Mask Attention Networks: Rethinking and Strengthen Transformer

Zhihao Fan[1]*, Yeyun Gong[2], Dayiheng Liu[3], Zhongyu Wei[1,6]†, Siyuan Wang[1],
Jian Jiao[4], Nan Duan[2], Ruofei Zhang[4], Xuanjing Huang[5]
[1]School of Data Science, Fudan University, China
[2]Microsoft Research Asia, [3]DAMO Academy, [4]Microsoft
[5]School of Computer Science, Fudan University, China
[6]Research Institute of Intelligent and Complex Systems, Fudan University, China
{fanzh18,zywei,wangsy18,xjhuang}@fudan.edu.cn, liudayiheng.ldyh@alibaba-inc.com
{yegong,Jian.Jiao,nanduan,bzhang}@microsoft.com

Abstract

Transformer is an attention-based neural network, which consists of two sublayers, namely, Self-Attention Network (SAN) and Feed-Forward Network (FFN). Existing research explores to enhance the two sublayers separately to improve the capability of Transformer for text representation. In this paper, we present a novel understanding of SAN and FFN as Mask Attention Networks (MANs) and show that they are two special cases of MANs with static mask matrices. However, their static mask matrices limit the capability for localness modeling in text representation learning. We therefore introduce a new layer named dynamic mask attention network (DMAN) with a learnable mask matrix which is able to model localness adaptively. To incorporate advantages of DMAN, SAN, and FFN, we propose a sequential layered structure to combine the three types of layers. Extensive experiments on various tasks, including neural machine translation and text summarization demonstrate that our model outperforms the original Transformer.

1 Introduction

Recently, Transformer (Vaswani et al., 2017) has been widely applied in various natural language processing tasks, such as neural machine translation (Vaswani et al., 2017) and text summarization (Zhang et al., 2019). To further improve the performance of the text representation, Transformer-based variants have attracted a lot of attention (Lu et al., 2019; Sukhbaatar et al., 2019a,b; Bugliarello and Okazaki, 2019; Ma et al., 2020).

Each building block of Transformer has two sublayers: Self-Attention Network (SAN) and Feed-Forward Network (FFN). Shaw et al. (2018) presents an extension to SAN which incorporates the relative positional information for the sequence. Sukhbaatar et al. (2019a) proposes attention span to control the maximum context size used in SAN and scales Transformer to long-range (∼ 8192 tokens) language modeling. Recently, some works targeting on FFN have been proposed. Lu et al. (2019) gives a new understanding of Transformer from a multi-particle dynamic system point of view and designs a macaron architecture following Strang-Marchuk splitting scheme. Sukhbaatar et al. (2019b) regards the FFN as the persistent memory in SAN to augment SAN. These works focus on enhancing SAN or FFN, but neglect the inner relationship between SAN and FFN that hinders further improvement.

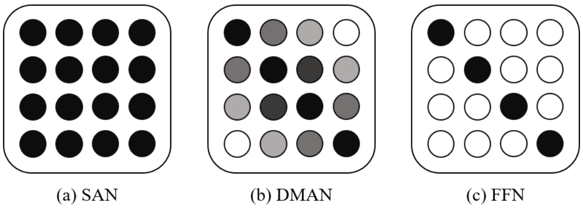


Figure 1: The mask matrices of (a) SAN, (b) DMAN and (c) FFN in Mask Attention Networks. Color that fades from black to white means the values in mask matrices decrease from 1 to 0.

In this work, we present a more systematic analysis for both SAN and FFN to reveal their connections. We introduce **Mask Attention Networks**(MANs), in which each network has a mask matrix that element-wise multiplies a key-query attention matrix. We show that SAN and FFN are two special cases in MANs with static mask matrices. The mask matrix of SAN is an all-ones matrix, while that of FFN is an identity matrix, which is shown as (a) and (c) in Figure 1. Since the mask matrix of SAN has no restriction on relationship modeling with other tokens, SAN is expert in long-range dependency modeling and capture the global

[*] Work is done during internship at Microsoft Research Asia.

