

# MUSE: PARALLEL MULTI-SCALE ATTENTION FOR SEQUENCE TO SEQUENCE LEARNING

**Anonymous authors**

Paper under double-blind review

## ABSTRACT

In sequence to sequence learning, the self-attention mechanism proves to be highly effective, and achieves significant improvements in many tasks. However, the self-attention mechanism is not without its own flaws. Although self-attention can model extremely long dependencies, the attention in deep layers tends to over-concentrate on a single token, leading to insufficient use of local information and difficulty in representing long sequences. In this work, we explore parallel multi-scale representation learning on sequence data, striving to capture both long-range and short-range language structures. To this end, we propose the Parallel **M**ulti-**S**cale **a**ttention (**MUSE**) and MUSE-simple. MUSE-simple contains the basic idea of parallel multi-scale representation learning, and it encodes the sequence in parallel, in terms of different scales with the help from self-attention, and pointwise transformation. MUSE builds on MUSE-simple and explores combining convolution and self-attention for learning sequence representations from more different scales. We focus on machine translation and the proposed approach achieves substantial performance improvements over Transformer, especially on long sequences. More importantly, we find that although conceptually simple, its success in practice requires intricate considerations, and the multi-scale attention must build on unified semantic space. Under common setting, the proposed model achieves substantial performance and outperforms all previous models on three main machine translation tasks. In addition, MUSE has potential for accelerating inference due to its parallelism.

## 1 INTRODUCTION

In recent years, Transformer has been remarkably adept at sequence learning 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., 2019b; Dai 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 self-attention mechanism is updating token representations based on a weighted sum of all input representations.

However, recent research (Tang et al., 2018) has shown that the Transformer has surprising shortcomings in long sequence learning, exactly because of its use of self-attention. As shown in Figure 1 (a), in the task of machine translation, the performance of Transformer drops with the increase of the source sentence length, especially for long sequences. The reason is that the attention can be over-concentrated and disperse, as shown in Figure 1 (b), and only a small number of tokens are represented by attention. It may work fine for shorter sequences, but for longer sequences, it causes insufficient representation of information and brings difficulty for the model to comprehend the source information intactly. In recent work, local attention that constrains the attention to focus on only part of the sequences (Child et al., 2019; Sukhbaatar et al., 2019a) is used to address this problem. However, it costs self-attention the ability to capture long-range dependencies and also does not demonstrate effectiveness in sequence to sequence learning tasks.

To build a module with both inductive bias of local and global context modelling in sequence to sequence learning, we hybrid self-attention with convolution and present Parallel multi-scale attention called MUSE. It encodes inputs into hidden representations and then applies self-attention and depth-separable convolution transformations in parallel. The convolution compensates for the in-

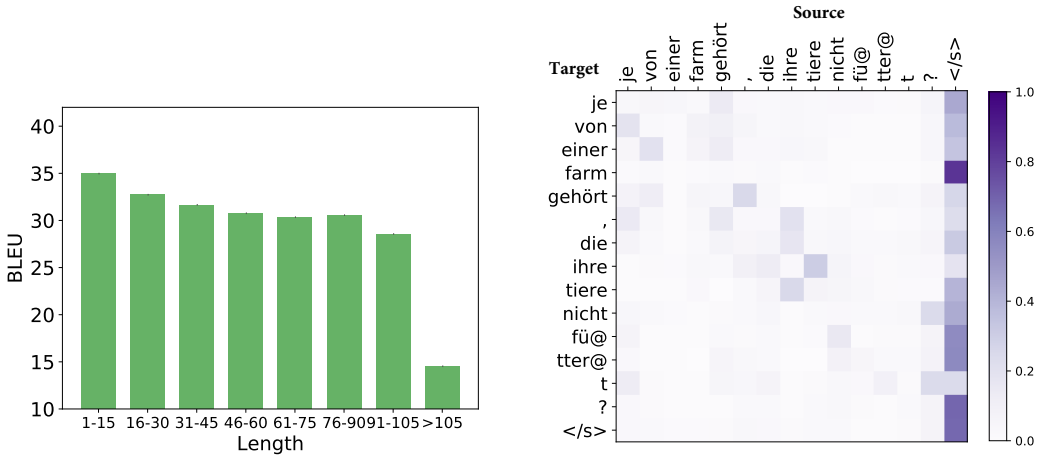


Figure 1: The left figure shows that the performance drops largely with the increase of sentence length on the De-En dataset. The right figure shows the attention map from the 3-th encoder layer. As we can see, the attention map is too dispersed to capture sufficient information. For example, “[EOS]”,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

sufficient use of local information while the self-attention focuses on capturing the dependencies. Moreover, this parallel structure is highly extensible, i.e., new transformations can be easily introduced as new parallel branches, and is also favourable to parallel computation.

