MUSE: MULTI-SCALE ATTENTION MODEL FOR SEQUENCE TO SEQUENCE LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

In this work, we rethink a fundamental question: "Is attention really all you need?". Although attention have achieved promising results on a variety of natural language processing tasks, we find that attention is still weak in long sentence modeling. The global attention map is too dispersed to capture valuable information. In such case, the local/token features that are also significant to sequence modeling are omitted to some extent. To address this problem, we propose a **MU**lti-**S**cale att**E**ntion model (**MUSE**) by concatenating attention networks with convolutional networks and position-wise feed-forward networks to explicitly capture local and token features. Experimental results show that the proposed model achieves substantial performance improvements over Transformer, especially on long sentences, and pushes the state-of-the-art from 35.6 to 36.3 on IWSLT 2014 German to English translation task, from 30.6 to 31.3 on IWSLT 2015 English to Vietnamese translation task. We also reach the state-of-art performance on WMT 2014 English to French translation dataset, with a BLEU score of 43.2.

1 INTRODUCTION

In recent years, Transformer has been widely used due to its promising performance on a variety of natural language processing tasks, like machine translation (Vaswani et al., 2017; Dehghani et al., 2018), text classification (Devlin et al., 2018; Yang et al., 2019), language modeling (Sukhbaatar et al., 2019; Dai et al., 2019; Child et al., 2019), etc. It is solely based on an attention mechanism that captures global dependencies between input tokens, dispensing with recurrence and convolutions entirely. The key idea of the attention mechanism is updating token representations based on a weighted sum of all input representations.

Despite significant results, the ability of attention to model long sentence has come into question (Tang et al., 2018). As shown in Figure 1 (a), the performance of Transformer drops largely with the increase of the source sentence length. Based on the empirical analysis, we find that the performance drop is mainly due to the dispersed attention map where local/token features, also significant to sequence modeling, are omitted to some extent, as shown in Figure 1 (b). Although the original Transformer encodes position information into token embeddings to address this problem, it is still unknown how much distance information is kept in the token representation with the increase of layer depth.

To address this problem, we introduce a multi-scale attention model called MUSE, which concatenates an attention network with a convolutional network and a position-wise feed-forward network at each layer to explicitly encode local and token features. A position-wise feed-forward network allows a token-level transformation, which keeps token-specific features well. A convolutional network is responsible for capturing local features. The dynamic convolution network (Wu et al., 2019) is adopted as an implementation for fast computation speed. A convolutional neural network consists of multiple convolution cells with different kernel sizes. As shown in Figure 2, the left figure shows the structure of the original Transformer. The middle shows MUSE (MUSE_FF) concatenating attention networks and feed-forward networks and the right shows the standard MUSE concatenating attention networks, feed-forward networks, and dynamic convolution networks.