[†] Corresponding author.

Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 1692–1701
June 6–11, 2021. ©2021 Association for Computational Linguistics

semantics. In contrast, mask of FFN disables it to perceive the information of other tokens and forces it into self-evolution. We believe that these two specialties endowed by two mask matrices make the success of Transformer in text representation.

Although positive results of Transformer have been reported, recent works (Shaw et al., 2018; Yang et al., 2018; Guo et al., 2019) have shown that modeling localness would further improve the performance through experiments. We argue that deficiency of Transformer in local structure modeling is caused by the attention computation with static mask matrix. In the framework of MANs, we find a problem that irrelevant tokens with overlapping neighbors incorrectly attend to each other with relatively large attention scores. For example "a black dog jump to catch the frisbee", though "catch" and "black" are neither relevant nor neighbors, for the reason that both of them are highly related to their common neighbor "dog" in attention, we demonstrate that the attention score from "catch" to "black" would be large, which also decreases the attention score from "catch" to "frisbee". The issue in self-attention not only introduces noise to the semantic modeling, but also mislead query tokens to overlook these neighbor tokens. This reveals that self-attention is insufficient in localness modeling and inspires us to mask tokens that not appear in neighborhood.

To strengthen Transformer in localness modeling with better keeping the advantage of SAN and FFN, we propose a Dynamic Mask Attention Network (DMAN) as shown in Figure 1(b), which originates from MANs. Observations reveal that tokens have different ranges of neighbors, for example, that of "dog", which is also connected with "frisbee", is larger than "black" and "catch". Instead of being static that determined in advance, the mask matrix of DMAN is dependent on the query context and relative distance. In DMAN, the tokens in a specific neighborhood are able to receive more attention beyond the normal self-attention mechanism. The dynamic endows DMAN with text representation in different scales, and we validate the superiority through experiments. In Transformer (Vaswani et al., 2017), SAN and FFN cooperate in a sequential layered structure SAN→FFN. Considering SAN, FFN, and DMAN all belong to MANs and have different advantages in text representation, instead of directly replacing SAN in previous works (Shaw et al., 2018; Yang et al., 2018; Guo et al., 2019), we propose to incorporate them with the architecture DMAN→SAN→ FFN.

The main contributions of this work are three-fold:

- We introduce Mask Attention Networks and reformulate SAN and FFN to point out that they are two special cases with static mask in MANs. We analyze the advantages of SAN and FFN in text representation learning and demonstrate that they are insufficient for localness modeling.
- Inspired by the different specialities of SAN and FFN, we propose Dynamic Mask Attention Network (DMAN) to model localness more effectively. We investigate the different collaboration methods of SAN, FFN, and DMAN, and propose a sequential layered structure DMAN→SAN→FFN.
- We conduct experiments on machine translation and abstract summarization. Experimental results show that our method outperforms original Transformer. We also perform ablation study to verify the effectiveness of different modules of our proposed model.

2 Model

In § 2.1, we review the Transformer architecture. We introduce Mask Attention Networks and reformulate SAN and FFN to point out they are two special cases in § 2.2, and analyze their deficiency in localness modeling in § 2.3. Then, in § 2.4, we describe Dynamic Mask Attention Network (DMAN) in detail. At last, in § 2.5, we discuss the collaboration of DMAN, SAN and FFN.

2.1 Transformer

Transformer has two sublayers: Self-Attention Network (SAN) and Feed-Forward Network (FFN).

