Self-Attention with Cross-Lingual Position Representation

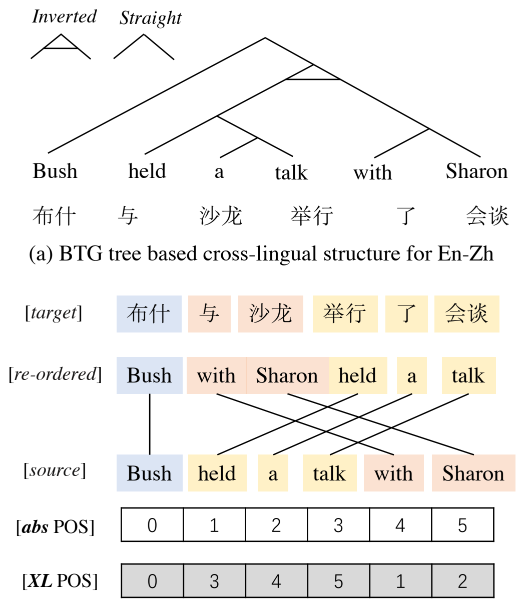
Liang Ding[†] **Longyue Wang**[‡] **Dacheng Tao**[†]
[†]UBTECH Sydney AI Centre, School of Computer Science,
Faculty of Engineering, The University of Sydney
{ldin3097,dacheng.tao}@sydney.edu.au
[‡]Tencent AI Lab
vinnylywang@tencent.com

Abstract

Position encoding (PE), an essential part of self-attention networks (SANs), is used to preserve the word order information for natural language processing tasks, generating fixed position indices for input sequences. However, in cross-lingual scenarios, *e.g.,* machine translation, the PEs of source and target sentences are modeled independently. Due to word order divergences in different languages, modeling the cross-lingual positional relationships might help SANs tackle this problem. In this paper, we augment SANs with *cross-lingual position representations* to model the bilingually aware latent structure for the input sentence. Specifically, we utilize bracketing transduction grammar (BTG)-based reordering information to encourage SANs to learn bilingual diagonal alignments. Experimental results on WMT'14 English⇒German, WAT'17 Japanese⇒English, and WMT'17 Chinese⇔English translation tasks demonstrate that our approach significantly and consistently improves translation quality over strong baselines. Extensive analyses confirm that the performance gains come from the cross-lingual information.

1 Introduction

Although self-attention networks (SANs) (Lin et al., 2017) have achieved the state-of-the-art performance on several natural language processing (NLP) tasks (Vaswani et al., 2017; Devlin et al., 2019; Radford et al., 2018), they possess the innate disadvantage of sequential modeling due to the lack of positional information. Therefore, absolute position encoding (APE) (Vaswani et al., 2017) and relative position encoding (RPE) (Shaw et al., 2018) were introduced to better capture the sequential dependencies. However, either absolute or relative PE is language-independent and its embedding remains fixed. This inhibits the capacity of SANs when modelling multiple languages, which have diverse word orders and structures (Gell-Mann and Ruhlen, 2011). Recent work have shown that modeling cross-lingual information (*e.g.,* alignment or reordering) at encoder or attention level improves translation performance for different language pairs (Cohn et al., 2016; Du and Way, 2017; Zhao et al., 2018; Kawara et al., 2018).


Inverted
Straight
Bush
held
a
talk
with
Sharon
布什
与
沙龙
举行
了
会谈

(a) BTG tree based cross-lingual structure for En-Zh

[*target*] 布什 与 沙龙 举行 了 会谈
[*re-ordered*] Bush with Sharon held a talk
[*source*] Bush held a talk with Sharon
[***abs*** POS] 0 1 2 3 4 5
[***XL*** POS] 0 3 4 5 1 2

(b) Absolute(***abs***) Position *vs*. Cross-Lingual(***XL***) Position

Figure 1: Illustration of cross-lingual position for English⇒Chinese translation task. (a) BTG tree shows the cross-lingual preordering. The top-left corner is the transduction grammar. (b) the difference between absolute position encoding (APE) and our proposed cross-lingual position encoding (XL PE) .