The main contributions are summarized as follows:

- We find that the attention mechanism alone suffers from dispersed weights and is not suitable for long sequence representation learning. The proposed method tries to address this problem and achieves much better performance on generating long sequence.
- We propose a parallel multi-scale attention and explore a simple but efficient method to successfully combine convolution with self-attention all in one module.
- MUSE outperforms all previous models with same training data and the comparable model size, with state-of-the-art BLEU scores on three main machine translation tasks.
- The proposed method enables parallel representation learning. Experiments show that the inference speed can be increased by 31% on GPUs.

## 2 MUSE: PARALLEL MULTI-SCALE ATTENTION

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

The encoder is a stack of  $N$  MUSE modules. Residual mechanism and layer normalization are used to connect two adjacent layers. The decoder is similar to encoder, except that each MUSE module in the decoder not only captures features from the generated text representations but also performs attention over the output of the encoder stack through additional context attention. 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, depth-wise separable 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:

$$X_i = X_{i-1} + Attention(X_{i-1}) + Conv(X_{i-1}) + Pointwise(X_{i-1}) \tag{1}$$

where “Attention” refers to self-attention, “Conv” refers to dynamic convolution, “Pointwise” refers to a position-wise feed-forward network. The followings list the details of each part.

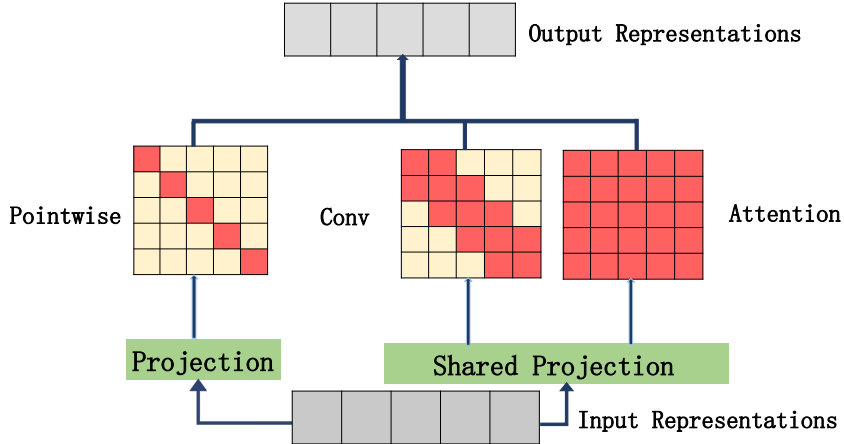


Figure 2: Multi-scale attention hybrids point-wise transformation, convolution, and self-attention to learn multi-scale sequence representations in parallel. We project convolution and self-attention into the same space to learn contextual representations.

We also propose MUSE-simple, a simple version of MUSE, which generates the output representation similar to the MUST model except for that it does not include the convolution operation:

$$X_i = X_{i-1} + Attention(X_{i-1}) + Pointwise(X_{i-1}) \quad (2)$$

## 2.1 ATTENTION MECHANISM FOR GLOBAL CONTEXT REPRESENTATION

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

$$Attention(X) = \sigma(QW^Q, KW^K, VW^V)W^O \quad (3)$$

where  $Q, K, V = Linear_1(X), Linear_2(X), Linear_3(X)$

Where  $W^O, W^Q, W^K$ , and  $W^V$  are projection parameters. The self-attention operation  $\sigma$  is the dot-product between key, query, and value pairs:

$$\sigma(Q_1, K_1, V_1) = \text{softmax}\left(\frac{Q_1 K_1^T}{\sqrt{d_k}}\right)V_1 \quad (4)$$

Note that we conduct a projecting operation over the value in our self-attention mechanism  $V_1 = VW^V$  here.

