Fast Directional Self-Attention Mechanism

Tao Shen†, Tianyi Zhou‡, Guodong Long†, Jing Jiang†, Chengqi Zhang†
†Centre for Artificial Intelligence, School of Software, University of Technology Sydney
‡Paul G. Allen School of Computer Science & Engineering, University of Washington
tao.shen@student.uts.edu.au, tianyizh@uw.edu, guodong.long@uts.edu.au
jing.jiang@uts.edu.au, chengqi.zhang@uts.edu.au

Abstract

In this paper, we propose a self-attention mechanism, dubbed "fast directional self-attention (Fast-DiSA)", which is a fast and light extension of "directional self-attention (DiSA)" [15]. The proposed Fast-DiSA performs as expressively as the original DiSA but only uses much less computation time and memory, in which 1) both token2token and source2token dependencies are modeled by a joint compatibility function designed for a hybrid of both dot-product and multi-dim ways; 2) both multi-head and multi-dim attention combined with bi-directional temporal information captured by multiple positional masks are in consideration without heavy time and memory consumption appearing in the DiSA. The experiment results show that the proposed Fast-DiSA can achieve state-of-the-art performance as fast and memory-friendly as CNNs. The code for Fast-DiSA is released at `https://github.com/taoshen58/DiSAN/tree/master/Fast-DiSA`.

1 Introduction

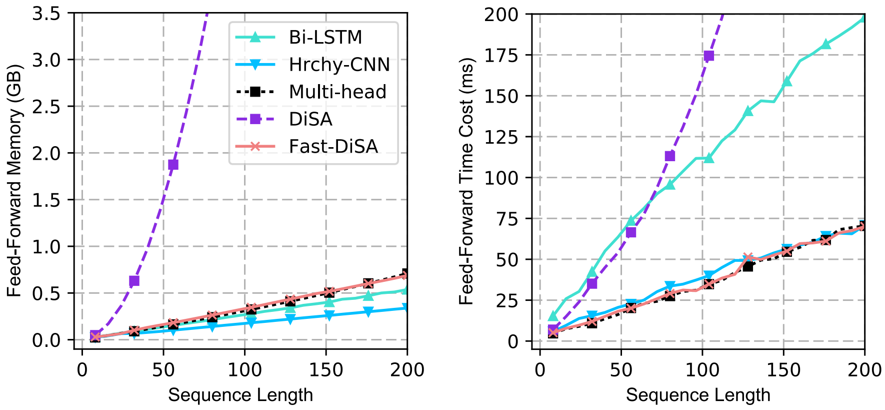
Recently, context fusion module with an attention mechanism for contextual feature modeling has been proved effective in a wide range of NLP tasks, e.g., machine translation [1], reading comprehension [14], etc. More recently, self-attention mechanisms were presented for context fusion with the merits of fewer parameters, more parallelizable computation and better empirical performance [10, 18, 15]. Then, neural nets based solely on self-attention mechanisms were proposed, which have been applied to machine translation [18] and sentence embedding [15] tasks.

Directional self-attention mechanism [15], a multi-dim token2token self-attention, has shown its better effectiveness on a wide range of tasks, however suffering from heavy memory and time consumption due to a 4D tensor (with the shape of [*batch size*, *sequence length*, *sequence length*, *feature channels*]) needed to be allocated for the alignment score and weighted sum in the attention.

In this paper, we propose an novel and light attention mechanism for context fusion, called "fast directional self-attention (Fast-DiSA)", in which, 1) the dependency calculated between tokens is in consideration of both token2token and source2token dependencies; 2) both multi-head (i.e., multi-subspace) and multi-dimensional (i.e., feature-wise) alignment scores are used without heavy computation time or memory load; and 3) multiple positional masks are integrated for bi-directional information embedding. Besides, the proposed model can be easily optimized under any parallel computing framework, which makes it possible to be used for long sequences and large-scale datasets.

In the experiments, we implement the proposed model on the sentence-encoding based natural language inference. The results demonstrate that the proposed model can achieve state-of-the-art with light time and memory consumption. To testify time-/memory-efficiency of the proposed model, we visualize the memory consumption and time cost *vs.* sequence length in Figure 1(a) and 1(b) respectively on random sequence data.