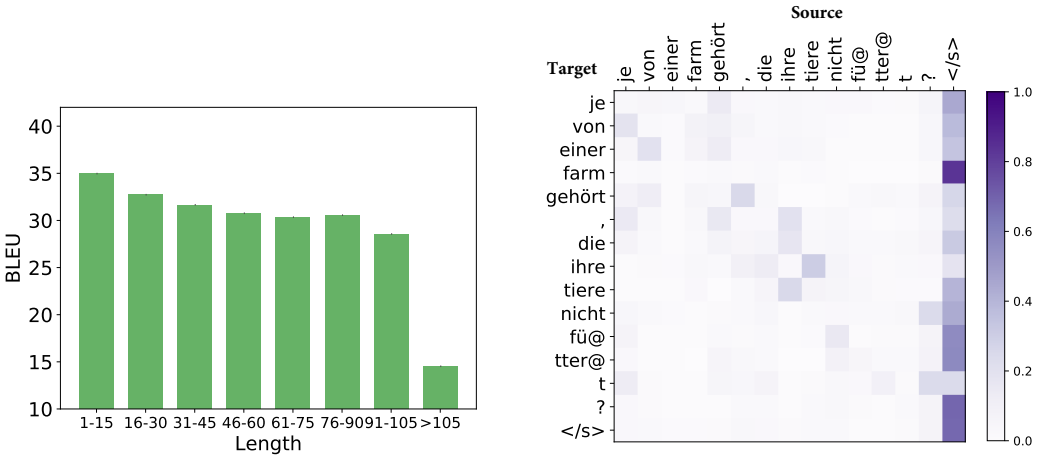

Figure 1: The left figure shows that the performance drops largely with the increase of sentence length on the De-En dataset. The right figure shows an attention map from the 3-th encoder layer. As we can see, the attention map is too dispersed to capture sufficient information. For example, "[EOS]", contributing little to word alignment, is surprisingly over attended.

We test these three models on various natural language processing tasks, including machine translation and language modeling. In particular, MUSE_FF yields large improvements over Transformer with 0.5 BLEU on IWSLT 2014 German to English translation dataset. The only difference between MUSE_FF and the original Transformer is the position of the feed-forward network. In the original Transformer, the feed-forward network is put after attention operations. Since global features have been encoded into token representation in the attention mechanism, it is hard for the feed-forward network to keep sufficient token features. In MUSE_FF, the feed-forward network is directly linked with the input representation and the token-features are kept well. The standard MUSE, fusing attention, feed-forward networks, and convolutional networks, achieves the best results and pushes the state-of-the-art to 36.3 on IWSLT 2014 German to English translation dataset, 31.3 on IWSLT 2015 English to Vietnamese translation dataset. We also reach the state-of-the-art performance on WMT English to French translation dataset, with a bleu score of 43.2. These results show that attention is not all you need and local/token features also matter.

The main contributions are summarized as follows:

- We find that attention suffers from dispersed weights. To address this problem, we propose a multi-scale attention model MUSE for sequence to sequence learning.
- It is interesting to see that the simple version of MUSE, fusing attention and a feed-forward network in a parallel way, achieves better results than the serial way in the original Transformer. It shows the importance of token features.
- On the simple version basis, we further introduce convolution networks into MUSE to capture local features at different granularity, which brings larger improvements over Transformer.
- MUSE achieves state-of-the-art BLEU scores on three machine translation tasks, IWSLT 2014 German to English translation, IWSLT 2015 English to Vietnamese translation, and WMT 2014 English to French translation.

2 MUSE: Multi-Scale Attention Model

Like other sequence-to-sequence models, MUSE also adopts an encoder-decoder framework. The encoder takes a sequence of word embeddings $(x_1, \cdots, x_n)$ as input where n is the length of input. It transfers word embeddings to a sequence of hidden representation $\boldsymbol{z} = (z_1, \cdots, z_n)$. Given $\boldsymbol{z}$, the decoder is responsible for generating a sequence of text $(y_1, \cdots, y_m)$ token by token.