Inspired by their work, we propose to augment SANs with *cross-lingual representations*, by encoding reordering indices at embedding level. Taking English⇒Chinese translation task for example, we first reorder the English sentence by deriving a latent bracketing transduction grammar

Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 1679–1685
July 5 - 10, 2020. ©2020 Association for Computational Linguistics

(BTG) tree (Wu, 1997) (Fig. 1a). Similar to absolute position, the reordering information can be represented as cross-lingual position (Fig. 1b). In addition, we propose two strategies to incorporate cross-lingual position encoding into SANs. We conducted experiments on three commonly-cited datasets of machine translation. Results show that exploiting cross-lingual PE consistently improves translation quality . Further analysis reveals that our method improves the alignment quality (§Sec. 4.3) and context-free Transformer (Tang et al., 2019) (§Sec. 4.4). Furthermore, contrastive evaluation demonstrates that NMT models benefits from the cross-lingual information rather than denoising ability (§Sec. 4.5).

2 Background

Position Encoding To tackle the position unaware problem, absolute position information is injected into the SANs:

$$\mathbf{PE}_{abs} = f(pos_{abs}/10000^{2i/d_{model}}) \quad (1)$$

where pos_{abs} denotes the numerical position indices, i is the dimension of the position indices and d_{model} means hidden size. $f(\cdot)$ alternately employs $sin(\cdot)$ and $cos(\cdot)$ for even and odd dimensions. Accordingly, the position matrix $\mathbf{PE}$ can be obtained given the input $\mathbf{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_T\} \in \mathbb{R}^{T \times d_{model}}$. Then, the position aware output $\mathbf{Z}$ is calculated by:

$$\mathbf{Z} = \mathbf{X} + \mathbf{PE}_{abs} \qquad \in \mathbb{R}^{T \times d_{model}} \quad (2)$$

Self-Attention The SANs compute the attention of each pair of elements in parallel. It first converts the input into three matrices $\mathbf{Q}, \mathbf{K}, \mathbf{V}$, representing queries, keys, and values, respectively:

$$\{\mathbf{Q}, \mathbf{K}, \mathbf{V}\} = \{\mathbf{Z}\mathbf{W}_Q, \mathbf{Z}\mathbf{W}_K, \mathbf{Z}\mathbf{W}_V\} \quad (3)$$

where $\mathbf{W}_Q, \mathbf{W}_K, \mathbf{W}_V \in \mathbb{R}^{d_{model} \times d_{model}}$ are parameter matrices. The output is then computed as a weighted sum of values by $\textsc{Att}(\mathbf{Q}, \mathbf{K}, \mathbf{V})$. SANs can be implemented with multi-head attention mechanism, which requires extra splitting and concatenation operations. Specifically, $\mathbf{W}_Q, \mathbf{W}_K, \mathbf{W}_V$ and $\mathbf{Q}, \mathbf{K}, \mathbf{V}$ in Eq. (3) is split into H sub-matrices, yielding H heads. For the h-th head, the output is computed by:

$$\mathbf{O}_h = \textsc{Att}(\mathbf{Q}_h, \mathbf{K}_h, \mathbf{V}_h) \in \mathbb{R}^{T \times d_v} \quad (4)$$

Where subspace parameters are $\mathbf{W}^h_Q, \mathbf{W}^h_K \in \mathbb{R}^{d_{model} \times d_k}$ and $\mathbf{W}^h_V \in \mathbb{R}^{d_{model} \times d_v}$, where d_k, d_v refer to the dimensions of keys and values in the subspace, and normally $d_k = d_v = d_{model}/\mathrm{H}$. Finally, these subspaces are combined with concatenation operation:

