Self-Attention with Structural Position Representations

Xing Wang
Tencent AI Lab
brightxwang@tencent.com

Zhaopeng Tu
Tencent AI Lab
zptu@tencent.com

Longyue Wang
Tencent AI Lab
vinnylywang@tencent.com

Shuming Shi
Tencent AI Lab
shumingshi@tencent.com

Abstract

Although self-attention networks (SANs) have advanced the state-of-the-art on various NLP tasks, one criticism of SANs is their ability of encoding positions of input words (Shaw et al., 2018). In this work, we propose to augment SANs with *structural position representations* to model the latent structure of the input sentence, which is complementary to the standard sequential positional representations. Specifically, we use dependency tree to represent the grammatical structure of a sentence, and propose two strategies to encode the positional relationships among words in the dependency tree. Experimental results on NIST Chinese⇒English and WMT14 English⇒German translation tasks show that the proposed approach consistently boosts performance over both the absolute and relative sequential position representations.

1 Introduction

In recent years, self-attention networks (SANs, Parikh et al., 2016; Lin et al., 2017) have achieved the state-of-the-art results on a variety of NLP tasks (Vaswani et al., 2017; Strubell et al., 2018; Devlin et al., 2019). SANs perform the attention operation under the position-unaware "bag-of-words" assumption, in which positions of the input words are ignored. Therefore, absolute position (Vaswani et al., 2017) or relative position (Shaw et al., 2018) are generally used to capture the sequential order of words in the sentence. However, several researches reveal that the sequential structure may not be sufficient for NLP tasks (Tai et al., 2015; Kim et al., 2017; Shen et al., 2019), since sentences inherently have hierarchical structures (Chomsky, 1965; Bever, 1970).

In response to this problem, we propose to augment SANs with *structural position representations* to capture the hierarchical structure of the input sentence. The starting point for our approach is a recent finding: the latent structure of a sentence can be captured by structural depths and distances (Hewitt and Manning, 2019). Accordingly, we propose *absolute structural position* to encode the depth of each word in a parsing tree, and *relative structural position* to encode the distance of each word pair in the tree.

We implement our structural encoding strategies on top of TRANSFORMER (Vaswani et al., 2017) and conduct experiments on both NIST Chinese⇒English and WMT14 English⇒German translation tasks. Experimental results show that exploiting structural position encoding strategies consistently boosts performance over both the absolute and relative sequential position representations across language pairs. Linguistic analyses (Conneau et al., 2018) reveal that the proposed structural position representation improves the translation performance with richer syntactic information. Our main contributions are:

- Our study demonstrates the necessity and effectiveness of exploiting structural position encoding for SANs, which benefits from modeling syntactic depth and distance under the latent structure of the sentence.

- We propose structural position representations for SANs to encode the latent structure of the input sentence, which are complementary to their sequential counterparts.

2 Background

Self-Attention SANs produce representations by applying attention to each pair of elements from the input sequence, regardless of their distance. Given an input sequence $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_I\} \in \mathbb{R}^{I \times d}$, the model first transforms

	Bush	held	a	talk	with	Sharon
Absolute Position	0	1	2	3	4	5
Relative Position	-3	-2	-1	0	+1	+2

1	0	2	1	2	1
-2	-1	-1	0	+3	+2

(a) **Sequential** Position Encoding (b) **Structural** Position Encoding
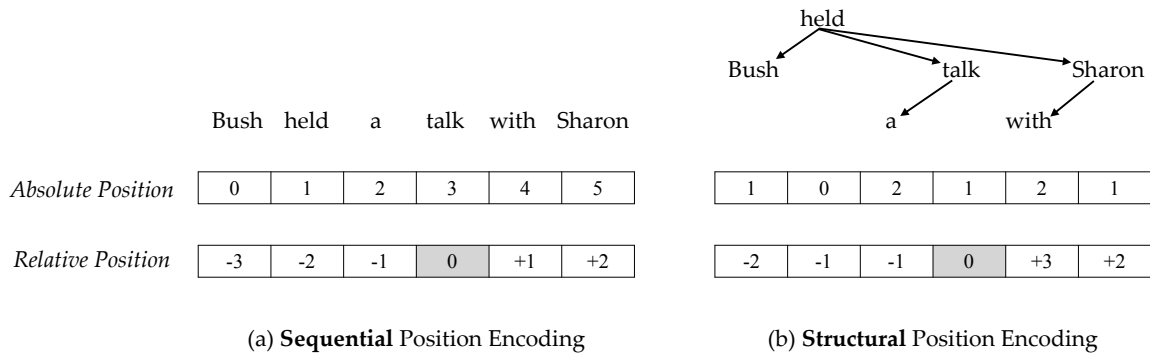
Figure 1: Illustration of (a) the standard *sequential position encoding* (Vaswani et al., 2017; Shaw et al., 2018), and (b) the proposed *structural position encoding*. The relative position in the example is for the word "talk".

