more frequently than over-translation in NMT and source words that need reordering are more likely to be missed. We propose a pre-ordering approach enhanced with position embedding to tackle the under-translation problem and augment our model with coverage mechanism

[1] (Mi et al., 2016) calls this phenomenon as "repeating and dropping translations". Here, we adopt (Tu et al., 2016)'s expressions, i.e. over-translation and under-translation.

to address the over-translation problem.
2) Our empirical experiments on Chinese-English translation tasks show the efficacy of our approach. We can obtain an average improvement of 1.65 BLEU score on multiple evaluation datasets (the largest improvement can be up to 2.43 BLEU points). Furthermore, the analysis on under-translation shows that our approach can substantially reduce the number of under-translation by 30.4% (compared to 17.4% using the coverage model).

2. Neural Machine Translation

Attention-based NMT contains two parts, encoder and decoder, Encoder transforms the source sentence $X = \{x_1, x_2, ..., x_{Tx}\}$ into context vectors $C = \{h_1, h_2, ..., h_{Tx}\}$. This context set is constructed by m stacked Long Short Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997) layers. h_j^k can be calculated as follows:

$$h_j^k = LSTM(h_{j-1}^k, h_j^{k-1}) \tag{1}$$

The decoder generates one target word at a time by maximizing the probability of $p(y_i|y_{<i}, C)$ as follows:

$$\begin{aligned} p(y_i|y_{<i}, C) &= p(y_i|y_{<i}, c_i) \\ &= softmax(W_{y_i}\widetilde{z_i} + b_s) \end{aligned} \tag{2}$$

where W_y is an embedding matrix containing row vectors of the target words and $\widetilde{z_i}$ is the attention output:

$$\widetilde{z_i} = tanh(W_c[z_i^m; c_i]) \tag{3}$$

The attention model calculates c_i as the weighted sum of the source-side context vectors:

$$c_i = \sum_{j=1}^{Tx} a_{i,j} h_i^m \tag{4}$$

Where a_{ij} can be computed by

$$a_{i,j} = \frac{exp(e_{i,j})}{\sum_{k=1}^{Tx} exp(e_{i,k})} \tag{5}$$

and

$$e_{i,j} = v_a^T tanh(W_a z_i + U_a h_j) \tag{6}$$

z_i^k is computed using the following formula:

$$z_j^k = LSTM(z_{j-1}^k, z_j^{k-1}) \tag{7}$$

3. Exploiting Pre-Ordering for NMT

In SMT, pre-ordering is a commonly used pre-processing technique (Collins et al., 2005; Zhang and Zong, 2009; Genzel, 2010; Hitschler et al., 2016), which makes the word order of a source sentence closer to that of a target sentence. This technology was originally proposed to alleviate the weakness of reordering in classical phrase-based SMT (Koehn et al., 2003). As SMT always penalizes the cases that move target phrases far away from their corresponding source positions. Fig. 1 shows an example of pre-ordering, in which when translating the original source sentence, the words in red and words in blue need to exchange their positions. With the pre-ordering, the word order in this source sentence is adjusted to the word order in reference. When translating the pre-ordered source sentence, the translation system does not need to reorder the source words.
Since we find that the source words should to be reordered during translation are more likely to be ignored by NMT. We believe that the pre-ordering can help to alleviate the under-translation problem.

3.1. Pre-Ordering

There are many pre-ordering methods introduced in SMT. The most common way to implement a pre-ordering system employs the rule-based approach. The early works rely on hand-written rules (Collins et al., 2005). Later some works could extract the pre-ordering rules automatically (Genzel, 2010; Hitschler et al., 2016). Here, we adopt the automatic rule-based pre-ordering approach. And the procedure is as follows:
With a parallel training corpus, we first train a pre-ordering system. The basic training procedure is extracting the pre-ordering rules, which can minimize the number of alignment crossings in the parallel corpus. More details can be found in (Genzel, 2010; Hitschler et al., 2016).
After acquiring the pre-ordering rules, we can use them to pre-order the source sentences. Note that the word order of the target sentence does not change.