As discussed in Vaswani et al. (2017), an attention function maps a query and a set of key-value pairs to an output shown in Equation 1.

$$\begin{aligned}\mathcal{A}(Q,K,V) &= \mathcal{S}(Q,K)V \\ \mathcal{S}(Q,K) &= \left[\frac{\exp\left(Q_iK_j^T/\sqrt{d_k}\right)}{\sum_k \exp\left(Q_iK_k^T/\sqrt{d_k}\right)}\right]\end{aligned} \tag{1}$$

where the queries Q, keys K and values $V \in \mathbb{R}^{T\times d_k}$ are all matrices.

SAN produces representations by applying attention function to each pair of tokens from the input sequence. It is beneficial to capture different contextual features with multiple individual attention functions. Given a text representation sequence $H^l \in \mathbb{R}^{T\times d}$. in the l-the layer.

$$\begin{aligned} H^l &= \left[A^1, \cdots, A^I\right] W_H \\ A^i &= \mathcal{A}\left(H^l W_Q^i, H^l W_K^i, H^l W_V^i\right) \end{aligned} \tag{2}$$

where $\{W_Q^i, W_K^i, W_V^i\} \in R^{d\times d_k}$ are trainable parameters, i denotes the attention head and d is the hidden size.

In FFN, the computation of each h_t^l in H^l is independent of others. It consists of two affine transformations with a pointwise non-linear function:

$$H^{l+1} = \text{ReLU}\left(H^l W_1\right) W_2 \tag{3}$$

where W_1 and W_2 are matrices of dimension $d \times d_f$ and $d_f \times d$, respectively. Typically, d_f is set to be 4 times larger than d.

2.2 Mask Attention Networks

On the basis of attention function in Equation 1, we define a new mask attention function:

$$\begin{aligned} \mathcal{A}_M(Q, K, V) &= \ \mathcal{S}_M(Q, K)V \\ \mathcal{S}_M(Q, K) &= \left[\frac{M_{i,j}\exp\left(Q_i K_j^T/\sqrt{d_k}\right)}{\sum_k M_{i,k}\exp\left(Q_i K_k^T/\sqrt{d_k}\right)}\right] \end{aligned} \tag{4}$$

where $M \in \mathbb{R}^{T\times T}$, $M_{i,j} \in [0, 1]$ is a mask matrix and can be static or dynamic. Intuitively, the value in each position of M can be viewed as the color shade in Figure 1.

With the knowledge of mask attention function, we introduce **Mask Attention Networks**(MANs), in which each network can be written as Equation 5.

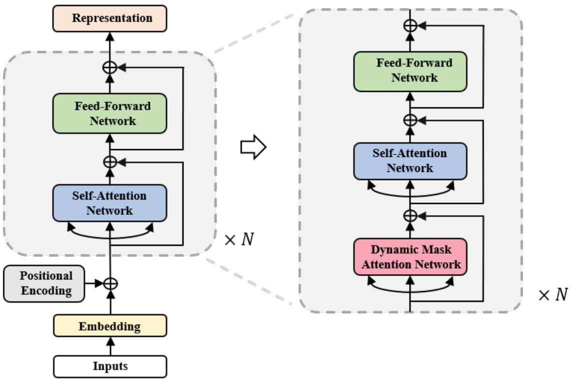
$$\begin{aligned} H^{l+1} &= \mathcal{F}\left(\left[A_{M^1}^1, \cdots, A_{M^I}^I\right]\right) W_H \\ A_{M^i}^i &= \mathcal{A}_{M^i}\left(H^l W_Q^i, H^l W_K^i, H^l W_V^i\right) \end{aligned} \tag{5}$$

where $\mathcal{F}$ is the activation function, M^i is the mask matrix for the i-th attention head.

Next, we show that SAN and FFN both belong to the Mask Attention Networks.