## 2.2 CONVOLUTION FOR LOCAL CONTEXT MODELING

We introduce convolution operations into MUSE to capture local context. To learn contextual sequence representations in the same hidden space, we choose depth-wise convolution (Chollet, 2017) (we denote it as DepthConv in the experiments) as the convolution operation because it includes two separate transformations, namely, point-wise projecting transformation and contextual transformation. It is because that original convolution operator is not separable, but DepthConv can share the same point-wise projecting transformation with self-attention mechanism. We choose dynamic convolution (Wu et al., 2019a), the best variant of DepthConv, as our 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:

$$Conv_k(X) = Depth\_conv_k(V_2)W^{out} \quad (5)$$

$$V_2 = XW^V$$

where  $W^V$  and  $W^{out}$  are parameters,  $W^V$  is a point-wise projecting transformation matrix. The *Depth\_conv* refers to depth convolution in the work of Wu et al. (2019a). For an input sequence  $X$ , the output  $O$  is computed as:

$$O_{i,c} = Depth\_conv_k(X) = \sum_{j=1}^k \left( \text{softmax} \left( \sum_{c=1}^d W_{j,c}^Q X_{i,c} \right) \cdot X_{i+j-\lceil \frac{k+1}{2} \rceil, c} \right) \quad (6)$$

where  $d$  is the hidden size. Note that we conduct the same projecting operation over the input in our convolution mechanism  $V_2 = XW^V$  here with that in self-attention mechanism.

**Shared projection** To learn contextual sequence representations in the same hidden space, the projection in the self-attention mechanism  $V_1 = VW^V$  and that in the convolution mechanism  $V_2 = XW^V$  is shared. because the shared projection can project the input feature into the same hidden space. If we conduct two independent projection here:  $V_1 = VW_1^V$  and  $V_2 = XW_2^V$ , where  $W_1^V$  and  $W_2^V$  are two parameter matrices, we call it as separate projection. We will analyze the necessity of applying shared projection here instead of separate projection.

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

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

### 2.3 POINT-WISE FEED-FORWARD NETWORK FOR CAPTURING TOKEN REPRESENTATIONS

To learn token level representations, MUSE concatenates an self-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.

$$Pointwise(X) = \max(0, XW_1 + b_1)W_2 + b_2 \quad (8)$$

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

## 3 EXPERIMENT

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

### 3.1 DATASETS

**WMT14 En-Fr and En-De datasets** The WMT 2014 English-French translation dataset, consisting of  $36M$  sentence pairs, is adopted as a big dataset to test our model. 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. Following Gehring et al. (2017), we also adopt a joint source and target BPE factorization with the vocabulary size of  $40K$ . For medium dataset, we borrow the setup of Vaswani et al. (2017) and adopt the WMT 2014 English-German translation dataset which consists of  $4.5M$  sentence pairs, the BPE vocabulary size is set to  $32K$ . The test and validation datasets we used are the same as Vaswani et al. (2017).

**IWSLT De-En and En-Vi datasets** Besides, we perform experiments on two small IWSLT datasets to test the small version of MUSE with other comparable models. 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 the En-Vi task, we build a dictionary including all source and target tokens. The vocabulary size for English is  $17.2K$ , and the vocabulary size for the Vietnamese is  $6.8K$ .

## 3.2 EXPERIMENTAL SETTINGS

### 3.2.1 MODEL

For fair comparisons, we only compare models reported with the comparable model size and the same training data. We do not compare Wu et al. (2019b) because it is an ensemble method. We build MUSE-base and MUSE-large with the parameter size comparable to Transformer-base and Transformer-large. We adopt multi-head attention (Vaswani et al., 2017) as implementation of self-attention in MUSE module. The number of attention head is set to 4 for MUSE-base and 16 for MUSE-large. We also add the network architecture built by MUSE-simple in the similar way into the comparison.

MUSE consist of 12 residual blocks for encoder and 12 residual blocks for decoder, the dimension is set to 384 for MUSE-base and 768 for MUSE-large. The hidden dimension of non linear transformation is set to 768 for MUSE-base and 3072 for MUSE-large.