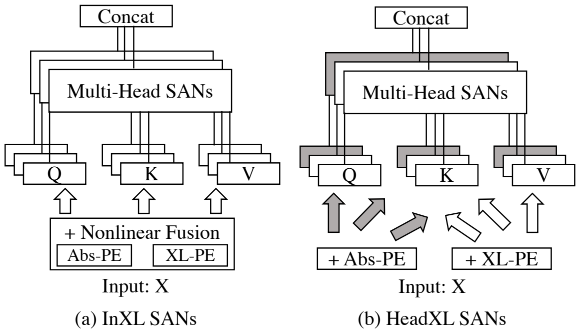


Figure 2: The proposed integration strategies.

$$\mathbf{O} = \textsc{concat}(\mathbf{O}_1, \dots, \mathbf{O}_H)\mathbf{W}_O \quad (5)$$

where $\mathbf{W}_O \in \mathbb{R}^{Hd_v \times d_{model}}$ and $\mathbf{O} \in \mathbb{R}^{T \times d_{model}}$ are the parameter matrix and output, respectively.

3 Approach

3.1 Cross-Lingual Position Representation

First, we built a BTG-based reordering model (Neubig et al., 2012) to generate a reordered source sentence according to the word order of its corresponding target sentence. Second, we obtained the reordered word indices pos_{XL} that correspond with the input sentence $\mathbf{X}$. To output the cross-lingual position matrix $\mathbf{PE}_{\text{XL}}$, we inherit the sinusoidal function in Eq. (1). Formally, the process is:

$$\mathbf{PE}_{\text{XL}} = f(\text{BTG}(\mathbf{X})) \quad (6)$$

3.2 Integration Strategy

As shown in Fig. 2, we propose two strategies to integrate the cross-lingual position encoding (XL PE) into SANs: inputting-level XL (**InXL**) SANs and head-level (**HeadXL**) SANs.

Inputting-level XL SANs As illustrated in Fig. 2a, we employ a non-linear function $\textsc{Tanh}(\cdot)$ to fuse $\mathbf{PE}_{abs}$ and $\mathbf{PE}_{\text{XL}}$:

$$\mathbf{PE}_{\text{IN-XL}} = \textsc{Tanh}(\mathbf{PE}_{abs}\mathbf{U} + \mathbf{PE}_{\text{XL}}\mathbf{V}) \quad (7)$$

where $\mathbf{U}, \mathbf{V}$ are trainable parameters. In our preliminary experiments, the non-linear function performs better than element-wise addition. This might because complex non-linear one have better fitting capabilities, thereby avoiding exceptional reordering to some extent. Next, we perform Eq. (2) to obtain the output representations:

$$\mathbf{Z}_{\text{IN-XL}} = \mathbf{X} + \mathbf{PE}_{\text{IN-XL}} \tag{8}$$

Similarly, we use Eq. (3)∼(5) to calculate multiple heads of SANs.

Head-level XL SANs Instead of projecting XL PE to all attention heads, we feed partial of them, such that some heads contain XL PE and others contain APE, namely HeadXL. As shown in Fig. 2b, we fist add APE and XL PE for $\mathbf{X}$, respectively:

$$\begin{aligned} \mathbf{Z}_{abs} &= \mathbf{X} + \mathbf{PE}_{abs} \\ \mathbf{Z}_{\text{XL}} &= \mathbf{X} + \mathbf{PE}_{\text{XL}} \end{aligned} \tag{9}$$

We denote the number of XL PE equipped heads as $\tau \in \{0, \ldots, H\}$. To perform the attention calculation, $\mathbf{W}_i$ is divided into $[\mathbf{W}_i^{\text{XL}} \in \mathbb{R}^{d_{model} \times \tau d_v}; \mathbf{W}_i^{abs} \in \mathbb{R}^{d_{model} \times (H-\tau) d_v}]$ for each $i \in \mathbf{Q}, \mathbf{K}, \mathbf{V}$, correspondingly generating two types of $\{\mathbf{Q}, \mathbf{K}, \mathbf{V}\}$ for XL PE heads and APE heads. According to Eq. (4), the output of each XL PE head is:

$$\mathbf{O}_h^{\text{XL}} = \textsc{Att}(\mathbf{Q}_h^{\text{XL}}, \mathbf{K}_h^{\text{XL}}, \mathbf{V}_h^{\text{XL}}) \in \mathbb{R}^{T \times d_v} \tag{10}$$