Preprint. Work in progress.



(a) (b)

Figure 1: (a) Memory consumption and (b) Time cost *vs.* sequence length on synthetic data with batch size of 64 and feature channels of 300. Baselines are detailed in Section 4.

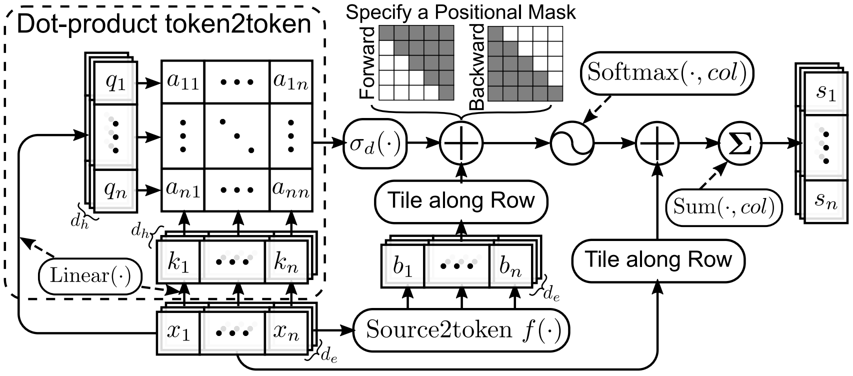


Figure 2: Fast-DiSA in single head.

2 Background

The background of this work is elaborated in the original paper of DiSA by Shen et al. [15]: 1) vanilla attention mechanism and multi-dimensional (multi-dim) mechanism are introduced in Section 2.2 and 3.1 respectively; 2) self-attention mechanism is introduced in Section 2.3; 3) token2token and source2token self-attention mechanisms are detailed in Section 3.2; 4) masked (directional) self-attention mechanism is demonstrated in Section 3.3.

3 Proposed Models

In this section, a mechanism named as "fast directional self-attention (Fast-DiSA)" based on DiSA is presented in Section 3.1 to associate scaled dot-product token2token self-attention with multi-dim source2token self-attention for two types of dependencies and then integrate the directional information. Next, an optimized Fast-DiSA algorithm is presented for the "Fast" and memory-efficiency.

3.1 Fast Directional Self-Attention (Fast-DiSA)

In contrast to the existing self-attention mechanisms that capture only one types of dependency, the proposed fast directional self-attention (Fast-DiSA) can 1) model both kinds of token2token and source2token dependencies by its a novel compatibility function, 2) the attention alignment scores is computed by a hybrid of both dot-product and multi-dim compatibility functions to measure a combined dependencies modeled in two mutually-beneficial approaches, and 3) model the diversely sequential information by multi-head technique [18] with multiple positional masks. An illustration of "single-head" Fast-DiSA are shown in Figure 2.

First, the scaled product attention mechanism [18] is employed to model syntactic dependency between each token pair due to the less computation resources required by the dot-product operations, in which the compatibility function is written as

$$f^t(x_i, x_j) = \langle W^{(t1)} x_i, W^{(t2)} x_j \rangle / \sqrt{d_h}, \tag{1}$$

where $W^{(t1)}, W^{(t2)} \in \mathbb{R}^{d_h \times d_e}$ are the learnable weights and $\langle \cdot, \cdot \rangle$ denotes the inner-product operation. Then, the multi-dim source2token self-attention is used to estimate the feature-wise dependency and contribution of each token to a specific task because applying multi-dim compatibility function to a source2token self-attention makes alignment scores at feature level and thus essentially improve the performance, however, without significant computation time and memory increase. The compatibility function is written as

$$f^s(x_i) = W^{(s2)} \sigma_s(W^{(s1)} x_i + b^{(s1)}) + b^{(s2)}, \tag{2}$$

where $W^{(s1)} \in \mathbb{R}^{d_h \times d_e}, W^{(s2)} \in \mathbb{R}^{d_e \times d_h}$ are the learnable weights and $\sigma_s(\cdot)$ is an activation function. Based on the above, the compatibility function in Fast-DiSA is broadcastly adding the scalar token2token self-attention alignment scores $f^t(x_i, x_j)$ to the feature-wise source2token self-attention alignment scores $f^s(x_i)$ for modeling these two kinds of dependencies. In addition, the positional masks defined in Shen et al. [15] can also be introduced to Fast-DiSA for bi-directional order information embedding. Note that, the attention of a token to itself should be enabled here for context-aware representations, otherwise a semantic fusion gate used in Shen et al. [15] will be used to position-wisely integrate original input with self-attention output [15].[1] Therefore, the compatibility function can be formulated by