it into queries $\mathbf{Q} \in \mathbb{R}^{I \times d}$, keys $\mathbf{K} \in \mathbb{R}^{I \times d}$, and values $\mathbf{V} \in \mathbb{R}^{I \times d}$:

$$\mathbf{Q}, \mathbf{K}, \mathbf{V} = \mathbf{X}\mathbf{W}_Q, \mathbf{X}\mathbf{W}_K, \mathbf{X}\mathbf{W}_V \in \mathbb{R}^{I \times d}, \quad (1)$$

where $\{\mathbf{W}_Q, \mathbf{W}_K, \mathbf{W}_V\} \in \mathbb{R}^{d \times d}$ are trainable parameters and d indicates the hidden size. The output sequence is calculated as

$$\mathbf{O} = \text{ATT}(\mathbf{Q}, \mathbf{K}) \, \mathbf{V} \qquad \in \mathbb{R}^{I \times d}, \quad (2)$$

where $\text{ATT}(\cdot)$ is a dot-product attention model.

Sequential Position Encoding To make use of the order of the sequence, information about the absolute or relative position of the elements in the sequence is injected into SAN:

- *Absolute* Sequential PE (Vaswani et al., 2017) is defined as

$$\text{ABSPE}(abs) = f(abs/10000^{2i/d}), \quad (3)$$

  where abs is the absolute position in the sequence and i is the dimension of position representations. $f(\cdot)$ is $\sin(\cdot)$ for the even dimension, and $\cos(\cdot)$ for the odd dimension.

  Vaswani et al. (2017) propose to conduct element-wise addition to combine the fixed sequential position representation with word embedding and feed the combination representation to the SANs.

- *Relative* Sequential PE (Shaw et al., 2018) is calculated as

$$\text{RELPE}(rel) = \mathbf{R}[rel], \quad (4)$$

  where rel is the relative position to the queried word, which is used to index a learnable matrix $\mathbf{R}$ that represents relative position embeddings.

Shaw et al. (2018) propose relation-aware SANs and take the relative sequential encoding as the additional key and value (Eq.2) in the attention computation.

Figure 1(a) shows an example of absolute (i.e., *abs*) and relative (i.e., *rel*) sequential positions.

3 Approach

3.1 Structural Position Representations

In this study, we choose dependency tree to represent sentence structure for its simplicity on modeling syntactic relationships among input words. Figure 1 shows an example to illustrate the idea of the proposed approach. From the perspective of relationship path between words, sequential PE measures the sequential distance between the words. As shown in Figure 1 (a), for each word, absolute sequential position represents the sequential distance to the beginning of the sentence, while relative sequential position measures the relative distance to the queried word ("talk" in the example).

The latent structure can be interpreted in various ways, from syntactic tree structures, e.g., constituency tree (Collins, 2003) or dependency tree (Kübler et al., 2009), to semantic graph structures, e.g., abstract meaning representation graph (Banarescu et al., 2013). In this work, dependency path, which is induced from the dependency tree, is adopted to provide a new perspective on modelling pairwise relationships.

Figure 1 shows the difference between the sequential path and dependency path. The sequential distance between the two words "held" and "talk" is 2, while their structural distance is only 1 as word "talk" is the dependent of the head "held" (Nivre, 2005).

Absolute Structural Position We exploit the tree depth of the word in the dependency tree as its absolute structural position. Specifically, we treat the main verb (Tapanainen and Jarvinen, 1997) of the sentence as the origin and use the distance of the dependency path from the target word to the origin as the absolute structural position

$$abs_{stru}(x_i) = distance_{tree}(x_i, origin), \quad (5)$$

where x_i is the target word, $tree$ is the given dependency structure and the origin is the main verb of the $tree$.

In the field of NMT, *BPE sub-words* and *end-of-sentence symbol* should be carefully handled as they do not appear in the conventional dependency tree. In this work, we assign the *BPE sub-words* share the absolute structural position of the original word and set the the first larger integer than the max absolute structural position in dependency tree as the absolute structural position of *end-of-sentence symbol*.