As a result, the final output of HeadXL is:

$$\begin{aligned} \textsc{HeadSan}(\mathrm{X}) = &\textsc{Concat}(\mathbf{O}_1^{\text{XL}}, \ldots, \mathbf{O}_\tau^{\text{XL}} \\ &\mathbf{O}_{\tau+1}^{abs}, \ldots, \mathbf{O}_H^{abs}) \mathbf{W}_O \end{aligned} \tag{11}$$

In particular, $\tau = 0$ refers to the original Transformer (Vaswani et al., 2017) and $\tau = H$ means that XL PE will propagate over all attention heads.

4 Experiments

We conduct experiments on word order-diverse language pairs: WMT'14 English⇒German (En-De), WAT'17 Japanese⇒English (Ja-En), and WMT'17 Chinese⇔English (Zh-En & En-Zh).

For English⇒German, the training set consists of 4.5 million sentence pairs and newstest2013 & 2014 are used as the dev. and test sets, respectively. BPE with 32K merge operations is used to handle low-frequency words. For Japanese⇒English, we follow Morishita et al. (2017) to use the first two sections as training data, which consists of 2.0 million sentence pairs. The dev. and test sets contain 1790 and 1812 sentences. For Chinese⇔English, we follow Hassan et al. (2018) to get 20 million sentence pairs. We develop on devtest2017 and test on newstest2017. We use SacreBLEU (Post, 2018) as the evaluation metric with statistical significance test (Collins et al., 2005).

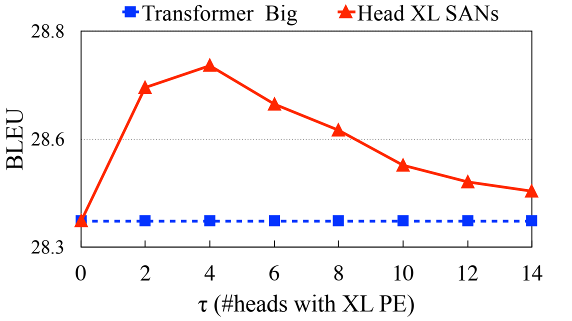


Figure 3: BLEU score on newstest2014 for different τ.

We evaluate the proposed XL PE strategies on Transformer. The baseline systems include Relative PE (Shaw et al., 2018) and directional SAN (DiSAN, Shen et al. 2018). We implement them on top of OpenNMT (Klein et al., 2017). In addition, we report the results of previous studies (Hao et al., 2019; Wang et al., 2019; Chen et al., 2019b,a; Du and Way, 2017; Hassan et al., 2018).

The reordered source sentences are generated by BTG-based preordering model (Neubig et al., 2012) trained with above sub-word level[1] parallel corpus. At training phase, we first obtain word alignments from parallel data using GIZA++ or FastAlign, and then the training process is to find the optimal BTG tree for source sentence consistent with the order of the target sentence based on the word alignments and parallel data. At decoding phase, we only provide source sentences as input and the model can output reordering indices, which will be fed into NMT model. Thus, bilingual alignment information is only used to preprocess training data, but not necessary at decoding time.

For fair comparison, we keep the Transformer decoder unchanged and validate different position representation strategies on the encoder. We conduct all experiments on the Transformer-Big with four V100 GPUs.

4.1 Effect of τ in HeadXL SANs

Fig. 3 reports the results of different τ for Head XL SANs. With increasing of XL PE-informed heads, the best BLEU is achieved when #heads = 4, which is therefore left as the default setting for HeadXL. Then, the BLEU score gradually decreases as the

[1] Garg et al. (2019) show that sub-word units are beneficial for statistical model.