The MUSE-large is trained on 4 Titan RTX GPUs while the MUSE-base is trained on a single NVIDIA RTX 2080Ti GPU. The batch size is calculated at the token level, which is called dynamic batching (Vaswani et al., 2017). We adopt dynamic convolution as the variant of depth-wise separable convolution. We tune the kernel size on the validation set. For convolution with a single kernel, we use the kernel size of 7 for all layers. In case of dynamic selected kernels, the kernel size is 3 for small kernels and 15 for large kernels for all layers.

### 3.2.2 TRAINING

The training hyper-parameters are tuned on the validation set.

**MUSE-large** For training MUSE-large, following Ott et al. (2018), parameters are updated every 32 steps. We train the model for 80K updates with a batch size of 5120 for En-Fr, and train the model for 30K updates with a batch size of 3584 for En-De. The dropout rate is set to 0.1 for En-Fr and 0.3 for En-De. We borrow the setup of optimizer from Wu et al. (2019a) and use the cosine learning rate schedule with 10000 warmup steps. The max learning rate is set to 0.001 on En-De translation and 0.0007 on En-Fr translation. For checkpoint averaging, following Wu et al. (2019a), we tune the average checkpoints for En-De translation tasks. For En-Fr translation, we do not average checkpoint but use the final single checkpoint.

**MUSE-base** We train and test MUSE-base on two small datasets, IWSLT 2014 De-En translation and IWSLT2015 En-Vi translation. Following Vaswani et al. (2017), we use Adam optimizer with a learning rate of 0.001. We use the warmup mechanism and invert the learning rate decay with warmup updates of 4K. 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 also updated every 4 steps. The dropout rate is set to 0.3. We save checkpoints every epoch and average the last 10 checkpoints for inference.

### 3.2.3 EVALUATION

During inference, we adopt beam search with a beam size of 5 for De-En, En-Fr and En-Vi translation tasks. The length penalty is set to 0.8 for En-Fr according to the validation results, 1 for the two small datasets following the default setting of Ott et al. (2019). We do not tune beam width and length penalty but use the setting reported in Vaswani et al. (2017). The BLEU<sup>1</sup> metric is adopted to evaluate the model performance during evaluation.

## 3.3 RESULTS

As shown in Table 1, MUSE outperforms all previously models on En-De and En-Fr translation, including both state-of-the-art models of stand alone self-attention (Vaswani et al., 2017; Ott et al., 2018), and convolutional models (Gehring et al., 2017; Kaiser et al., 2017; Wu et al., 2019a). This result shows that either self-attention or convolution alone is not enough for sequence to sequence learning. The proposed parallel multi-scale attention improves over them both on En-De and En-Fr.

<sup>1</sup><https://github.com/moses-smt/mosesdecoder/blob/master/scripts/generic/multi-bleu.perl>

Model	En-De	En-Fr
ConvSeq2seq (Gehring et al., 2017)	25.2	40.5
SliceNet (Kaiser et al., 2017)	26.1	-
Transformer (Vaswani et al., 2017)	28.4	41.0
Weighted Transformer (Ahmed et al., 2017)	28.9	41.4
Layer-wise Coordination (He et al., 2018)	29.1	-
Transformer (relative position) (Shaw et al., 2018)	29.2	41.5
Transformer (Ott et al., 2018)	29.3	43.2
Evolved Transformer (So et al., 2019)	29.8	41.3
DynamicConv (Wu et al., 2019a)	29.7	43.2
Local Joint Self-attention (Fonollosa et al., 2019)	29.7	43.3
<b>MUSE-simple</b>	29.8	43.2
<b>MUSE</b>	<b>29.9</b>	<b>43.5</b>

Table 1: MUSE-large outperforms all previous models under the standard training and evaluation setting on WMT14 En-De and WMT14 En-Fr datasets.

Model	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., 2019)	-	34.5
DynamicConv (Wu et al., 2019a)	-	35.2
Macaron (Lu et al., 2019)	-	35.4
Local Joint Self-attention (Fonollosa et al., 2019)	-	35.7
<b>MUSE-simple</b>	30.7	35.8
<b>MUSE</b>	<b>31.3</b>	<b>36.3</b>

Table 2: MUSE-base outperforms previous state-of-the-art models on IWSLT De-En translation datasets and outperforms previous models without BPE processing on IWSLT En-Vi.