3.2. Position Embedding

As mentioned before, the most noticeable feature of pre-ordering is that it can make the word order in source more consistent with the word order in target. Intuitively, monotone translation is preferred. That is to say the words in the similar positions between the source and target sentences are more likely to be translation pairs. Thus, we further enhance the pre-ordering model with the position embedding to encourage monotone translation.
Actually, previous studies (Cohn et al., 2016; Gehring et al., 2017; Vaswani et al., 2017) have shown that the position information is effective for NMT, and these studies are all based on the following assumption:
Assumption: *a word at a given relative position j in the source (whose length is denoted as J) is more likely to align to a word at a similar relative position i in the target (whose length is denoted as I), i.e. $\frac{j}{J} \approx \frac{i}{I}$.*
Obviously, pre-ordering can make more words satisfy this assumption. We design the procedure as follows:
We first randomly generate the respective position embedding matrix for the source and target positions, which are denoted as $E_s \in \mathbb{R}^{n \times l}$ and $E_t \in \mathbb{R}^{n \times l}$, respectively, where n is the position embedding dimension, and l is largest sentence length. $E_s(j)$ denotes the position embedding for source position j and $E_t(i)$ denotes the position embedding for target position i. Note that the position embedding is optimized during training, like the word embedding.
Then, we redesign the attention part in Eq. 6 as follows:

$$\begin{aligned} e_{i,j} = v_a^T tanh(&W_a z_i + U_a h_j + \\ &W_t E_t(i) + W_s E_s(j)) \end{aligned} \tag{8}$$

where $W_t \in \mathbb{R}^{m \times n}$ and $W_s \in \mathbb{R}^{m \times n}$ are the weight matrices for position embedding with m and n being the hid-

Source:
美国 官员(the us officials) 以 咬文 嚼字 的 外交 用词 (with carefully worded diplomatic rhetoric) 坚称(insisted) 。

Pre-Ordering:
美国 官员 (the us officials) 坚称 (insisted) 以 咬文 嚼字 的 外交 用词 (with carefully worded diplomatic rhetoric)。

Reference:
the us officials insisted with carefully worded diplomatic rhetoric.

Figure 1: A example of pre-ordering.

den states dimension and position embedding dimension, respectively.

As shown in Eq. 8, our attention model contains two parts, namely, hidden states based attention (attention between t_i and h_j) and position embedding based attention (attention between $E_t(i)$ and $E_s(j)$). We hope that when some source words are dropped by hidden states based attention, position embedding based attention could pick them up, and vice versa.

3.3. Coverage Mechanism

In Section 3.2, we propose an approach which combines the pre-ordering model with position embedding. Our experimental results show that this approach can alleviate the under-translation problem, especially can sharply reduce the number of under-translation cases for the words that should be reordered during translation. However, the model lacks the ability to handle the over-translation problem. The detailed statistical data is shown in Section 5.2.

To tackle the over-translation problems, we enhance our model with the coverage mechanism. The coverage mechanism is originally proposed in SMT to indicate whether a source word translated or not. Then, some studies (Tu et al., 2016; Mi et al., 2016) exploit the coverage for NMT. We believe that the coverage mechanism could help to overcome the over-translation problems as they can let NMT consider less about the translated words.

Here, we employ the method proposed in (Tu et al., 2016), which maintains a coverage vector to keep track of the attention history. Then the coverage vector is fed to attention model to adjust the attention in the next step. More Specifically, two steps are needed:

We need to maintain a coverage vector, which summarizes the attention record at each decode step as follows:

$$C_{i,j} = C_{i-1,j} + \frac{1}{\Phi_j} a_{i,j} = \frac{1}{\Phi_j} \sum_{k=1}^{i} a_{i,j} \tag{9}$$

where Φ_j is the fertility for word x_j, and can be computed by

$$\Phi_j = N \cdot \sigma(U_f h_j) \tag{10}$$

Where N is the largest fertility, and U_f is the weight matrix. More details can be found in (Tu et al., 2016).

After generating a coverage vector, we need use this as the complementary information to adjust attention in the next time step. Thus, we rewrite the Eq. 8 as follows:

$$\begin{aligned} e_{i,j} =& v_a^T tanh(W_a z_i + U_a h_j + \\ & W_t E_t(i) + W_s E_s(j) + V_a C_{i-1,j}) \end{aligned} \tag{11}$$

where $C_{i-1,j}$ is the coverage vector of source word x_j before time i, and V_a is the weight matrix for coverage vector.

4. Experimental Settings

4.1. Dataset

We test the proposed approaches on Chinese-to-English translation, which includes 2.1M[2] sentence pairs. NIST 2003 (MT03) dataset is used for validation. NIST2004-2006 (MT04-06) and NIST 2008 (MT08) datasets are used for testing.