Relative Structural Position For the *relative* structural position $rel_{stru}(x_i, x_j)$, we calculate $rel_{stru}(x_i, x_j)$ in the dependency tree following two hierarchical rules:

1. if x_i and x_j are at same dependency edge, $rel_{stru}(x_i, x_j) = abs_{stru}(x_i) - abs_{stru}(x_j)$.

2. if x_i and x_j are at different dependency edges, $rel_{stru}(x_i, x_j) = f_{stru}(i - j) * (abs_{stru}(x_i) + abs_{stru}(x_j))$, where

$$f_{stru}(x) = \begin{cases} 1 & x > 0 \\ 0 & x = 0 \\ -1 & x < 0 \end{cases} \quad (6)$$

Following Shaw et al. (2018), we use clipping distance r to limit the maximum relative position.

3.2 Integrating Structural PE into SANs

We inherit position encoding functions from sequential approaches (Eq.3 and Eq.4) to implement the structural position encoding strategies. Since structural position representations capture complementary position information to their sequential counterparts, we also exploit to integrate the structural position encoding into SANs with the sequential counterparts approaches (Vaswani et al., 2017; Shaw et al., 2018).

For the *absolute* position, we use the nonlinear function to fuse the sequential and structure posi-

tion representations[1]:

$$asb(x_i) = f_{abs}(\text{ABSPE}(abs_{seq}), \\ \text{ABSPE}(abs_{stru})) \quad (7)$$

where f_{abs} is the nonlinear function. $\text{ABSPE}(abs_{seq})$ and $\text{ABSPE}(abs_{stru})$ are absolute sequential and structural position embedding in Eq.3 and Eq.5 respectively.

For the *relative* position, we follow Shaw et al. (2018) to extend the self-attention computation to consider the pairwise relationships and project the relative structural position as described at Eq.(3) and Eq.(4) in Shaw et al. (2018)[2].

4 Related Work

There has been growing interest in improving the representation power of SANs (Dou et al., 2018, 2019; Yang et al., 2018; Wang et al., 2018; Wu et al., 2018; Yang et al., 2019a,b; Sukhbaatar et al., 2019). Among these approaches, a straightforward strategy is that augmenting the SANs with position representations (Shaw et al., 2018; Ma et al., 2019; Bello et al., 2019; Yang et al., 2019c), as the position representations involves element-wise attention computation. In this work, we propose to augment SANs with structural position representations to model the latent structure of the input sentence.

Our work is also related to the structure modeling for SANs, as the proposed model utilizes the dependency tree to generate structural representations. Recently, Hao et al. (2019c,b) integrate the recurrence into the SANs and empirically demonstrate that the hybrid models achieve better performances by modeling structure of sentences. Hao et al. (2019a) further make use of the multi-head attention to form the multi-granularity self-attention, to capture the different granularity phrases in source sentences. The difference is that we treat the position representation as a medium to transfer the structure information from the dependency tree into the SANs.

---

[1]We also use parameter-free element-wise addition method to combine two absolute position embedding and get 0.28 BLEU point improvement on development set of NIST Chinese⇒English over the baseline model that only uses absolute sequential encoding.

[2]Due to the space limitations we do not show these functions. Please refer to Shaw et al. (2018) for more detail.

#	Sequential		Structural		Spd.	BLEU
	Abs.	*Rel.*	*Abs.*	*Rel.*		
1			×	×	2.81	28.33
2	×	×	✓	×	2.53	35.43
3			×	✓	2.65	34.23
4			×	×	3.23	44.31
5	✓	×	✓	×	2.65	44.84
6			✓	✓	2.52	45.10
7			×	×	3.18	45.02
8	✓	✓	✓	×	2.64	45.43
9			✓	✓	2.48	45.67

Table 1: Impact of the position encoding components on Chinese⇒English NIST02 development dataset using Transformer-Base model. "Abs." and "Rel." denote *absolute* and *relative* position encoding, respectively. "Spd." denotes the decoding speed (sentences/second) on a Tesla M40, the speed of structural position encoding strategies include the step of dependency parsing.

5 Experiment

We conduct experiments on the widely used NIST Chinese⇒English and WMT14 English⇒German data, and report the 4-gram BLEU score (Papineni et al., 2002).

Chinese⇒English We use the training dataset consists of about 1.25 million sentence pairs. NIST 2002 (MT02) dataset is used as development set. NIST 2003, 2004, 2005, 2006 datasets are used as test sets. We use byte-pair encoding (BPE) toolkit to alleviate the out-of-vocabulary problem with 32K merge operations.