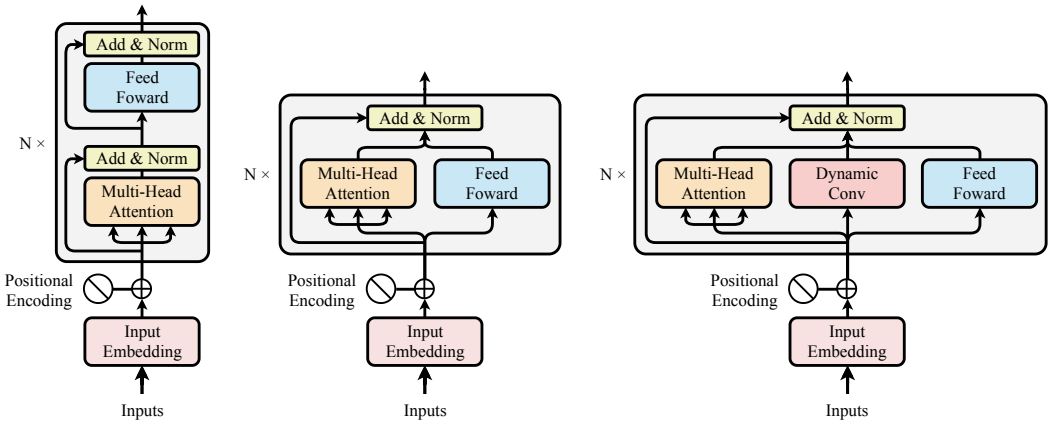

Figure 2: The left figure shows the original Transformer model. The middle is the simple version MUSE (MUSE_FF) only concatenating an attention network and a position-wise feed-forward network. The right is the standard MUSE concatenating an attention network, a feed-forward network and a dynamic convolution network together.

The encoder is a stack of N layers. Each layer consists of a MUSE module. Residual mechanism and layer normalization are used to connect two adjacent layers.

The decoder is also a stack of N layers. Each layer contains two sub-layers: a MUSE module and a context attention module. The MUSE module is responsible for capturing features from the generated text representations. The context-attention performs attention over the output of the encoder stack. Residual mechanism and layer normalization are also used to connect two modules and two adjacent layers.

The key part in the proposed model is the MUSE module, which contains three main parts: self-attention for capturing global features, dynamic convolution for capturing local features, and a position-wise feed-forward network for capturing token features. The module takes the output of $(i-1)$ layer as input and generates the output representation in a fusion way:

$$X_i = MultiHead(X_{i-1}) + Conv(X_{i-1}) + FFN(X_{i-1}) \tag{1}$$

where "MultiHead" refers to self-attention, "Conv" refers to dynamic convolution, "FFN" refers to a position-wise feed-forward network. The followings list the details of each part.

2.1 Self-Attention for Global Context Representation

Self-attention is responsible for capturing global attention. For a given input sequence X, it first projects X into three representations, key K, query Q, and value V. Then, it uses a multi-head attention mechanism to get the output representation:

$$\begin{aligned} MultiHead(X) &= Concat(head_1, \cdots, head_m)W^O \\ \text{where } head_i &= Attention(QW_i^Q, KW_i^K, VW_i^V) \\ Q, K, V &= Linear_1(X),\ Linear_2(X),\ Linear_3(X) \end{aligned} \tag{2}$$

Where W^O, W_i^Q, W_i^K, and W_i^V are projection parameters. m is the number of head. The attention operation is the dot-production between key, query, and value pairs:

$$Attention(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V \tag{3}$$

2.2 Dynamic Convolution for Local Context Modeling

We introduce convolution operations into MUSE to capture local context. In order to save parameter size, we use dynamic convolution networks as implementation. Each convolution sub-module contains multiple cells with different kernel sizes . They are used for capturing different-range features.

The output of the convolution cell with kernel size k is:

$$Conv_k(X) = dynmaic\_conv_k(XW^{in})W^{out} \tag{4}$$

where W^{in} and W^{out} are projection parameters. The $dynamic\_conv$ refers to dynamic convolution in the work of Wu et al. (2019). For an input sequence X, the output O is computed as:

$$O_{i,c} = dynamic\_conv_k(X) = \sum_{j=1}^{k} \big(\text{softmax}(\sum_{c=1}^{d} W^Q_{j,c} X_{i,c}) \cdot X_{i+j-\lceil \frac{k+1}{2} \rceil, c}\big) \tag{5}$$

where d is the hidden size.

Weight Tying To decrease the model memory usage, we share the weight metrics of W^{in} in dynamic convolution (Eq. 3) and the concatenation of W^V_i in self-attention (Eq. 1).

$$Conv_k(X) = dynmaic\_conv_k(XW^V)W^{out} \tag{6}$$

where W^V is the concatenation of W^V_i in self attention.

Dynamically Selected Convolution Kernels We introduce a gating mechanism to automatically select the weight of different convolution cells.

$$Conv(X) = \sum_{i=1}^{n} \frac{\exp(\alpha_i)}{\sum_{j=1}^{n} \exp(\alpha_j)} Conv_{k_i}(X) \tag{7}$$

where α_i is a scalar initialized with $1/n$. n is the number of cells.