4.2. Training and Evaluation Details

We use the Zoph_RNN toolkit[3] to implement our described methods. The encoder and decoder include two stacked LSTM layers. The word embedding dimension, the size of hidden layers and the position embedding dimension are all set to 1,000. Minibatch size is set to 128. We limit the vocabulary to 30K most frequent words for both the source and target languages. Other words are replaced by a special symbol UNK. The largest source and target length is set to 50. At test time, we employ beam search with beam size 12. when the length of test sentence exceeds 50, the embedding for the position > 50 is set to zero. We use case-insensitive 4-gram BLEU score as the automatic metric (Papineni et al., 2002) for translation quality evaluation.

4.3. Pre-Ordering Tool

We use Otedama[4] as the pre-ordering tool. Otedama is an open-source tool for rule-based syntactic pre-ordering. Hyper-parameters we used in Otedama are set as follows: window size is set to 3, matching feature is 10, and the max waiting time is 30 minute. The others are set to the default values. More details can be found in (Hitschler et al., 2016).

4.4. Translation Methods

In the experiments, we compare our approaches with other models, and we list all the translation methods as follows:

1) **Moses**: It is the state-of-the-art phrase-based SMT system (Koehn et al., 2007). Our system is built using the default settings.

2) **Baseline**: It is the baseline attention-based NMT system (Luong et al., 2015; Zoph and Knight, 2016).

[2]LDC2000T50, LDC2002L27, LDC2002T01, LDC2002E18, LDC2003E07, LDC2003E14, LDC2003T17, LDC2004T07.

[3]https://github.com/isi-nlp/ZophRNN. We extend this toolkit with global attention, and change the attention model to the way shown in Eq. 6.

[4]https://github.com/StatNLP/otedama.

3) **+Pre-Ordering**: It is the NMT system which only uses the pre-ordering approach.
4) **+Position**: It is the NMT system which only employs the position embedding.
5) **+Pre-Ordering+Position**: It is the NMT system using both pre-ordering and position embedding together.
6) **+Coverage**: It is the NMT system with the coverage mechanism (Tu et al., 2016).
7) **+Pre-Ordering+Position+Coverage**: It is the NMT system with the pre-ordering, position embedding and coverage mechanism.

5. Translation Results

5.1. Translation Quality

Table 3 reports the translation results measured in BLEU score. The first question we are interested in is whether or not can the system only using the pre-ordering improve the translation quality. Compared to the baseline system (Row 2), our pre-ordering approach (Row 3) improves the translation results with 0.32 BLEU, indicating that only using pre-ordering in NMT can improve the final results while the improvements are quite small.

The next focus is the effect of combining the pre-ordering system with the position embedding. The system with pre-ordering and position embedding (Rows 5) outperforms the baseline by an average of 1.13 BLEU points. As a comparison, the system only using the position embedding (Row 4) improves the baseline with 0.37 BLEU. Thus, we find an interesting result that when using pre-ordering and position embedding separately, the respective improvement is quite small (0.32 BLEU and 0.37 BLEU, respectively), but using them together can significantly boost the performance (1.13 BLEU), suggesting that pre-ordering and position embedding can enhance each other.

The system which combines the pre-ordering, position embedding and coverage mechanism together (Row 7) further improves the baseline with 1.65 BLEU. As a comparison, the system with only coverage leads to 0.68 BLEU improvement.

5.2. Under-translation and Over-translation

Besides the translation quality, our approaches also aim to reduce the under-translation and over-translation cases in NMT. Therefore, we randomly select 500 source sentences and analyze the translation results produced by different systems to evaluate their performances on the under-translation and over-translation. Table 4 lists the numbers of the under-translation and over-translation produced by different methods.

We first focus on the under-translation cases. Comparing to the baseline (Row 1), the system only using pre-ordering (Row 2) can reduce 3 cases (from 48 to 45) in which the words that require reordering are missed during translation. And the system only using position embedding (Row 3) can reduce 5 cases (from 48 to 43). When we use pre-ordering and position embedding together (Row 4), the the under-translation cases are reduced by 13 ones (from 48 to 35). In addition, the other two kinds of under-translations are also reduced by 7 (from 26 to 19) and 4 (from 18 to 14) times, respectively. The statistics show that the system using the pre-ordering and position embedding can alleviate the under-translation problem, especially for the words that need reordering during translation. Fig. 2 shows an example, in which source words in red and source words in blue need to be reordered during translation. The baseline translates the blue words while drops the red ones. Our approaches using pre-ordering and position embedding can fix this under-translation.