Compared to Evolved Transformer(So et al., 2019) which is constructed by NAS and also mixes convolutions of different kernel size, MUSE achieves 2.2 BLEU gains in En-Fr translation.

Relative position or local attention constraints bring improvements over origin self-attention model, but parallel multi-scale outperforms them.

MUSE can also scale to small model and small datasets, as depicted in Table 2, MUSE-base pushes the state-of-the-art from 35.7 to 36.3 on IWSLT De-En translation dataset.

It is shown in Table 1 and Table 2 that MUSE-simple which contains the basic idea of parallel multi-scale attention achieves state-of-the-art performance on three tasks.

### 3.4 HOW DO WE PROPOSE EFFECTIVE PARALLEL MULTI-SCALE ATTENTION?

In this subsection we compare MUSE and its variants on IWSLT 2015 De-En translation to answer the question.

**Does concatenating self-attention with convolution certainly improve the model?** To bridge the gap between point-wise transformation which learns token level representations and self-attention which learns representations of global context, we introduce convolution to enhance our multi-scale attention. As we can see from the first experiment group of Table 3, convolution is important in the parallel multi-scale attention. However, it is not easy to combine convolution and self-attention in one module to build better representations on sequence to sequence tasks. As shown in the first line of both second and third group of Table 3, simply learning local representations by using convolution or depth-wise separable convolution in parallel with self-attention harms the performance.

Model	BLEU
MUSE	36.3
MUSE-simple(without DepthConv)	35.8
substitute DepthConv with Convolution(k=3)	35.2
substitute DepthConv with Convolution(k=5)	35.0
substitute DepthConv with Convolution(k=7)	34.5
DepthConv without shared projection	34.9
DepthConv single kernel(k=3)	36.2
DepthConv single kernel(k=7)	36.2
DepthConv single kernel(k=15)	36.0
DepthConv single kernel(k=31)	35.8
DepthConv single kernel(grow kernels among layers:3,7,15,31)	35.9
DepthConv dynamically selected kernel(k=3,15)	36.3

Table 3: Comparisons between MUSE and its variants on the IWSLT 2015 De-En translation task.

Furthermore, combining depth-wise separable convolution (in this work we choose its best variant dynamic convolution as implementation) is even worse than combining convolution.

**Why do we choose DepthConv and what is the importance of sharing Projection of DepthConv and self-attention?** We conjecture that convolution and self-attention both learn contextual sequence representations and they should share the point transformation and perform the contextual transformation in the same hidden space. We first project the input to a hidden representation and perform a variant of depth-wise convolution and self-attention transformations in parallel. The first two experiments in third group of Table 3 show that validating the utility of sharing Projection in parallel multi-scale attention, shared projection gain 1.4 BLEU scores over separate projection, and bring improvement of 0.5 BLEU scores over MUSE-simple (without DepthConv).

**How much is the kernel size?** Comparative experiments show that the too large kernel harms performance both for DepthConv and convolution. Since there exists self-attention and point-wise transformations, simply applying the growing kernel size schedule proposed in SliceNet (Kaiser et al., 2017) doesn’t work. Thus, we propose to use dynamically selected kernel size to let the learned network decide the kernel size for each layer.

### 3.5 FURTHER ANALYSIS

**Parallel multi-scale attention brings time efficiency on GPUs** The underlying parallel structure (compared to the sequential structure in each block of Transformer) allows MUSE to be efficiently computed on GPUs. For example, we can combine small matrices into large matrices, and while it does not reduce the number of actual operations, it can be better paralleled by GPUs to speed up computation. Concretely, for each MUSE module, we first concentrate  $W^Q, W^K, W^V$  of self-attention and  $W_1$  of point feed-forward transformation into a single encoder matrix  $W^{Enc}$ , and then perform transformation such as self-attention, depth-separable convolution, and nonlinear transformation, in parallel, to learn multi-scale representations in the hidden layer.  $W^O, W_2, W^{out}$  can also be combined a single decoder matrix  $W^{Dec}$ . The decoder of sequence to sequence architecture can be implemented similarly.