#	System	Architecture	BLEU	#Param.
1	Vaswani et al. (2017)	Transformer BIG	28.4	213M
2	Hao et al. (2019)	Transformer BIG w/ BiARN	28.98	323.5M
3	Wang et al. (2019)	Transformer BIG w/ Structure PE	28.88	–
4	Chen et al. (2019b)	Transformer BIG w/ MPRHead	29.11	289.1M
5	Chen et al. (2019a)	Transformer BIG w/ Reorder Emb	29.11	308.2M
6	This work	Transformer BIG	28.36	282.55M
7		+ Relative PE	28.71	+0.06M
8		+ DiSAN	28.76	+0.04M
9		+ InXL PE	28.66	+0.01M
10		+ HeadXL PE	28.72	+0.00M
11		+ Combination	29.05$^{\uparrow}$	+0.01M

Table 1: Experiments on WMT'14 En-De. "↑"indicates significant difference ($p < 0.01$) from Transformer BIG. "#Param" denotes the number of parameters. "+ Combination" represents combining #9 and #10 methods.

System	JaEn	ZhEn	EnZh
Du and Way (2017)	25.65	–	–
Hassan et al. (2018)	–	24.20	–
Transformer BIG	29.22	23.94	33.79
+ Relative PE	29.62	24.36	34.21
+ DiSAN	29.73	24.44	34.31
+ InXL PE	29.52	24.44	34.23
+ HeadXL PE	29.62	24.39	34.20
+ Combination$^{\uparrow}$	29.85	24.71	34.51

Table 2: Experiments on Ja-En, Zh-En and En-Zh.

Model	AER	P	R
Transformer BIG	29.7%	69.9%	72.7%
+ InXL	27.5%	72.2%	74.1%
+ HeadXL	26.9%	75.4%	73.9%
+ Combination	24.7%	75.0%	77.6%

Table 3: The AER scores of alignments on En-De.

number of APE-informed heads decrease (τ ↑), indicating that sequential position embedding is still essential for SANs.

4.2 Main Results

Tab. 1 shows the results on En-De, inputting-level cross-lingual PE (+InXL PE) and head-level cross-lingual PE (+HeadXL PE) outperform Transformer BIG by 0.30 and 0.36 BLEU points, and combining these two strategies[2] achieves a 0.69 BLEU point increase. For Ja-En, Zh-En, and En-Zh (Tab. 2), we observe a similar phenomenon, demonstrating that XL PE on SANs do improve the translation performance for several language pairs. It is worth noting that our approach introduces nearly no additional parameters (+0.01M over 282.55M).

4.3 Alignment Quality

Our proposed XL PE intuitively encourages SANs to learn bilingual diagonal alignment, so has the potential to induce better attention matrices. We explore this hypothesis on the widely used Gold Alignment dataset[3] and follow Tang et al. (2019) to perform the alignment. The only difference being that we average the attention matrices across all heads from the penultimate layer (Garg et al., 2019). The alignment error rate (AER, Och and Ney 2003), precision (P) and recall (R) are reported as the evaluation metrics. Tab. 3 summarizes the results. We can see: 1) XL PE allows SANs to learn better attention matrices, thereby improving alignment performance (27.4 / 26.9 vs. 29.7); and 2) combining the two strategies delivers consistent improvements (24.7 vs. 29.7).

4.4 Gain for Context-Free Model

Tang et al. (2019) showed that context-free Transformer (directly propagating the source word embeddings with PE to the decoder) achieved comparable results to the best RNN-based model. We argue that XL PE could further enhance the context-free Transformer. On English⇒German dataset,

[2] Replace $\mathbf{PE}_{\text{XL}}$ in Eq. (9) with $\mathbf{PE}_{\text{IN-XL}}$ in Eq. (8).

[3] http://www-i6.informatik.rwth-aachen.de/goldAlignment, the original dataset is German-English, we reverse it to English-German.

System	BLEU	#Param.
LSTM (6 layers)	24.12	178.90M
Big-noEnc-noPos	9.97	171.58M
+ Absolute PE	24.11	+0.00M
+ Relative PE	24.47	+0.01M
+ InXL PE	24.68	+0.01M

Table 4: Gains over Encoder-Free Transformer.

we compare LSTM-based model, Transformer Big-noenc-nopos, +APE, +RPE and +InXL PE. For fair comparison, we set the LSTM hidden size to 1024. In Tab. 4, we can see: 1) position information is the most important component for the context-free model, bringing +14.45 average improvement; 2) InXL PE equipped context-free Transformer significantly outperforms the LSTM model while consuming less parameters; and 3) compared to the increment on standard Transformer (+0.30 over 28.36), InXL PE improves more for context-free Transformer (+0.57 over 24.11), where the improvements are +2.3% vs. +1.1%.