However, when considering the over-translation, we can find a drawback of the system using the pre-ordering and position embedding, it increases 4 (from 32 to 36) over-translation cases. It is thus necessary to augment our model with coverage mechanism. When augmenting our model

Source:
灾难 发生 至今 , 目前 已 有 放弃 寻找 下落 仍 不明 的 数千 名 观光客 的 声音 , 当中 许多 都 是 外国 游客 。

Reference:
since the disaster occurred , there is a voice now to give up the search for thousands of tourists still unaccounted for , many of them foreign tourists .

Pre-Ordering:
灾难 发生 至今 , 目前 已 有 的 声音 放弃 寻找 下落 仍 不明 的 数千 名 观光客, 当中 许多 都 是 外国 游客 。

Baseline:
since the occurrence of the incident , there have been a few thousand tourists who are still unknown , many of whom are foreign tourists .

+Pre-Ordering+Position:
since the occurrence of the disaster , it has a voice now to abandon several thousands of tourists who are still unknown , many of them are foreign tourists .

Figure 2: A translation example investigating pre-ordering and position embedding.

Source:
美国 官员 以 咬文 嚼字 的 外交 用词 坚称 , 他们 虽然 愿意 和 北韩 谈判 , 但是 在 北韩 遵行 禁止 核子 武器 的 各 项 协定 之前 , 他们 不 会 考虑 展开 谈判 程序 。

Reference:
the us officials insisted with carefully worded diplomatic rhetoric that although they are willing to negotiate with north korea , They will not consider any negotiation procedures before north korea abides by various agreements on banning nuclear weapons .

Pre-Ordering:
美国 官员 坚称 以 咬文 嚼字 的 外交 用词 , 他们 虽然 愿意 和 北韩 谈判 , 但是 在 北韩 遵行 禁止 核子 武器 的 各 项 协定 之前 , 他们 不 会 考虑 展开 谈判 程序 。

Baseline:
us officials insist that they are willing to negotiate with north korea . however , they will not consider implementing the negotiation process prior to north korea compliance with all the agreements on banning nuclear weapons .

+Pre-Ordering+Position:
us officials insisted on <UNK> 's diplomatic terms that although they were willing to negotiate with north korea . they will not consider starting the negotiation process until north korea complies with north korea complies with the agreements to ban nuclear weapons .

+Coverage:
united states officials assert that although they are willing to negotiate with north korea. they would not consider initiating negotiations until north korea had adhered to the agreements prohibiting nuclear weapons .

+Pre-Ordering+Position+Coverage:
us officials insisted with <UNK> 's diplomatic terms that although they were willing to negotiate with north korea . they will not consider starting the negotiation until north korea had the agreements on banning nuclear weapons .

Figure 3: A translation example investigating pre-ordering, position embedding and coverage mechanism.

#	System	MT03	MT04	MT05	MT06	MT08	Ave
1	Moses	38.54	39.01	36.55	35.59	24.76	34.89
2	Baseline	38.99	40.69	35.20	38.60	28.48	36.39
3	+ Pre-Ordering	39.06	41.06*	35.80†	38.96*	28.65	36.71
4	+ Position	39.08	41.40†	36.30†	38.16*	28.84*	36.76
5	+ Pre-Ordering+Position	39.92†	41.71†	36.95†	39.75†	29.27†	37.52
6	+ Coverage	39.09	41.26*	36.90†	39.19†	28.93*	37.07
7	+ Pre-Ordering+Position+Coverage	**40.42**†	**42.23**†	**37.63**†	**39.94**†	**29.97** †	**38.04**

Table 3: Translation results (BLEU score) for different translation methods. "*" indicates that it is statistically significant better ($p < 0.05$) than Baseline and "†" indicates $p < 0.01$.

#	Sytem	Under-translation			Over-translation
		Reorder	No reorder	Sub-sentence	
1	Baseline	48	26	18	32
2	+Pre-Ordering	45	24	18	31
3	+Position	43	23	16	35
4	+Pre-Ordering+Position	35	19	14	36
5	+Coverage	41	21	14	**26**
6	+Pre-Ordering+Position+Coverage	**34**	**18**	**12**	27

Table 4: The numbers of under-translation and over-translations produced by different NMT systems.

with coverage mechanism (Row 7), the number of under-translation further decreases. The most important is that it can reduce 9 over-translation cases (from 36 to 27). Fig. 3 shows an example, in which source words in red and source words in blue need to exchange their positions during translation. The result is that the baseline translates the blue words while drops the red ones. Although our approach using pre-ordering and position embedding can fix this under-translation while produces a new over-translation (in green). Fortunately, when we add the coverage mechanism, this over-translation is rectified.