2.3 Feed-forward Network for Capturing Token Representations

To capture token features, MUSE concatenates an attention network with a position-wise feed-forward network at each layer. Since the linear transformations are the same across different positions, the position-wise feed-forward network can be seen as a token feature extractor.

$$FFN(x) = (0, H_l W_1 + b_1)W_2 + b_2 \tag{8}$$

where W_1, b_1, W_2, and b_2 are projection parameters.

3 Experiment

We evaluate MUSE on three machine translation tasks. This section describes the used datasets, experimental settings, detailed results, and analysis.

3.1 Evaluation Datasets

WMT14 EN-FR datasets. The WMT 2014 English-French translation dataset, consisting of $36M$ sentence pairs, is adopted as a benchmark dataset. We use the standard split of development set and test set. We use *newstest2014* as the test set and use *newstest2012 +newstest2013* as the development set. We also adopt a joint source and target BPE factorization with the vocabulary size of $40K$.

IWSLT DE-EN and EN-VI datasets. Besides, two small IWSLT datasets are also adopted. The IWSLT 2014 German-English translation dataset consists of $160k$ sentence pairs. We also adopt a joint source and target BPE factorization with the vocabulary size of $32K$. The IWSLT 2015 English-Vietnamese translation dataset consists of $133K$ training sentence pairs.

For all three datasets, We adopt the default split of development set and test set respectively. Following the fairseq reposity (Ott et al., 2019), we use the BPE technique to preprocess the DE-EN and EN-FR datasets and remove BPE before the evaluation. For the EN-VI task, the vocabulary size for English is $17.2K$, and the vocabulary size for the Vietnamese is $6.8K$.

Model	Parameter Size	EN-VI	DE-EN
NBMT (Huang et al., 2017)	-	28.1	30.1
SACT (Lin et al., 2018)	-	29.1	-
NP2MT (Feng et al., 2018)	-	30.6	31.7
Fixup (Zhang et al., 2019b)	44M	-	34.5
DynamicConv (Wu et al., 2019)	39M	-	35.2
Macaron (Lu et al., 2019)	43M	-	35.4
MAtt (Zhang et al., 2019a)	92M	-	35.6
MUSE	49M	**31.3**	**36.3**

Table 1: BLEU scores of MUSE and state-of-the-art approaches on IWSLT DE-EN, IWSLT EN-VI translation datasets.

Model	Parameter Size	EN-FR
CNNSeq2seq (Gehring et al., 2017)	216M	40.5
Transformer (Vaswani et al., 2017)	213M	41.0
RNMT+ (Chen et al., 2018)	379M	41.0
Weighted Transformer (Ahmed et al., 2017)	213M	41.4
Relative Transformer (Shaw et al., 2018)	-	41.5
ScalingNMT (Ott et al., 2018)	210M	**43.2**
DynamicConv (Wu et al., 2019)	213M	**43.2**
MUSE	233M	**43.2**

Table 2: BLEU scores of models on WMT EN-FR translation dataset.

3.2 Experimental Settings

We build a model consisting of 12 encoder layers and 12 decoder layers. The hidden dimension is set to 384 on DE-EN and EN-VI translation, and 768 on EN-FR translation. The embedding layer is initialized by a normal distribution. For the rest parameters, we use the default initialization method by *pytorch*. Suppose d_{in} is the input dimension, the parameters of the fully connected layer are initialized by an uniform distribution $(-1/\sqrt{d_{in}}, 1/\sqrt{d_{in}})$.