4.5 Effects of Noisy Reordering Information

To demonstrate that our improvements come from cross-lingual position information rather than noisy position signals, we attack our model by adding noises[4] into reordered indices of training sentences. As shown in Fig. 4, our method can tolerate partial reordering noises and maintain performance to some extent. However, as noise increases, translation quality deteriorates, indicating that noises in reordering information do not work as regularization. This contrastive evaluation also confirms that the model does not benefit from the noise as much as it benefits from the reordering information.

5 Related Work

Augmenting SANs with position representation SANs ignore the position of each token due to its position-unaware "bag-of-words" assumption. The most straightforward strategy is adding the position representations as part of the token representations (Vaswani et al., 2017; Shaw et al., 2018). Besides above sequential PE approaches, Wang et al. (2019) enhanced SANs with structural positions extracted from the syntax dependencies. However, none of them considered modeling the cross-lingual position information between languages.

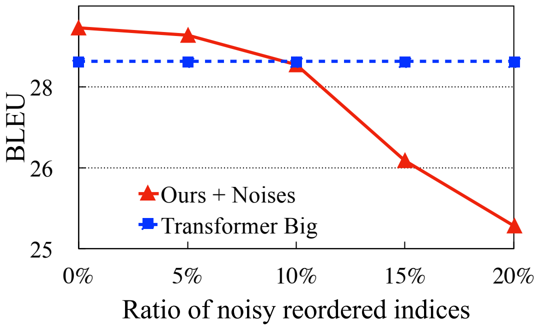


Figure 4: Experiments with noise attacks. Ratio of noisy reordered indices ranges from 0% to 20%.

Modeling cross-lingual divergence There has been many works modeling cross-lingual divergence (*e.g.*, reordering) in statistical machine translation (Nagata et al., 2006; Durrani et al., 2011, 2013). However, it is difficult to migrant them to neural machine translation. Kawara et al. (2018) pre-reordered the source sentences with a recursive neural network model. Chen et al. (2019a) learned the reordering embedding by considering the relationship between the position embedding of a word and SANs-calculated sentence representation. Yang et al. (2019) showed that SANs in machine translation could learn word order mainly due to the PE, indicating that modeling cross-lingual information at position representation level may be informative. Thus, we propose a novel cross-lingual PE method to improve SANs.

6 Conclusions and Future Work

In this paper, we presented a novel cross-lingual position encoding to augment SANs by considering cross-lingual information (*i.e.,* reordering indices) for the input sentence. We designed two strategies to integrate it into SANs. Experiments indicated that the proposed strategies consistently improve the translation performance. In the future, we plan to extend the cross-lingual position encoding to non-autoregressive MT (Gu et al., 2018) and unsupervised NMT (Lample et al., 2018).

Acknowledgments

This work was supported by Australian Research Council Projects FL-170100117. We are grateful to the anonymous reviewers and the area chair for their insightful comments and suggestions.

[4] We randomly swap two reordered positional indexes with different ratios.

References

Kehai Chen, Rui Wang, Masao Utiyama, and Eiichiro Sumita. 2019a. Neural machine translation with reordering embeddings. In *ACL*.

Kehai Chen, Rui Wang, Masao Utiyama, and Eiichiro Sumita. 2019b. Recurrent positional embedding for neural machine translation. In *EMNLP*.

Trevor Cohn, Cong Duy Vu Hoang, Ekaterina Vymolova, Kaisheng Yao, Chris Dyer, and Gholamreza Haffari. 2016. Incorporating structural alignment biases into an attentional neural translation model. In *NAACL*.