$$f^{fd}(x_i, x_j) = \sigma_t[f^t(x_i, x_j)] + f^s(x_i) + M_{ij}, \tag{3}$$

where $\sigma_t(\cdot)$ is an activation function. Then, a feature-wise $\mathrm{softmax}$ function is applied to the calculated alignment scores $[f^{fd}(x_i, x_j)]_{i=1}^n$ for each head/governor token x_j and corresponding all dependent tokens $[x_i]_{i=1}^n$, which results in a categorical distribution at feature level. The attention probabilities of the governor token x_j to x_i for all d_e feature channels are represented as

$$\{p(z_k = i | \boldsymbol{x}, x_j)\}_{k=1}^{d_e} \triangleq p_i^j = e^{f^{fd}(x_i, x_j)} / \sum\nolimits_{l=1}^{n} e^{f^{fd}(x_l, x_j)}, \text{ for } j = 1, \ldots, n. \tag{4}$$

Then, output of single-head Fast-DiSA mechanism is calculated by using an expectation of dependent tokens under the obtained categorical distribution with dual-dependency modeled for each governor token, i.e.,

$$\boldsymbol{s} = \{s_j\}_{j=1}^n = \{[\mathbb{E}_{i \sim p(z_k | \boldsymbol{x}, x_j)}([x_i]_k)]_{k=1}^{d_e}\}_{j=1}^n = \{\sum\nolimits_{i=1}^{n} p_i^j \cdot x_i\}_{j=1}^n. \tag{5}$$

Next, the multi-head [18] with multiple positional masks (e.g., forward, backward and multi-length window) techniques are applied to the single-head Fast-DiSA for modeling distinctly temporal information in the meantime.

For the simplicity in formula, we first denote the entire single-head Fast-DiSA mechanism as a function, $\text{Single-Fast-DiSA}(\boldsymbol{x}, M)$, where the two arguments are input sequence $\boldsymbol{x} \in \mathbb{R}^{d \times n}$ and positional mask $M \in \mathbb{R}^{n \times n}$, respectively. Then, the multi-head Fast-DiSA (simply demoted as Fast-DiSA) can be readily formulated by

$$\boldsymbol{s} = [H_1; \ldots; H_h], \text{ where } H_l = \text{Single-Fast-DiSA}(W_l^{(m)} \boldsymbol{x}, M^{(l)}), \tag{6}$$

where h stands for the number of heads, $W_l^{(m)} \in \mathbb{R}^{d_h \times d_e}$ is used to project the sequence of input representations to l-th subspace with d_h hidden units, $M^{(l)}$ represents a positional mask used in l-th subspace, $[\cdot; \ldots; \cdot]$ denotes a vertical concatenation operation, and $\boldsymbol{s} \in \mathbb{R}^{h \cdot d_h \times n}$ is the output.

In practice, we only use forward and backward positional mask to embed bi-directionally temporal information, therefore applying the forward positional mask to half of heads and the backward positional mask to the remaining.

[1] Empirically, we found that if single layer of token2token self-attention with an source2token self-attention is used to encode sentence, the semantic fusion gate is important for better performance; however, if a deep stacked self-attention mechanism like the model used by Vaswani et al. [18] is desired, the fusion gate is redundant, and, the attention of token to itself can also be disabled due to the residual connection [9].

3.2 Optimized Fast-DiSA Mechanism

The proposed Fast-DiSA can be optimized for less time consumption and lower memory requirement by fully exploiting the parallelism of matrix multiplication. The optimized Fast-DiSA mechanism is detailed in Algorithm 1.