We calculate the batch size at a token level, which is so-called dynamic batching (Vaswani et al., 2017). For the DE-EN dataset, we train the model for $20K$ steps with a batch size of $4K$. The parameters are updated every 4 steps. The dropout rate is set to 0.4. For the EN-VI dataset, We train the model for $10K$ steps with a batch size of $4K$. The parameters are updated every 4 steps. The dropout rate is set to 0.3. For EN-FR translation, we train the model for $25K$ updates with a batch size of $3,584$, following Ott et al. (2018). The parameters are updated every 32 steps. The dropout rate is set to 0.3. The models for EN-FR are trained on 4 Titan RTX GPUs while the models for EN-VI and DE-EN are trained on a single NVIDIA RTX 2080Ti GPU.

We tune the hyper-parameter on the valid set. We use Adam optimizer with a learning rate of 0.001. Following Vaswani et al. (2017), we use a learning rate warmup mechanism and invert the learning rate decay with warmup updates of $4K$. We adopt an early stopping mechanism in the training. To be specific, we stop training 10 epochs after the epoch with the lowest valid loss and then average the last 10 checkpoints for inference. During inference, we adopt beam search with a beam size of 5 for DE-EN and EN-VI translation, and a beam size of 4 for EN-FR translation. The length penalty is set to 0.8 for all EN-FR and it is set to 1 for other datasets. The BLEU[1] metric is adopted to evaluate the model performance on the machine translation datasets during evaluation.

3.3 Results

As shown in Table 1, MUSE outperforms the previously reported models and establishes new state-of-the-art results on the IWSLT DE-EN and EN-VI machine translation tasks. To be specific, MUSE

[1]https://github.com/moses-smt/mosesdecoder/blob/master/ scripts/generic/multi-bleu.perl

Model	Parameter Size	BLEU
Transformer (6L, dim = 512)	43M	35.3
Transformer (12L, dim = 512)	74M	*diverge*
Transformer (12L, dim = 384)	44M	*diverge*
DynamicConv (6L, dim = 512)	39M	35.7
DynamicConv (12L, dim = 512)	63M	35.6
DynamicConv (12L, dim = 384)	38M	35.6
MUSE (6L, dim = 512)	47M	35.3
MUSE (12L, dim = 512)	82M	36.0
MUSE (12L, dim = 384)	49M	**36.3**

Table 3: Comparisons between baselines and MUSE on the IWSLT 2015 DE-EN translation task. The gradients of Transformer and DynamicConv diverge under deep structures while MUSE yet gets better results with the increase of layer depth. To avoid the effect of parameter size on model performance, we also evaluate MUSE with the similar model size as Transformer (6L, dim = 512). The small version of MUSE still beats Transformer by a large margin.

achieves a 36.3 BLEU score on DE-EN translation and a 31.3 BLEU score on EN-VI translation. Furthermore, compared with the approaches with the similar model size, Fixup and Macaron, MUSE achieves almost 1.0 BLEU score improvement on DE-EN datast. On the WMT EN-FR machine translation dataset, MUSE also achieves state-of-the-art results, with a BLEU score of 43.2, as shown in Table 2.

Table 3 shows the performance of MUSE and baselines under different layers and dimensions. With the increase of layer depth, the gradient of Transformer and DynamicConv diverges. By contrast, the performance of MUSE even gets slightly improvement under deeper structures. These results prove the stability of MUSE which can be applied on more complex tasks in the future.

Furthermore, it is interesting to see that DynamicConv achieves better results than Transformer (Table 3). DynamicConv mainly focuses on capturing local features while Transformer is responsible for capturing global features via attention. These results indicate that local features may be more important than global features in sequence to sequence learning because of their higher flexibility. Second, the proposed model, MUSE (12 layers), augmenting self-attention with local and token features, beats Transformer and DynamicConv by a large margin, with a 0.5 BLEU improvement on DE-EN translation. To avoid the effect of parameter size on model performance, we also evaluate MUSE with the similar model size as Transformer (6L, dim = 512). The small version of MUSE still beats Transformer by a large margin.

3.4 Ablation Analysis

To explore and understand the idea of combining multi-scale information, we also conduct a series of ablation studies by taking results on IWSLT 2015 DE-EN translation as an example.