For SAN, let $M = [1] \in \mathbb{R}^{T\times T}$ be an all-ones matrix and $\mathcal{F} = \mathcal{F}_{id}$ be the identity function, its mask attention function would be formalized:

$$\begin{aligned} \mathcal{S}_{[1]}(Q, K) &= \left[\frac{1 \cdot \exp\left(Q_i K_j^T/\sqrt{d_k}\right)}{\sum_k \exp\left(Q_i K_k^T/\sqrt{d_k}\right)}\right] = \mathcal{S}(Q, K) \\ \mathcal{A}_{[1]}(Q, K, V) &= \mathcal{S}_{[1]}(Q, K)V = \mathcal{A}(Q, K, V) \end{aligned} \tag{6}$$



a) Transformer: SAN→FFN

b) DMAN→SAN→FFN

Figure 2: Overview of our proposed model. Left is the Transformer architecture, right is our DMAN→SAN→FFN one.

Then, the MAN degenerates into SAN.

$$\begin{aligned} H^{l+1} &= \mathcal{F}_{id}\left(\left[A_{[1]}^1, \cdots, A_{[1]}^h\right]\right) W_H \\ &= \left[A^1, \cdots, A^h\right] W_H \end{aligned} \tag{7}$$

For FFN, let $M = \mathbb{I} \in \mathbb{R}^{T\times T}$ be the identity matrix, $\mathcal{F} = \text{ReLU}$ and head number $I = 1$.

$$\begin{aligned} \mathcal{S}_{\mathbb{I}}(Q, K) &= \left[\frac{1_i(j) \cdot \exp\left(Q_i K_j^T/\sqrt{d_k}\right)}{\sum_k 1_i(k) \cdot \exp\left(Q_i K_k^T/\sqrt{d_k}\right)}\right] = \mathbb{I} \\ \mathcal{A}_{\mathbb{I}}(Q, K, V) &= \mathcal{S}_{\mathbb{I}}(Q, K)V = \mathbb{I}V = V \end{aligned} \tag{8}$$

where $1_i(x)$ is an indicator function that equal to 1 if $x = i$, otherwise 0.

The MAN degenerates into FFN.

$$H^{l+1} = \text{ReLU}\left(\left[A_M^1\right]\right) W_H = \text{ReLU}\left(H^l W_V^1\right) W_H \tag{9}$$

In summary, SAN and FFN are two special cases in MANs with different static mask matrices.

2.3 Deficiency of SAN and FFN in Localness Modeling

The mask matrix of SAN is an all-ones matrix and that of FFN is an identity matrix, they are two extreme cases in MANs. We analyze that these two static MANs are deficient in localness modeling. Intuitively, through blocking other tokens in advance, FFN focuses on its own information and is unable to perceive the information except itself, let alone its neighbors. In SAN, each token is equally accessible to any other ones. As the example in Introduction shows, we find that tokens not in neighborhood are also likely to attend to each other with relatively large scores. Therefore, SAN might introduce noises to semantic modeling and overlook the relation of neighboring signals.

We demonstrate the issue of self-attention. Generally assuming that $\left[a, b, c\right]$ appear in sequence, and (a, b), (b, c) are two neighbor pairs, but a, c are not neighbors.

First, to explicitly define the relationship of tokens, we introduce $U_\delta(h)$ as the set of tokens at the distance of δ from h with key and query linear transformation in SAN, in other words, $u \in U_\delta(h) \Leftrightarrow ||hW_Q - uW_K||_2^2 \leq \delta$. For example, if (a, b) is a neighbor pair, there would exist some small $\delta \geq 0$ such that $a \in U_\delta(b)$ and $b \in U_\delta(a)$.

Second, we know that the larger the inner product is, the smaller the Euclidean distance is, and vice versa. With the awareness of the relationships between $\left[a, b, c\right]$, we have $a, b \in U_\delta(a)$, $b, c \in U_\delta(c)$ and $a, b, c \in U_\delta(b)$ for some small $\delta \geq 0$.