English⇒German We use the dataset consisting of about 4.5 million sentence pairs as the training set. The newstest2013 and newstest2014 are used as the development set and the test set. We also apply BPE with 32K merge operations to obtain subword unit.

We evaluate the proposed position encoding strategies on TRANSFORMER (Vaswani et al., 2017) and implement them on top of THUMT (Zhang et al., 2017). We use the Stanford parser (Klein and Manning, 2003) to parse the sentences and obtain the structural structural absolute and relative position as described in Section 3. When using relative structural position encoding, we use clipping distance $r = 16$. To make a fair comparison, we valid different position encoding strategies on the encoder and keep the TRANSFORMER decoder unchanged.

We study the variations of the BASE model on Chinese⇒English task, and evaluate the overall performance with the BIG model on both translation tasks.

5.1 Model Variations

We evaluate the importance of the proposed absolute and relative structural position encoding strategies and study the variations with *Transformer-Base* model on Chinese⇒English data. The experimental results on the development set are shown in Table 1.

Effect of Position Encoding We first remove the sequential encoding from the *Transformer* encoder (Model #1) and observe the translation performance degrades dramatically ($28.33 - 44.31 = -15.98$), which demonstrates the necessity of the position encoding strategies.

Effect of Structural Position Encoding Then we valid our proposed structural position encoding strategies over the position-unaware model (Models #2-3). We find that absolute and relative structural position encoding strategies improve the translation performance by 7.10 BLEU points and 5.90 BLEU points respectively, which shows that the introducing of the proposed absolute and relative structural positions improves the translation performance in terms of BLEU score.

Combination of Sequential and Structural Position Encoding Strategies We integrate the absolute and relative structural position encoding strategies into the *Base* model equipped with absolute sequential position encoding (Models #4-6). We observe that the proposed two approaches are able to achieve improvements over the *Base* model with decoding speed marginally decreases.

Finally, we valid the proposed structural position encoding over the *Base* model equipped with absolute and relative sequential position encoding (Models #7-9). We find that sequential relative encoding obtains 0.71 BLEU points improvement (Model #7 vs. Model #4) and structural position encoding achieves a further improvement in performance by 0.65 BLEU points (Model #9 vs. Model #7), demonstrating the effectiveness of the proposed structural position encoding strategies.

5.2 Main Results

We valid the proposed structural encoding strategies over *Transformer-Big* model in

1406

Model Architecture	Zh⇒En					En⇒De
	MT03	MT04	MT05	MT06	Avg	WMT14
Hao et al. (2019c)	-	-	-	-	-	28.98
Transformer-Big	45.30	46.49	45.21	44.87	45.47	28.58
+ *Structural* PE	45.62	47.12$^\uparrow$	45.84	45.64$^\Uparrow$	46.06	28.88
+ *Relative* Sequential PE	45.45	47.01	45.65	45.87$^\Uparrow$	46.00	28.90
+ *Structural* PE	45.85$^\uparrow$	47.37$^\Uparrow$	46.20$^\Uparrow$	46.18$^\Uparrow$	46.40	29.19$^\Uparrow$

Table 2: Evaluation of translation performance on NIST Zh⇒En and WMT14 En⇒De test sets. Hao et al. (2019c) is a *Transformer-Big* model which adopted an additional recurrence encoder with the attentive recurrent network to model syntactic structure. "$\uparrow$ / $\Uparrow$": significant over the *Transformer-Big* ($p < 0.05/0.01$), tested by bootstrap resampling (Koehn, 2004).

Model	Surface			Syntactic				Semantic					
	SeLen	WC	Avg	TrDep	ToCo	BShif	Avg	Tense	SubN	ObjN	SoMo	CoIn	Avg
BASE	92.20	63.00	77.60	44.74	79.02	71.24	65.00	89.24	84.69	84.53	52.13	62.47	74.61
+ *Rel. Seq.* PE	89.82	63.17	76.50	45.09	78.45	71.40	64.98	88.74	87.00	85.53	51.68	62.21	75.03
+ *Stru.* PE	89.54	62.90	76.22	46.12	79.12	72.36	65.87	89.30	85.47	84.94	52.90	62.99	75.12

Table 3: Performance on linguistic probing tasks. The probing tasks were conducted by evaluating linguistics embedded in the *Transformer-Base* encoder outputs. "Base", "+ Rel. Seq. PE", "+ Stru. PE" denote *Transformer-Base*, *Transformer-Base* with relative sequential PE, *Transformer-Base* with relative sequential PE and structural PE models respectively.