Algorithm 1 Optimized Fast-DiSA Mechanism

Input: a sequence of representations $\boldsymbol{x} \in \mathbb{R}^{d_e \times n}$, head number h, hidden unit number d_h, positional masks $\{M^l\}_{l=1}^h$ and weights: $\{W_l^{(m)} \in \mathbb{R}^{d_h \times d_e}\}_{l=1}^h, \{W_l^{(t1)}, W_l^{(t2)}, W_l^{(s1)}, W_l^{(s2)} \in \mathbb{R}^{d_h \times d_h}\}_{l=1}^h$
Output: a sequence of context-aware representations $\boldsymbol{s} = [s_1, \dots, s_n] \in \mathbb{R}^{h \cdot d_h \times n}$

```
procedure FAST DIRECTIONAL SELF-ATTENTION (FAST-DISA)
    for all $l = 1, \dots, h$ do    ▷ Fulfilling attention procedure in each subspace in parallel
        $\boldsymbol{x^l} = W_l^{(m)} \boldsymbol{x}$    ▷ Transforming input representations into $l$-th subspace
        $R_l \leftarrow (W_l^{(t1)} \boldsymbol{x^l})^T (W_l^{(t2)} \boldsymbol{x^l})$    ▷ Scaled dot-product token2token alignment scores
        $S_l \leftarrow W_l^{(s2)} \sigma_s(W_l^{(s1)} \boldsymbol{x^l} + b_l^{(s1)}) + b_l^{(s2)}$    ▷ Multi-dim source2token alignment scores
        $E_l^R \leftarrow \exp(\sigma_t(R_l)) \cdot \exp(M_l)$ ▷ Applying exp and positional mask to alignment scores
        $E_l^S \leftarrow \exp(S_l); E_l^X \leftarrow \boldsymbol{x^l} \cdot E_l^S$ ▷ Applying original input $\boldsymbol{x}$ and exp to alignment scores
        $H_l \leftarrow E_l^R E_l^X / E_l^R E_l^S$    ▷ Performing weighted sum to produce output in $l$-th subspace
    end for
    Return $\boldsymbol{s} \leftarrow [H_i; \dots; H_h]$    ▷ Vertically concatenating all the outputs from subspaces
end procedure
```

Memory-Efficiency: Compared to multi-dim token2token self-attention [15] that inherently requires use 4-D tensors (i.e., [batch size, seq len, seq len, channels]) to store multi-dim alignment scores during training phase, the optimized Fast-DiSA, despite multi-dim alignment scores associated, does not use any high-rank tensor but only matrix multiplication to avoid memory explosion problem.

Time-Efficiency: Under any parallel computing platform or deep learning framework, the computations for optimized Fast-DiSA mechanism, mainly consisting of multi-head matrix operation, can be highly parallelized among all heads. In contrast to efficient multi-head attention [18], the Fast-DiSA merely adds two fully connected layer and a element-wise matrix production operation.

4 Experiments

We only conduct the experiments of the proposed Fast-DiSA on natural language inference task to simply demonstrate the performance, time and memory efficiency of proposed model. However, the Fast-DiSA is a very unified, effective and efficient context fusion module that can be applied to most NLP tasks as an alternative to the RNNs and CNNs. The further application of Fast-DiSA will be explored in the future. The experimental codes are implemented using Python with Tensorflow deep learning framework and run on a single NVIDIA GTX 1080Ti graphic card. Besides, data of both time and memory consumption are collected under Tensorflow-1.7 with CUDA9 and cuDNN7.

The baseline context fusion methods for comparison include: **1) Bi-LSTM**[8]: 600D bi-directional LSTM consisting of 300D forward plus 300D backward LSTMs; **2) Hrchy-CNN**[7]: 3-layer 300D stacking CNN with kernel length of 5, to which gated linear units [6] and residual connection [9] is applied; **3) Multi-head**[18]: 600D multi-head self-attention with 8 heads and 75 hidden units per heads, to which the positional encoding used in [18] is applied for temporal order information; **4) DiSA**[15]: 600D directional self-attention mechanism consisting of 300D forward and 300D backward masked self-attentions;

Natural Language Inference aims at speculating the semantic relationship between a premise sentence and a corresponding hypothesis, where the relationship could be *entailment*, *neutral* or *contradiction*. In the experiment, we compare Fast-DiSA to other baselines on the Stanford Natural Language Inference [2] (SNLI) dataset, which contains standard training/dev/test split of 549,367/9,842/9,824 samples.