Third, we are able to estimate the semantic distance between a and c as the Equation 10 shows.

$$\begin{aligned}
&||aW_Q - cW_K||_2^2 \\
=&||aW_Q - bW_K + bW_K - bW_Q + bW_Q - cW_K||_2^2 \\
\leq&3||aW_Q - bW_K||_2^2 + 3||bW_K - bW_Q||_2^2 \\
+&3||bW_Q - cW_K||_2^2) \leq 9\delta
\end{aligned} \tag{10}$$

Thus, though a and c are not neighbors, no matter how irrelevant the semantics of a and c, $c \in U_{9\delta}(a)$ that c would play an important role in modeling semantics of a.

The upper phenomenon illustrates following normal attention function in Equation 1, some tokens not in neighborhood not are still likely to occupy an important position in attention weight that can not be ignored.

2.4 Dynamic Mask Attention Network

With the knowledge of MANs, we propose to mask other tokens that not in neighborhood of the target token for better local semantic modeling.

For example, we build a distance-dependent mask matrix SM. If each token only model the relationship with those tokens within b units of itself, we can set

$$\mathrm{SM}[t, s] = \begin{cases} 0, & \mid t - s \mid > b \\ 1, & \mid t - s \mid \leq b \end{cases} \tag{11}$$

where t, s are the positions of query and key, and $\mathrm{SM}[t, s]$ is the value of the t-th row and s-th column of SM .

By means of SM, we take those tokens within b units into account and ignore others. The static mask does assign more weights to a specific neighborhood, but lacks flexibility. Considering the neighborhood size varies with different query tokens, number of tokens that benefit for different query tokens' local semantic representation are different. Moreover, their mask matrices should match different attention heads and layers in MANs.

We propose Dynamic Mask Attention Network (DMAN) that replaces the static mask matrix. Incorporating query tokens, relative distance, attention head and layer, we build a dynamic mask function which replaces the hard $0/1$ mask gate in Equation 11 with a soft one through sigmoid activation function in Equation 12.

$$\mathrm{DM}_i^l[t, s] = \sigma\Big(h_t^l W^l + P_{t-s}^l + U_i^l\Big) \tag{12}$$

where s, t are the positions of query and key, i is the attention head, l is the layer. P_{t-s}^l is parameterized scalar for the positions t and s, U_i^l is for the i-th head, and $W^l \in \mathbb{R}^{d \times 1}$. W^l, P_{t-s}^l and U_i^l are trainable parameters.

2.5 Collaboration of Mask Attention Networks

Until here, we have three sub-networks of MANs, namely, SAN, FFN and DMAN. SAN that does not mask any tokens and specializes in global semantic modeling. FFN that masks all tokens except itself and focuses on self-processing. DMAN masks the tokens not in neighborhood and is able to model local structure more effectively.

Transformer is composed of SAN and FFN that achieves positive results in various NLP tasks, the stacking method of Transformer inspires us to stack DMAN, SAN and FFN to incorporate their advantages. We insert DMAN in the manner of DMAN→SAN→FFN, which is shown in Figure 2. With this architecture, we first model the localness then globalness, and take the step for self-evolution in the end.

3 Experiments

In this section, we introduce our experiments. We first describe the experimental details in § 3.1. Then we show the experimental results in § 3.2.

Model	IWSLT14 De-En		WMT14 En-De			
	small	params	base	params	big	params
Transformer (Vaswani et al., 2017)	34.4	36M	27.3	62M	28.4	213M
Convolutional Transformer (Yang et al., 2019b)	-	-	28.2	88M	28.7	-
Weighted Transformer (Ahmed et al., 2017)	-	-	28.4	65M	28.9	213M
Local Transformer (Yang et al., 2018)	-	-	28.5	89M	29.2	268M
Relative Transformer (Shaw et al., 2018)	-	-	26.8	-	29.2	-
Scaling NMT (Ott et al., 2018)	-	-	-	-	29.3	213M
Dynamic Conv (Wu et al., 2019)	35.2	-	-	-	29.7	213M
Ours	**36.3**	37M	**29.1**	63M	**30.4**	215M

Table 1: Translation performance (BLEU) on IWSLT14 De-En and WMT14 En-De testsets.