Chinese⇒English and English⇒German data, and list the results in Table 2.

For Chinese⇒English, Structural position encoding (+ *Structural* PE) outperforms the *Transformer-Big* by 0.59 BLEU points on average over four NIST test sets. Sequential relative encoding approach (+*Relative* Sequential PE) outperforms the *Transformer-Big* by 0.53 BLEU points, and structural position encoding (+ *Structural* PE) achieves further improvement up to +0.40 BLEU points and outperforms the *Transformer-Big* by 0.93 BLEU points. For English⇒German, similar phenomenon is observed, which reveals that the proposed structural position encoding strategy can consistently boost translation performance over both the absolute and relative sequential position representations.

5.3 Linguistic Probing Evaluation

We conduct probing tasks[3] (Conneau et al., 2018) to evaluate structure knowledge embedded in the encoder output in the variations of the *Base* model that are trained on En⇒De translation task.

We follow Wang et al. (2019) to set model configurations. The experimental results on probing tasks are shown in Table 3, and the BLEU scores of "Base", "+ Rel. Seq. PE", "+ Stru. PE" are

27.31, 27.99 and 28.30. From the table, we can see 1) adding the relative sequential positional embedding achieves improvement over the baseline on semantic tasks (75.03 vs. 74.61). This may indicate the model benefits more from semantic modeling; 2) with the structural positional embedding, the model obtains improvement on syntactic tasks (65.87 v.s. 64.98), which indicates that the representations preserve more syntactic knowledge.

6 Conclusion

In this paper, we have presented a novel structural position encoding strategy to augment SANs by considering the latent structure of the input sentence. We extract structural absolute and relative positions from the dependency tree and integrate them into SANs. Experimental results on Chinese⇒English and English⇒German translation tasks have demonstrated that the proposed approach consistently improve translation performance over both the absolute and relative sequential position representations.

Future directions include inferring the structure representations from the AMR (Song et al., 2019) or the external SMT knowledge (Wang et al., 2017). Furthermore, the structural position encoding can be also applied to the decoder with RNN Grammars (Dyer et al., 2016; Eriguchi et al., 2017), which we leave for future work.

---

[3]https://github.com/facebookresearch/SentEval/tree/master/data/probing

References

Laura Banarescu, Claire Bonial, Shu Cai, Madalina Georgescu, Kira Griffitt, Ulf Hermjakob, Kevin Knight, Philipp Koehn, Martha Palmer, and Nathan Schneider. 2013. Abstract meaning representation for sembanking. In *the 7th Linguistic Annotation Workshop and Interoperability with Discourse*.

Irwan Bello, Barret Zoph, Ashish Vaswani, Jonathon Shlens, and Quoc V Le. 2019. Attention augmented convolutional networks. *arXiv*.

Thomas G Bever. 1970. The cognitive basis for linguistic structures. *Cognition and the development of language*, 279(362).

Noam Chomsky. 1965. *Aspects of the Theory of Syntax*, volume 11. MIT press.

Michael Collins. 2003. Head-driven statistical models for natural language parsing. *Computational linguistics*, 29(4).

Alexis Conneau, Germán Kruszewski, Guillaume Lample, Loïc Barrault, and Marco Baroni. 2018. What you can cram into a single vector: Probing sentence embeddings for linguistic properties. 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*.

Zi-Yi Dou, Zhaopeng Tu, Xing Wang, Shuming Shi, and Tong Zhang. 2018. Exploiting deep representations for neural machine translation. In *EMNLP*.

Ziyi Dou, Zhaopeng Tu, Xing Wang, Longyue Wang, Shuming Shi, and Tong Zhang. 2019. Dynamic layer aggregation for neural machine translation. In *AAAI*.

Chris Dyer, Adhiguna Kuncoro, Miguel Ballesteros, and Noah A Smith. 2016. Recurrent neural network grammars. In *NAACL*.

Akiko Eriguchi, Yoshimasa Tsuruoka, and Kyunghyun Cho. 2017. Learning to parse and translate improves neural machine translation. In *ACL*.

Jie Hao, Xing Wang, Shuming Shi, Jinfeng Zhang, and Zhaopeng Tu. 2019a. Multi-granularity self-attention for neural machine translation. In *EMNLP*.