Following the method of applying sentence-encoding to NLI given in [3], two parameter-tied sentence-encoding models are applied to the premise and the hypothesis sentences, generating the premise and hypothesis representations, s^p and s^h respectively. The relationship represented by the concatenation of s^p, s^h, $s^p - s^h$ and $s^p \odot s^h$ is passed into a 3-way neural classifier for relationship prediction.

Table 1: Experimental results for different methods on SNLI. $|\boldsymbol{\theta}|$: the number of parameters (excluding word embedding part). **Time/Epoch**: averaged training time per epoch with batch size 128. **Inf. Time**: averaged inference time with batch size 128. **Memory**: memory load on synthetic data of sequence length 64 and batch size 64 with back-propagation considered. **Train Acc.** and **Test Acc.**: the accuracies on training and test sets.

Model	$\|\boldsymbol{\theta}\|$	**Time/Epoch**	**Inf. Time**	**Memory**	**Train Acc.**	**Test Acc.**
300D SPINN-PI encoders [3]	3.7m				89.2	83.2
600D Bi-LSTM encoders [12]	2.0m				86.4	83.3
600D Bi-LSTM enc.+intra-attn [12]	2.8m				84.5	84.2
600D Deep Gated Attn. [4]	11.6m				90.5	85.5
600D Gumbel TreeLSTM enc. [5]	10.0m				93.1	86.0
600D Residual stacked enc. [13]	29.0m				91.0	86.0
300D Reinforced SAN [16]	3.1m				92.6	86.3
Distance-based SAN [11]	4.7m	-s			89.6	86.3
Bi-LSTM [8]	2.9m	854s	9.1s	942MB	90.4	85.0
Hrchy-CNN [7]	3.4m	195s	1.8s	309MB	91.3	83.9
Multi-head [18]	2.0m	179s	1.5s	466MB	89.6	84.2
DiSA [15]	2.3m	390s	5.2s	6682MB	91.1	85.6
Fast-DiSA	2.9m	180s	1.6s	558MB	91.8	**86.3**

The experiment results, in terms of performance, time efficiency and memory load, for models from the official leaderboard, baselines and this paper are shown in Table 1. The proposed model, Fast-DiSA, achieves state-of-the-art performance by less cost in both time and memory. Compared to methods from the leaderboard, Fast-DiSA outperforms RNN-based encoders (e.g., Bi-LSTM encoders and Residual stacked enc.), RNN+Attention encoders (e.g., Bi-LSTM enc.+intra-attn and Deep Gated Attn.) and even semantic tree based encoders (e.g., SPINN-PI encoders and Gumbel TreeLSTM enc.) by a large margin. Compared to the two competitive self-attention networks, where Reinforced SAN uses a hybrid of soft and hard attention mechanism to model the dependency leading to a difficulty in training and Distance-based SAN[2] use bi-directional and distance-aware multi-head self-attention plus gate and layer normalization leading to more parameters and lower time efficiency, the proposed Fast-DiSA, a model in end-to-end supervised fashion, can deliver same state-of-the-art performance by using much fewer parameters and less computation time.

Table 2: An ablation study of Fast-DiSA on SNLI.

Model	$\|\boldsymbol{\theta}\|$	**Inf. Time**	**Memory**	**Test Accuracy**
Fast-DiSA	2.9m	1.6	558MB	86.3
Fast-DiSA w/o forward and backward position mask	2.9m	1.6	559MB	85.3 (-1.0)
Fast-DiSA w/o dot-product token2token self-attention	2.5m	1.5	430MB	85.8 (-0.5)
Fast-DiSA w/o multi-dim source2token self-attention	2.5m	1.4	433MB	84.9 (-1.4)
Fast-DiSA w/o all proposed modules	1.8m	1.1	320MB	84.3 (-2.0)

In contrast to the models from baselines, Fast-DiSA substantially outperforms LSTM with $4 \sim 5\times$ time efficiency and also achieves much better effectiveness than hrchy-CNN by using comparable parameter number and computation time. Compared to the dot-product self-attention (Multi-Head), Fast-DiSA uses similar computation time and memory to perform multi-dim self-attention for better performance. And, compared to the multi-dim self-attention (DiSA), Fast-DiSA uses much less memory and time but performs better prediction quality.