Finally we conduct the ablation study and analysis in § 4.

3.1 Experimental Setting

3.1.1 Machine Translation

Machine translation is an important application of natural language processing (Vaswani et al., 2017). We evaluate our methods on two widely used public datasets: IWSLT14 German-to-English (De-En) and WMT14 English-to-German (En-De). IWSLT14 De-En dataset consists of about 153K/7K/7K sentence pairs for training/validation/testing. WMT14 En-De dataset consists of about 4.5M sentence pairs, and the models were validated on newstest2013 and examined on newstest2014.

Our data processing follows Lu et al. (2019). For IWSLT2014, we set our model into the small one, the hidden size, embeddings and attention heads to 512, 512, and 4 respectively. For the WMT14 dataset, following the Transformer setting of Vaswani et al. (2017), we set our model into the base and big ones which both consist of a 6-layer encoder and 6-layer decoder, the hidden nodes are set to 512 and 1024, and the number of attention heads are 8 and 16. For each setting (small, base and big), we replace all layers in Transformer by our MAN layer. To make a relatively fair comparison, we set the dimensionality of the inner-layer of the FFN in the MAN layers to two times of the dimensionality of the hidden states.

We train our proposed model with cross-entropy with 0.1 label smoothing rate. Inverse-sqrt learning rate scheduler are employed, the peak learning rates are 1.5e-2, 1e-2 and 7e-3 with 8k warmup, 50k update, 80k update and 80k update for transformer big, base and small model with max-tokens 4096, 12288 and 8192 per batch. The dropout rates are 0.3, 0.1 and 0.3 for small, base and big models. The optimizer of model is Adam with (0.9,0.98). The beam size and length penalty for base and big models are 4 and 0.6, for small model is 5 and 1.0. The base and large model are trained on 8 V100 GPUs, and the small model is trained on 2 P40.

3.1.2 Abstract Summarization

Automatic summarization aims to produce a concise and fluent summary conveying the key information in the input text. We focus on abstractive summarization, a generation task where the summary is not limited in reusing the phrases or sentences in the input text. We use the CNN/Daily Mail (See et al., 2017) and Gigaword (Rush et al., 2015) for model evaluation.

Following Song et al. (2019), we set the hidden size, embeddings and attention heads to 768, 768, and 12 respectively. Our model consists of a 6-layer encoder and 6-layer decoder. For the convenience of comparison, the training follows classic seq2seq model without copy, converge or RL. We remove duplicated trigrams in beam search (Paulus et al., 2018). Moreover, the dimensionality of the inner-layer of the FFN in the MAN layers is set to two times of the dimensionality of the hidden states.

In training, inverse-sqrt learning rate scheduler is employed. The peak learning rates are 1e-3 and 8e-4, max-tokens per batch are 8192 and 12288 for CNN/Daily Mail and Gigaword, respectively. The warmup steps is 8k and the total updates is 50k. The optimizer of model is Adam with (0.9,0.98). The dropout and clip-norm are both 0.1. During decoding, the beam size are both 5, the max length and length penalty are 50 and 2.0 for CNN/Daily

Mail, 30 and 1.0 for Gigaword. The models are trained on 4 P40 GPUs.

3.2 Experimental Results

3.2.1 Machine Translation

In machine translation, BLEU (Papineni et al., 2002) is employed as the evaluation measure. Following common practice, we use tokenized case-sensitive BLEU and case-insensitive BLEU for WMT14 En-De and IWSLT14 De-En, respectively. We take Transformer (Vaswani et al., 2017) as the baseline and compare with other concurrent methods. Convolutional Transformer (Yang et al., 2019b) restricts the attention scope to a window of neighboring elements in order to model locality for self-attention model. Local Transformer (Yang et al., 2018) casts localness modeling as a learnable Gaussian bias, which indicates the central and scope of the local region to be paid more attention.