In Table 4, we conduct comparisons to show the speed gains with the aforementioned implementation, the batch size is set to one sample per batch to simulate online inference environment. Under the settings, where the numbers of parameters are similar for MUSE and Transformer, about 31% increase in inference speed can be obtained. The experiments use MUSE with 6 MUSE-simple modules and Transformer with 6 base blocks. The hidden size is set to 512. It is worth noticing that for the MUSE structure used in the main experiments, ideally a similar speedup can be witnessed if the computing device is powerful enough. However, such is not the case in our preliminary experiments. We also need to point out the implementation is far from fully optimized and the results are only meant to demonstrate the feasibility of the procedure.

Model	Inference Speed (tokens/s)
Transformer	132
MUSE	173
Acceleration	31%

Table 4: The comparison between the inference speed of MUSE and Transformer.

**Parallel multi-scale attention generates much better long sequence** As demonstrated in Figure 3, MUSE generates better sequences of various length than self-attention, but it is remarkably adept at generate long sequence, e.g. for sequence longer than 100, MUSE is two times better.

**Lower layers prefer local context and higher layers prefer more contextual representations** 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).

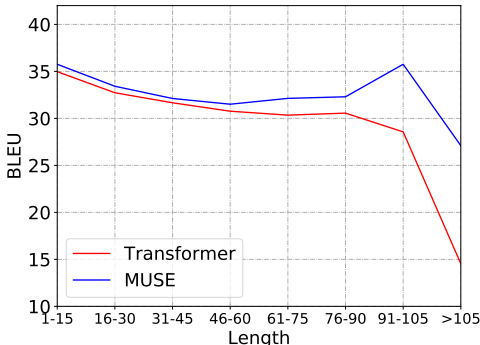


Figure 3: BLEU scores of models on different groups with different source sentence lengths. The experiments are conducted on the De-En dataset. MUSE performs better than Transformer, especially on long sentences.

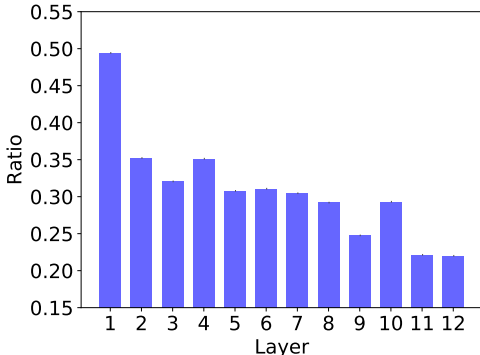


Figure 4: Dynamically selected kernels at each layer: The blue bars represent the ratio between the percentage of the convolution with smaller kernel sizes and the percentage of the convolution with large kernel sizes.

**MUSE not only gains BLEU scores, but also generates more reasonable sentences and increases the translation quality.** We conduct the case study on the De-En dataset and the cases are shown in Table 5. In case 1, although 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 convolution 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

Sequence to sequence learning is an important task in machine learning. It evolves understanding and generating sequence. Machine translation is the touchstone of sequence to sequence learning. Traditional approaches usually adopt long-short term memory networks (Sutskever et al., 2014; Ma et al., 2018) to learn the representation of sequences. However, these models either are built upon auto-regressive structures requiring longer encoding time or perform worse on real-world natural language processing tasks. Recent studies explore convolutional neural networks (CNN) (Gehring et al., 2017) or self-attention (Vaswani et al., 2017) to support high-parallel sequence modeling and



does not require auto-regressive structure during encoding, thus bringing large efficiency improvements. They are strong at capturing local or global dependencies.

There are several studies on combining self-attention and convolution. However, they do not surpass both convolutional and self-attention mechanisms. Sukhbaatar et al. (2019b) propose to augment convolution with self attention by directly concentrating them in computer vision tasks. However, as demonstrated in Table 3 their method does not work for sequence to sequence learning task. Since state-of-the-art models on question answering tasks still consist of self-attention and do not adopt ideas in QANet (Yu et al., 2018). Both self-attention (Ott et al., 2018) and convolution (Wu et al., 2019a) outperforms Evolved transformer by near 2 BLEU scores on En-Fr translation. It seems that learning global and local context through stacking self-attention and convolution layers does not beat either self-attention or convolution models. In contrast, the proposed parallel multi-scale attention outperforms previous convolution or self-attention based models on main translation tasks, showing its effectiveness for sequence to sequence learning.