Model	Parameter Size	BLEU
Transformer	43M	35.3
MUSE_FF	44M	35.8
MUSE	49M	36.3

Table 4: Comparisons between the standard MUSE and baselines. The better performance of MUSE_FF over Transformer shows the effectiveness of token features. With dynamic convolutions, the standard MUSE brings larger improvements, showing the effectiveness of local features.

First, as we can see, the performance of MUSE_FF beats the original Transformer with a 0.5 BLEU improvement. MUSE_FF concatenates attention networks and feed-forward networks in a parallel way while the original Transformer serially combines them. Since the attention already fuses global features into token features, it is hard for the feed-forward to keep sufficient features. In MUSE, the

feed-forward networks are directly linked with the input, thus it keeps token-level features well. The better performance of MUSE_FF shows the effectiveness of local features.

To capture more complex local features, we also introduce dynamic convolutions into MUSE. As shown in Table 4, without dynamic convolutions, the performance of MUSE_FF drops from 36.3 to 35.8. It indicates the importance of local features.

3.5 VISUALIZATION ANALYSIS

MUSE contains multiple dynamic convolution cells, whose streams are fused by a gated mechanism. The weight for each dynamic cell is a scalar. Here we analyze the weight of different dynamic convolution cells in different layers. Figure 4 shows that as the layer depth increases, the weight of dynamic convolution cells with small kernel sizes gradually decreases. It demonstrates that lower layers prefer local features while higher layers prefer global features. It is corresponding to the finding in Ramachandran et al. (2019).

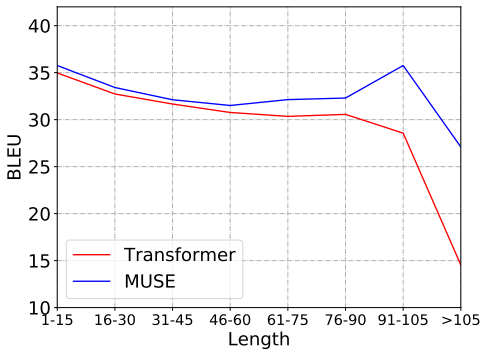

Figure 3: BLEU scores of on different groups with different source sentence lengths. The experiments are conducted on the DE-EN dataset. MUSE performs better than Transformer, espcially on long sentences.

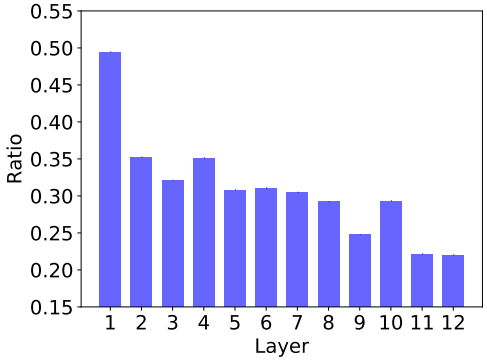

Figure 4: Dynamically selected kernels at each layer: The blue bars represent the ratio between the percentage of the convolution with smaller kernel sizes and the percentage of the convolution with large kernel sizes.

3.6 CASE STUDY

We conduct the case study on the De-En dataset and the cases are shown in Table 5. In case 1, alghough the baseline transformer translates many correct words according to the source sentence, the translated sentence is not fluent at all. It indicates that Transformer does not capture the relationship between some words and their neighbors, such as “right” and “clap”. By contrast, MUSE captures them well by combining local attention with global self-attention. In case 2, the cause adverbial clause is correctly translated by MUSE while transformer misses the word “why” and fails to translate it.

4 RELATED WORK

Text understanding and representation is an essential step in natural language processing. Traditional approaches usually adopt long-short term memory networks (Hochreiter & Schmidhuber, 1997) or convolutional neural networks (CNN) (Krizhevsky et al., 2012) to get the representation of a text sequence. However, these models either are built upon auto-regressive structures requiring longer encoding time or perform worse on real-world natural language processing tasks.