Jie Hao, Xing Wang, Shuming Shi, Jinfeng Zhang, and Zhaopeng Tu. 2019b. Towards better modeling hierarchical structure for self-attention with ordered neurons. In *EMNLP*.

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

John Hewitt and Christopher D. Manning. 2019. A structural probe for finding syntax in word representations. In *NAACL*.

Yoon Kim, Carl Denton, Luong Hoang, and Alexander M Rush. 2017. Structured attention networks. In *ICLR*.

Dan Klein and Christopher D Manning. 2003. Accurate unlexicalized parsing. In *ACL*.

Philipp Koehn. 2004. Statistical significance tests for machine translation evaluation. In *EMNLP*.

Sandra Kübler, Ryan McDonald, and Joakim Nivre. 2009. Dependency parsing. *Synthesis Lectures on Human Language Technologies*, 1(1).

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*.

Chunpeng Ma, Akihiro Tamura, Masao Utiyama, Eiichiro Sumita, and Tiejun Zhao. 2019. Improving neural machine translation with neural syntactic distance. In *NAACL)*.

Joakim Nivre. 2005. Dependency grammar and dependency parsing. *MSI report*, 5133(1959).

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *ACL*.

Ankur Parikh, Oscar Täckström, Dipanjan Das, and Jakob Uszkoreit. 2016. A Decomposable Attention Model for Natural Language Inference. In *EMNLP*.

Peter Shaw, Jakob Uszkoreit, and Ashish Vaswani. 2018. Self-Attention with Relative Position Representations. In *NAACL*.

Yikang Shen, Shawn Tan, Alessandro Sordoni, and Aaron Courville. 2019. Ordered neurons: Integrating tree structures into recurrent neural networks. In *ICLR*.

Linfeng Song, Daniel Gildea, Yue Zhang, Zhiguo Wang, and Jinsong Su. 2019. Semantic neural machine translation using AMR. *TACL*.

Emma Strubell, Patrick Verga, Daniel Andor, David Weiss, and Andrew McCallum. 2018. Linguistically-Informed Self-Attention for Semantic Role Labeling. In *EMNLP*.

Sainbayar Sukhbaatar, Edouard Grave, Piotr Bojanowski, and Armand Joulin. 2019. Adaptive attention span in transformers. In *ACL*.

Kai Sheng Tai, Richard Socher, and Christopher D Manning. 2015. Improved semantic representations from tree-structured long short-term memory networks. In *ACL*.

Pasi Tapanainen and Timo Jarvinen. 1997. A non-projective dependency parser. In *Fifth Conference on Applied Natural Language Processing*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is All You Need. In *NIPS*.

Qiang Wang, Fuxue Li, Tong Xiao, Yanyang Li, Yinqiao Li, and Jingbo Zhu. 2018. Multi-layer representation fusion for neural machine translation. In *COLING*.

Xing Wang, Zhengdong Lu, Zhaopeng Tu, Hang Li, Deyi Xiong, and Min Zhang. 2017. Neural machine translation advised by statistical machine translation. In *AAAI*.

Xing Wang, Zhaopeng Tu, Longyue Wang, and Shuming Shi. 2019. Exploiting sentential context for neural machine translation. In *ACL*.

Wei Wu, Houfeng Wang, Tianyu Liu, and Shuming Ma. 2018. Phrase-level self-attention networks for universal sentence encoding. In *ACL*.

Baosong Yang, Jian Li, Derek F. Wong, Lidia S. Chao, Xing Wang, and Zhaopeng Tu. 2019a. Context-aware self-attention networks. In *AAAI*.

Baosong Yang, Zhaopeng Tu, Derek F. Wong, Fandong Meng, Lidia S. Chao, and Tong Zhang. 2018. Modeling localness for self-attention networks. In *EMNLP*.

Baosong Yang, Longyue Wang, Derek F. Wong, Lidia S. Chao, and Zhaopeng Tu. 2019b. Convolutional self-attention networks. In *NAACL*.

Shan Yang, Heng Lu, Shiying Kang, Lei Xie, and Dong Yu. 2019c. Enhancing hybrid self-attention structure with relative-position-aware bias for speech synthesis. In *ICASSP*.

Jiacheng Zhang, Yanzhuo Ding, Shiqi Shen, Yong Cheng, Maosong Sun, Huanbo Luan, and Yang Liu. 2017. THUMT: An Open Source Toolkit for Neural Machine Translation. *arXiv preprint arXiv:1706.06415*.