## 5 CONCLUSION AND FUTURE WORK

Although the self-attention mechanism has been prevalent in sequence modeling, we find that attention suffers from dispersed weights especially for long sequences, resulting from the insufficient local information.

To address this problem, we present Parallel Multi-scale Attention (MUSE) and MUSE-simple. MUSE-simple introduces the idea of parallel multi-scale attention into sequence to sequence learning. And MUSE fuses self-attention, convolution, and point-wise transformation together to explicitly learn global, local and token level sequence representations. Especially, we find from empirical results that the shared projection plays an important part in its success, and is essential for our multi-scale learning.

Beyond the inspiring new state-of-the-art results on three machine translation datasets, detailed analysis and model variants also verify the effectiveness of MUSE.

In future work, we would like to explore the detailed effects of shared projection on contextual representation learning. We are excited about the future of parallel multi-scale attention and plan to apply this simple but effective idea to other tasks including image and speech.

## REFERENCES

- Karim Ahmed, Nitish Shirish Keskar, and Richard Socher. Weighted transformer network for machine translation, 2017.
- Rewon Child, Scott Gray, Alec Radford, and Ilya Sutskever. Generating long sequences with sparse transformers, 2019.
- Francois Chollet. Xception: Deep learning with depthwise separable convolutions. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul 2017. doi: 10.1109/cvpr.2017.195. URL <http://dx.doi.org/10.1109/CVPR.2017.195>.
- 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 Łukasz 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.

- José A. R. Fonollosa, Noe Casas, and Marta R. Costa-jussà. Joint source-target self attention with locality constraints, 2019.
- 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.
- Tianyu He, Xu Tan, Yingce Xia, Di He, Tao Qin, Zhibo Chen, and Tie-Yan Liu. Layer-wise coordination between encoder and decoder for neural machine translation. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (eds.), *Advances in Neural Information Processing Systems 31*, pp. 7944–7954. Curran Associates, Inc., 2018.
- Po-Sen Huang, Chong Wang, Sitao Huang, Dengyong Zhou, and Li Deng. Towards neural phrase-based machine translation. *arXiv preprint arXiv:1706.05565*, 2017.
- Lukasz Kaiser, Aidan N. Gomez, and Francois Chollet. Depthwise separable convolutions for neural machine translation, 2017.
- 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.
- Shuming Ma, Xu Sun, Yizhong Wang, and Junyang Lin. Bag-of-words as target for neural machine translation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 332–338, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-2053. URL <https://www.aclweb.org/anthology/P18-2053>.
- 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, Piotr Bojanowski, and Armand Joulin. Adaptive attention span in transformers. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019a. doi: 10.18653/v1/p19-1032. URL <http://dx.doi.org/10.18653/v1/p19-1032>.
- Sainbayar Sukhbaatar, Edouard Grave, Guillaume Lample, Herve Jegou, and Armand Joulin. Augmenting self-attention with persistent memory, 2019b.
- Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. Sequence to sequence learning with neural networks, 2014.

- 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.
- 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*, 2019a.
- Lijun Wu, Yiren Wang, Yingce Xia, Fei Tian, Fei Gao, Tao Qin, Jianhuang Lai, and Tie-Yan Liu. Depth growing for neural machine translation. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019b. doi: 10.18653/v1/p19-1558. URL <http://dx.doi.org/10.18653/v1/p19-1558>.
- Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. Xlnet: Generalized autoregressive pretraining for language understanding, 2019.
- Adams Wei Yu, David Dohan, Minh-Thang Luong, Rui Zhao, Kai Chen, Mohammad Norouzi, and Quoc V. Le. Qanet: Combining local convolution with global self-attention for reading comprehension, 2018.
- Hongyi Zhang, Yann N Dauphin, and Tengyu Ma. Fixup initialization: Residual learning without normalization. *arXiv preprint arXiv:1901.09321*, 2019.

## A APPENDIX

## A.1 CASE STUDY

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