Unlike these approaches, Transformer (Vaswani et al., 2017) drops CNN or LSTM structures and only keeps an attention mechanism. It supports high-parallel sequence modeling and does not require auto-regressive structure during encoding, thus bringing large efficiency improvements. Due to its strong ability in capturing global dependencies, some researchers also apply attention to computer vision tasks (Wang et al., 2018; Bello et al., 2019). Further, Ramachandran et al. (2019) claim

Case 1	
Source	wenn sie denken, dass die auf der linken seite jazz ist und die, auf der rechten seite swing ist, dann klatschen sie bitte.
Target	if you think the one on the left is jazz and **the one on the right is swing, clap your hands.**
Transformer	if you think it's jazz on the left, and **those on the right side of the swing are clapping**, please.
MUSE	if you think the one on the left is jazz, **and the one on the right is swing, please clap.**
Case 2	
Source	und deswegen haben wir uns entschlossen in berlin eine halle zu bauen, in der wir sozusagen die elektrischen verhltnisse der insel im mastab eins zu drei ganz genau abbilden knnen.
Target	and **that's why we decided** to build a hall in berlin, where we could precisely reconstruct, so to speak, the electrical ratio of the island on a one to three scale.
Transformer	and so in berlin, **we decided** to build a hall where we could sort of map the electrical proportions of the island at scale one to three very precisely.
MUSE	and **that's why we decided** to build a hall in berlin, where we can sort of map the electric relationship of the island at the scale one to three very precisely.

Table 5: Case study on the De-En dataset. The blue bolded words denote the **wrong** translation and red bolded words denote the **correct** translation. In case 1, transformer fails to capture the relationship between some words and their neighbors, such as "right" and "clap". In case 2, the cause adverbial clause is correctly translated by MUSE while transformer misses the word "why" and fails to translate it.

that all convolutional nets can be replaced by self-attention in computer vision tasks to improve performance. Furthermore, layer normalization and residual structure make it possible to build a very deep model, enabling the model with a powerful learning ability (Wang et al., 2019). Besides, Zhang et al. (2019a) proposes to build a deep transformer of 12 layers with a large number of parameters.

In recent years, some researches are focusing on exploring simpler or stronger networks than Transformer. Wu et al. (2019) propose a very lightweight convolution network, which is simpler and more efficient than Transformer. So et al. (2019) adopt NAS, a neural architecture search method, to search for a better alternative to Transformer by fusing attention and convolutions together. In this work, we propose a multi-scale attention model by combing global features, local features, and token features together to improve the generalization ability.

5 Conclusion and Future Work

In this work, we rethink a fundamental question: "Is attention really all you need?". We find that attention suffers from dispersed weights especially for long text modeling. To address this problem, we present MUSE, a model that fuses self-attention, convolution, and fully connected layers together to explicitly capture global features, local features, and token features respectively. Beyond the inspiring results on large datasets, exploratory analysis and model ablation also verify the effectiveness of MUSE. Although our empirical results prove the effectiveness of local features and global features, it is still unknown how this information affects model learning. In future work, we would like to explore the detailed effects of global/local features on model learning.

REFERENCES

Karim Ahmed, Nitish Shirish Keskar, and Richard Socher. Weighted transformer network for machine translation, 2017.

Irwan Bello, Barret Zoph, Ashish Vaswani, Jonathon Shlens, and Quoc V. Le. Attention augmented convolutional networks. *CoRR*, abs/1904.09925, 2019.

Mia Xu Chen, Orhan Firat, Ankur Bapna, Melvin Johnson, Wolfgang Macherey, George Foster, Llion Jones, Mike Schuster, Noam Shazeer, Niki Parmar, and et al. The best of both worlds: Combining recent advances in neural machine translation. *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2018. doi: 10.18653/v1/p18-1008. URL http://dx.doi.org/10.18653/v1/p18-1008.

Rewon Child, Scott Gray, Alec Radford, and Ilya Sutskever. Generating long sequences with sparse transformers, 2019.