Overall, our approach can substantially reduce the under-translation cases by 30.4%. As a comparison, the system only using coverage reduces 17.4%. For the over-translations, our approach achieves a similar improvement with the coverage model.

6. Related Work

Our work exploits pre-ordering for NMT to improve the under-translation and over-translation. There are two closely related studies:

Improving the under-translation and over-translation. Some previous works attribute the problems of the under-translation and over-translation to the lack of coverage mechanism. Thus they introduce coverage mechanism to NMT. (Tu et al., 2016) maintains a coverage vector at each decode step to collect the attention record, then uses coverage vector to adjust the attention in next time step. (Mi et al., 2016) also maintains a coverage vector, and the difference is that their model introduces a specific coverage embedding for each source word. Further (Tu et al., 2017) proposes a reconstructor for NMT, which can ensure that the information in the source side can be adequately transformed to target side. (Feng et al., 2016) attributes this problem to the lack of explicit distortion and fertility in NMT, and they propose a recurrent attention mechanism to model distortion and fertility. Different from the above methods, we treat this problem with another perspective, as we observe that the words need to be reordered during translation are more likely to be ignored by NMT. Thus we exploit the pre-ordering for NMT to alleviate this problem.

Exploiting techniques in SMT for NMT. Our work is also inspired by the works which incorporating the techniques in SMT to NMT. The earlier related work is conducted on the SMT framework, which is deeply discussed in the reviewed paper (Zhang and Zong, 2015). Here, we only focus on the work which combines the SMT and NMT on NMT framework. Specifically, (Arthur et al., 2016) incorporates word translation table in attention part to adjust the final loss. (Zhang and Zong, 2016) moves forward further by incorporating a bilingual dictionaries in NMT. (Stahlberg et al., 2016) and (He et al., 2016) rescore word candidates with SMT features. (Gülçehre et al., 2015) improves the beam search with language model. (Zhou et al., 2017) proposes a neural combination model to fuse the NMT translation results and SMT translation results. (Wang et al., 2017) improves the NMT system with the SMT recommendations. (Zhang et al., 2014) proposes bilingually-constrained recursive auto-encoders to learn phrase embeddings, which can distinguish the phrases with different semantic meanings. (Tang et al., 2016) explores the possibility to incorporate phrase memory into NMT, in which the decoder can generate a sequence of multiple words all at once.

In this work, we exploit another new technique in SMT, pre-ordering, to NMT to improve the translation performance.

7. Conclusions and Future Work

We have exploited the pre-ordering approach to alleviate the under-translation problem in NMT. Specifically, we pre-order the source sentences to make their word order more consist with the word order in target. Then, we enhance the monotone translation by using the position embedding. Finally, we augment our model with the coverage mechanism.

Our empirical experiments on Chinese-English translation show that the proposed approach can significantly improve the translation quality and substantially reduce the under-translation cases.

However, the under-translation and over-translation problems are still unsolved. In our future work, we plan to propose more effective methods to alleviate the problems. For example, we will design more accurate pre-ordering approaches.

8. Acknowledgments

The research work in this paper has been supported by the National Key Research and Development Program of China under Grant No. 2016QY02D0303 and the Natural Science Foundation of China under Grant No. 61333018 and 61673380.

9. Bibliographical References

Arthur, P., Neubig, G., and Nakamura, S. (2016). Incorporating discrete translation lexicons into neural machine translation. *In Proceedings of EMNLP 2016*, pages 1557–1567.

Bahdanau, D., Cho, K., and Bengio, Y. (2015). Neural machine translation by jointly learning to align and translate. *In Proceedings of ICLR 2015*.

Cho, K., Gulcehre, B. v. M. C., Bahdanau, D., Schwenk, F. B. H., and Bengio, Y. (2014). Learning phrase representations using rnn encoder–decoder for statistical machine translation. *In Proceedings of EMNLP 2014*.

Cohn, T., Hoang, C. D. V., Vymolova, E., Yao, K., Dyer, C., and Haffari, G. (2016). Incorporating structural alignment biases into an attentional neural translation model. *In Proceedings of NAACL 2016*, pages 876–885.

Collins, M., Koehn, P., and Ivona, K. (2005). Clause restructuring for statistical machine translation. *In Proceedings of ACL 2005*, pages 531–540.

Feng, S., Liu, S., Li, M., and Zhou, M. (2016). Implicit distortion and fertility models for attention-based encoder-decoder nmt model. *arXiv preprint arXiv:1601.03317*.