The results for machine translation are shown in Table 1. Our model exceeds the baseline Transformer and other models. For the IWSLT14 dataset, our small model outperforms the Transformer small by 1.6 points in terms of BLEU. For the WMT14 dataset, our base model exceeds its Transformer counterpart by 1.8 BLEU points. Furthermore, the performance of our base model is even better than that of the Transformer big model reported in (Vaswani et al., 2017), but with much less parameters. Our big model outperforms the Transformer big by 2.0 BLEU points.

Compare with Convolutional Transformer and Local Transformer, our model also achieve 1.7 and 1.2 points improvement in BLEU, respectively. This validates that the superiority of our model to systematically solve the localness modeling problem in Transformer.

3.2.2 Abstractive Summarization

We use the F1 score of ROUGE (Lin and Hovy, 2003) as the evaluation metric[1]. In Table 2, we compare our model against the baseline Transformer (Vaswani et al., 2017) and several generation models on CNN/Daily Mail and Gigaword. LEAD3 (Nallapati et al., 2016) extracts the first three sentences in a document as its summary. PT-GEN+Converage (See et al., 2017) is a sequence-to-sequence model based on the pointer-generator network. As shown in Table 2, our model outperforms Transformer by 1.4 in ROUGE-1, 2.2 in ROUGE-2 and 1.2 in ROUGE-L in CNN/Daily Mail. In Gigaword dataset, ours exceeds the baseline by 0.7 in ROUGE-1, 0.5 in ROUGE-2 and 0.7 in ROUGE-L.

As a summary, in machine translation and abstractive summarization our proposed model achieves better results than the Original Transformer (Vaswani et al., 2017).

4 Further Analysis

In this section, we conduct further analysis for our model. We first investigate stacking methods for different sublayers in § 4.1. Then we compare strategies of static mask and dynamic mask in § 4.2. Finally, we analyse the behavior of SAN and DMAN in localness modeling through attention scores in § 4.3.

4.1 Investigate Stacking Methods for Different Sublayers

Here, we investigate different collaboration mechanisms of the elements in MANs. Under our design principles, there are three elements: FFN, SAN, and DMAN. For the convenience of comparison, we take FFN as the last component in the sequential layered structure. We try different collaboration methods and test them on IWSLT2014 German-to-English (De-En). The results are shown in the Table 3. We conclude that:

1. Our proposed $C\#5$ achieves the best performance that verify the effectiveness of our proposed sequential layered structure.

2. All of $C\#3$, $C\#4$ and $C\#5$ outperform $C\#1$ and $C\#2$, and the least improvement in BLEU is 0.2. This shows that no matter what collaboration method, models with the participation of DMAN perform better than models without DMAN, which validates the capability of DMAN.

3. Both $C\#5$ and $C\#4$ are better than $C\#3$ and $C\#2$. This indicates that models without DMAN or SAN are not comparable to models with all three modules. This shows that DMAN and SAN have their own strengths, namely, localness modeling and globalness modeling, and are able to make up for each other's defects through collaboration.

4. $C\#5$ is better than $C\#4$. This indicates that

[1] https://github.com/pltrdy/files2rouge

Model	CNN/Daily Mail				Gigaword			
	R-1	R-2	R-L	R-avg	R-1	R-2	R-L	R-avg
LEAD-3 (Nallapati et al., 2016)	40.42	17.62	36.67	31.57	-	-	-	-
PTGEN+Coverage (See et al., 2017)	39.53	17.28	36.38	31.06	-	-	-	-
Dynamic Conv (Wu et al., 2019)	39.84	16.25	36.73	30.94	-	-	-	-
Transformer (Vaswani et al., 2017)	39.50	16.06	36.63	30.73	37.57	18.90	34.69	30.38
Ours	**40.98**	**18.29**	**37.88**	**32.38**	**38.28**	**19.46**	**35.46**	**31.06**

Table 2: Evaluation results on CNN/Daily Mail and Gigaword. R is short for ROUGE.