We also conduct an ablation study of Fast-DiSA to verify the capability of each part in context fusion. The results shown in Table 2 demonstrate that all token2token dependency, source2token dependency and directional masks contribute important information to sequence modeling.

[2] Distance-based SAN is even less time efficient than DiSA as shown in [11]

5 Conclusion

In this paper, we proposed a time- and memory-efficient self-attention mechanism, called fast directional self-attention (Fast-DiSA), for context fusion. The proposed Fast-DiSA merges the global and data-driven dependencies by a novel hybrid compatibility function in both dot-product and additive ways. Then, the attention alignment scores is in consideration of both multi-head and multi-dim attentions without any memory or time explosion problem. Additionally, different positional masks are simultaneously applied to the subspaces for bi-directional information modeling. In the experiments, the proposed Fast-DiSA shows not only its capability in sequence modeling but is as efficient as CNNs.

In the future, there are several interesting works can be done: 1) applying more kinds of positional masks [15] or distance-ware masks [11] to different subspace, 2) applying the Fast-DiSA as a context fusion model to more NLP tasks for a better state-of-the-art performance, and 3) integrating the Fast-DiSA with a hierarchical self-attention technique [17] for fast and efficient passage- or document-level context fusion.

References

[1] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. In *ICLR*, 2015.

[2] Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. A large annotated corpus for learning natural language inference. In *EMNLP*, 2015.

[3] Samuel R Bowman, Jon Gauthier, Abhinav Rastogi, Raghav Gupta, Christopher D Manning, and Christopher Potts. A fast unified model for parsing and sentence understanding. In *ACL*, 2016.

[4] Qian Chen, Xiaodan Zhu, Zhen-Hua Ling, Si Wei, Hui Jiang, and Diana Inkpen. Recurrent neural network-based sentence encoder with gated attention for natural language inference. In *RepEval@ EMNLP*, 2017.

[5] Jihun Choi, Kang Min Yoo, and Sang goo Lee. Learning to compose task-specific tree structures. In *AAAI*, 2018.

[6] Yann N Dauphin, Angela Fan, Michael Auli, and David Grangier. Language modeling with gated convolutional networks. *arXiv preprint arXiv:1612.08083*, 2016.

[7] Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N Dauphin. Convolutional sequence to sequence learning. *arXiv preprint arXiv:1705.03122*, 2017.

[8] Alex Graves, Navdeep Jaitly, and Abdel-rahman Mohamed. Hybrid speech recognition with deep bidirectional lstm. In *Automatic Speech Recognition and Understanding (ASRU), 2013 IEEE Workshop on*, pages 273–278. IEEE, 2013.

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[10] Minghao Hu, Yuxing Peng, and Xipeng Qiu. Reinforced mnemonic reader for machine comprehension. *CoRR, abs/1705.02798*, 2017.

[11] Jinbae Im and Sungzoon Cho. Distance-based self-attention network for natural language inference. *arXiv preprint arXiv:1712.02047*, 2017.

[12] Yang Liu, Chengjie Sun, Lei Lin, and Xiaolong Wang. Learning natural language inference using bidirectional lstm model and inner-attention. *arXiv preprint arXiv:1605.09090*, 2016.

[13] Yixin Nie and Mohit Bansal. Shortcut-stacked sentence encoders for multi-domain inference. *arXiv preprint arXiv:1708.02312*, 2017.

[14] Minjoon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. Bidirectional attention flow for machine comprehension. In *ICLR*, 2017.

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

[16] Tao Shen, Tianyi Zhou, Guodong Long, Jing Jiang, Sen Wang, and Chengqi Zhang. Reinforced self-attention network: a hybrid of hard and soft attention for sequence modeling. *arXiv preprint arXiv:1801.10296*, 2018.

[17] Tao Shen, Tianyi Zhou, Guodong Long, Jing Jiang, and Chengqi Zhang. Bi-directional block self-attention for fast and memory-efficient sequence modeling. In *ICLR*, 2018.

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