Michael Collins, Philipp Koehn, and Ivona Kučerová. 2005. Clause restructuring for statistical machine translation. In *ACL*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*.

Jinhua Du and Andy Way. 2017. Pre-reordering for neural machine translation: Helpful or harmful? *The Prague Bulletin of Mathematical Linguistics*, 108(1).

Nadir Durrani, Alexander Fraser, Helmut Schmid, Hieu Hoang, and Philipp Koehn. 2013. Can Markov models over minimal translation units help phrase-based SMT? In *ACL*.

Nadir Durrani, Helmut Schmid, and Alexander Fraser. 2011. A joint sequence translation model with integrated reordering. In *ACL*.

Sarthak Garg, Stephan Peitz, Udhyakumar Nallasamy, and Matthias Paulik. 2019. Jointly learning to align and translate with transformer models. In *EMNLP*.

Murray Gell-Mann and Merritt Ruhlen. 2011. The origin and evolution of word order. *Proceedings of the National Academy of Sciences*, 108(42).

Jiatao Gu, James Bradbury, Caiming Xiong, Victor O.K. Li, and Richard Socher. 2018. Non-autoregressive neural machine translation. In *ICLR*.

Jie Hao, Xing Wang, Baosong Yang, Longyue Wang, Jinfeng Zhang, and Zhaopeng Tu. 2019. Modeling recurrence for transformer. In *NAACL*.

Hany Hassan, Anthony Aue, Chang Chen, Vishal Chowdhary, Jonathan Clark, Christian Federmann, Xuedong Huang, Marcin Junczys-Dowmunt, William Lewis, Mu Li, et al. 2018. Achieving human parity on automatic chinese to english news translation. *arXiv*.

Yuki Kawara, Chenhui Chu, and Yuki Arase. 2018. Recursive neural network based preordering for english-to-japanese machine translation. In *ACL, Student Research Workshop*.

Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander Rush. 2017. Opennmt: Open-source toolkit for neural machine translation. In *ACL*.

Guillaume Lample, Myle Ott, Alexis Conneau, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2018. Phrase-based & neural unsupervised machine translation. In *EMNLP*.

Zhouhan Lin, Minwei Feng, Cicero Nogueira dos Santos, Mo Yu, Bing Xiang, Bowen Zhou, and Yoshua Bengio. 2017. A structured self-attentive sentence embedding. In *ICLR*.

Makoto Morishita, Jun Suzuki, and Masaaki Nagata. 2017. Ntt neural machine translation systems at wat 2017. *IJCNLP*.

Masaaki Nagata, Kuniko Saito, Kazuhide Yamamoto, and Kazuteru Ohashi. 2006. A clustered global phrase reordering model for statistical machine translation. In *COLING*.

Graham Neubig, Taro Watanabe, and Shinsuke Mori. 2012. Inducing a discriminative parser to optimize machine translation reordering. In *EMNLP*.

Franz Josef Och and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Computational linguistics*, 29(1).

Matt Post. 2018. A call for clarity in reporting bleu scores. In *WMT*.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training.

Peter Shaw, Jakob Uszkoreit, and Ashish Vaswani. 2018. Self-attention with relative position representations. In *NAACL*.

Tao Shen, Tianyi Zhou, Guodong Long, Jing Jiang, Shirui Pan, and Chengqi Zhang. 2018. Disan: Directional self-attention network for rnn/cnn-free language understanding. In *AAAI*.

Gongbo Tang, Rico Sennrich, and Joakim Nivre. 2019. Understanding neural machine translation by simplification: The case of encoder-free models. In *RANLP*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NeurIPS*.

Xing Wang, Zhaopeng Tu, Longyue Wang, and Shuming Shi. 2019. Self-attention with structural position representations. In *EMNLP*.

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

Baosong Yang, Longyue Wang, Derek F. Wong, Lidia S. Chao, and Zhaopeng Tu. 2019. Assessing the ability of self-attention networks to learn word order. In *ACL*.

Yang Zhao, Jiajun Zhang, and Chengqing Zong. 2018. Exploiting pre-ordering for neural machine translation. In *LREC*.