Gehring, J., Auli, M., Grangier, D., Yarats, D., and Dauphin, Y. N. (2017). Convolutional sequence to sequence learning. *arXiv preprint arXiv:1601.03317*.

Genzel, D. (2010). Automatically learning source-side reordering rules for large scale machine translation. *In Proceedings of COLING 2010*, pages 376–384.

Gülçehre, C., Firat, O., Xu, K., Cho, K., Barrault, L., Lin, H.-C., Bougares, F., Schwenk, H., and Bengio, Y. (2015). On using monolingual corpora in neural machine translation. *CoRR, abs/1503.03535*.

He, W., He, Z., Wu, H., and Wang, H. (2016). Improved neural machine translation with smt features. *In Proceedings of AAAI 2016*.

Hitschler, J., Laura, J., Sariya, K., Mayumi, O., Benjamin, K., and Stefan, R. (2016). Otedama: Fast rule-based pre-ordering for machine translation. *The Prague Bulletin of Mathematical Linguistics*, 106:159–168.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8):1735–1780.

Junczys-Dowmunt, M., Dwojak, T., and Hoang, H. (2016). Is neural machine translation ready for deployment? a case study on 30 translation directions. *arXiv preprint arXiv:1610.01108*.

Kalchbrenner, N. and Blunsom, P. (2013). Recurrent continuous translation models. *In Proceedings of EMNLP 2013*.

Koehn, P., Och, F. J., and Marcu, D. (2003). Statistical phrase-based translation. *In Proceedings of HLT/NAACL 2003*, pages 48–54.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., et al. (2007). Moses: Open source toolkit for statistical machine translation. *In Proceedings of ACL 2007*, pages 177–180.

Luong, M.-T., Pham, H., and Manning, C. D. (2015). Effective approaches to attention-based neural machine translation. *In Proceedings of EMNLP 2015*.

Mi, H., Sankaran, B., Wang, Z., and Ittycheriah, A. (2016). A coverage embedding model for neural machine translation. *In Proceedings of EMNLP 2016*.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). Bleu: a method for automatic evaluation of machine translation. *In Proceedings of ACL 2002*, pages 311–318.

Stahlberg, F., Hasler, E., Waite, A., and Byrne, B. (2016). Syntactically guided neural machine translation. *arXiv preprint arXiv:1605.04569*.

Sutskever, I., Vinyals, O., and Le, Q. V. (2014). Sequence to sequence learning with neural networks. *In Proceedings of NIPS 2014*.

Tang, Y., Meng, F., Lu, Z., Li, H., and Yu, P. L. (2016). Neural machine translation with external phrase memory. *arXiv preprint arXiv:1606.01792*.

Tu, Z., Lu, Z., Liu, Y., Liu, X., and Li, H. (2016). Coverage-based neural machine translation. *In Proceedings of ACL 2016*.

Tu, Z., Liu, Y., Shang, L., Liu, Xiaohua, and Li, H. (2017). Neural machine translation with reconstruction. In *Proceedings of AAAI 2017*.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., and Kaiser, Å. (2017). Attention is all you need. *arXiv preprint arXiv:1601.03317*.

Wang, X., Lu, Z., Tu, Z., Li, H., Xiong, D., and Zhang, M. (2017). Neural machine translation advised by statistical machine translation. *In Proceedings of AAAI 2017*.

Wu, Y., Schuster, M., Chen, Z., Le, Q. V., Norouzi, M., Macherey, W., Krikun, M., Cao, Y., Gao, Q., Macherey, K., et al. (2016). Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*.

Zhang, J. and Zong, C. (2009). A framework for effectively integrating hard and soft syntactic rule into phrase based translation. *In Proceedings of PACLIC*, pages 579–588.

Zhang, J. and Zong, C. (2015). Deep neural networks in machine translation: An overview. *IEEE Intelligent Systems*, 30(5):16–25.

Zhang, J. and Zong, C. (2016). Bridging neural machine

translation and bilingual dictionaries. *arXiv preprint arXiv:1610.07272*.

Zhang, J., Liu, S., Li, M., Zhou, M., and Zong, C. (2014). Bilingually-constrained phrase embeddings for machine translation. *In Proceedings of ACL*, pages 111–121.

Zhou, L., Hu, W., Zhang, J., and Zong, C. (2017). Neural system combination for machine translation. *In Proceedings of ACL 2017*, pages 378–384.

Zoph, B. and Knight, K. (2016). Multi-source neural translation. *In Proceedings of NAACL 2016*, pages 30–34.