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc Le, and Ruslan Salakhutdinov. Transformer-xl: Attentive language models beyond a fixed-length context. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019. doi: 10.18653/v1/p19-1285. URL http://dx.doi.org/10.18653/v1/p19-1285.

Mostafa Dehghani, Stephan Gouws, Oriol Vinyals, Jakob Uszkoreit, and ukasz Kaiser. Universal transformers, 2018.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2018.

Jiangtao Feng, Lingpeng Kong, Po-Sen Huang, Chong Wang, Da Huang, Jiayuan Mao, Kan Qiao, and Dengyong Zhou. Neural phrase-to-phrase machine translation, 2018.

Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N Dauphin. Convolutional sequence to sequence learning. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 1243–1252. JMLR. org, 2017.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8): 1735–1780, 1997.

Po-Sen Huang, Chong Wang, Sitao Huang, Dengyong Zhou, and Li Deng. Towards neural phrase-based machine translation. *arXiv preprint arXiv:1706.05565*, 2017.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems 2012. Proceedings of a meeting held December 3-6, 2012, Lake Tahoe, Nevada, United States.*, pp. 1106–1114, 2012.

Junyang Lin, Xu Sun, Xuancheng Ren, Muyu Li, and Qi Su. Learning when to concentrate or divert attention: Self-adaptive attention temperature for neural machine translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 2985–2990, 2018.

Yiping Lu, Zhuohan Li, Di He, Zhiqing Sun, Bin Dong, Tao Qin, Liwei Wang, and Tie-Yan Liu. Understanding and improving transformer from a multi-particle dynamic system point of view, 2019.

Myle Ott, Sergey Edunov, David Grangier, and Michael Auli. Scaling neural machine translation. *arXiv preprint arXiv:1806.00187*, 2018.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. fairseq: A fast, extensible toolkit for sequence modeling. *arXiv preprint arXiv:1904.01038*, 2019.

Prajit Ramachandran, Niki Parmar, Ashish Vaswani, Irwan Bello, Anselm Levskaya, and Jonathon Shlens. Stand-alone self-attention in vision models, 2019.

Peter Shaw, Jakob Uszkoreit, and Ashish Vaswani. Self-attention with relative position representations. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, 2018. doi: 10.18653/v1/n18-2074. URL http://dx.doi.org/10.18653/v1/n18-2074.

David R So, Chen Liang, and Quoc V Le. The evolved transformer. *arXiv preprint arXiv:1901.11117*, 2019.

Sainbayar Sukhbaatar, Edouard Grave, Guillaume Lample, Herve Jegou, and Armand Joulin. Augmenting self-attention with persistent memory, 2019.

Gongbo Tang, Mathias Müller, Annette Rios, and Rico Sennrich. Why self-attention? A targeted evaluation of neural machine translation architectures. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pp. 4263–4272, 2018.

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

Qiang Wang, Bei Li, Tong Xiao, Jingbo Zhu, Changliang Li, Derek F. Wong, and Lidia S. Chao. Learning deep transformer models for machine translation. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019. doi: 10.18653/v1/p19-1176. URL http://dx.doi.org/10.18653/v1/p19-1176.

Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun 2018. doi: 10.1109/cvpr.2018.00813. URL http://dx.doi.org/10.1109/cvpr.2018.00813.

Felix Wu, Angela Fan, Alexei Baevski, Yann N Dauphin, and Michael Auli. Pay less attention with lightweight and dynamic convolutions. *arXiv preprint arXiv:1901.10430*, 2019.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. Xlnet: Generalized autoregressive pretraining for language understanding, 2019.

Biao Zhang, Ivan Titov, and Rico Sennrich. Improving deep transformer with depth-scaled initialization and merged attention, 2019a.

Hongyi Zhang, Yann N Dauphin, and Tengyu Ma. Fixup initialization: Residual learning without normalization. *arXiv preprint arXiv:1901.09321*, 2019b.