#	Method	BLEU
C#1	FFN→SAN→FFN	35.51
C#2	SAN→SAN→FFN	35.66
C#3	DMAN→DMAN→FFN	35.86
C#4	SAN→DMAN→FFN	35.91
C#5	DMAN→SAN→FFN	**36.35**

Table 3: Performance of different collaboration methods of DMAN, SAN and FFN. We evaluate on IWSLT2014 De-En.

model	BLEU
Transformer	34.40
SMAN$_1$	35.52
SMAN$_2$	35.55
DMAN	**36.35**

Table 4: Performance of SMAN and DMAN on IWSLT2014 De-En.

first modeling the localness and then globalness would be better than the inverse order.

4.2 Static Mask and Dynamic Mask

In this section, we compare the performance of Static Mask Attention Network (SMAN) and Dynamic Mask Attention Network (DMAN). Both of them follow the collaboration strategy of DMAN(SMAN)→SAN→FFN. In SMAN, we set a fixed mask boundary which has been determined in advance following Equation 11. Empirically, we propose two static mask strategies: (a) SMAN$_1$, the boundary b depends on sentence length L, $b = \sqrt{L}/2$; (b) SMAN$_2$, b is set to 4, which is chosen from 2, 4, 6, 8 through validation.

The results in IWSLT2014 De-En are shown in Table 4. The performance of SMAN$_1$ and SMAN$_2$ are very close. They both outperform the Transformer but fall behind our proposed DMAN. This indicates that our proposed DMAN is superior to SMAN. SMAN fails to manage various neighborhood for different query tokens, but DMAN can model localness with more flexibility according to these factors.

4.3 Analysis of DMAN in Localness Modeling

In this section, we analyse the behavior of DMAN and SAN in localness modeling through attention scores in Equation 4. To quantify the role of neighbors in semantic modeling, we compute the sum of attention scores within some particular window size. Generally, if the attention score from a to c is bigger than b to c, we consider that a contributes more to the semantic modeling of c compared to b, in other words, model utilizes more information of a than b to learn the semantic representation of c. Therefore, larger attention scores mean that model utilizes more information of the corresponding tokens to learn the semantic representation of query token.

For each sentence in dataset $X_i = (x_{i,1}, \cdots, x_{i,T_i}) \in \mathcal{D}$, we utilize $\bar{s}^l_{i,DMAN}$ and $\bar{s}^l_{i,SAN} \in \mathbb{R}^{T_i \times T_i}$ to denote the average attention scores $\mathcal{S}_M(Q, K)$ in Equation 4 across different heads in the l-th layer for DMAN and SAN, respectively. We sum the attention scores of these tokens $x_{i,k}$ within the window size w of the query $x_{i,j}$ in the l-th layer, and average the sum across X_i and dataset $\mathcal{D}$ following Equation 13.

$$attn\_s_{w,l,*} = \frac{1}{|\mathcal{D}|} \sum_{X_i \in \mathcal{D}} \frac{1}{T_i} \sum_{x_{i,j} \in X_i} \sum_{|k-j| \le w} \bar{s}^l_{i,*}[j,k] \tag{13}$$

where $* \in \{DMAN, SAN\}$, and $\bar{s}^l_{i,*}[j,k]$ is the

	w	#1	#3	#6
DMAN	1	76.58	60.43	60.86
SAN	1	12.80	40.39	45.55
DMAN	2	86.17	75.56	73.89
SAN	2	18.73	45.62	52.72
DMAN	4	95.09	86.20	85.58
SAN	4	30.38	55.17	62.77

Table 5: The values of attention scores $attn\_s_{w,l,DMAN}$ and $attn\_s_{w,l,SAN}$, which is shown in Equation 13. $\mathcal{D}$ is the test set of IWSLT14 De-En, window size $w = 1, 2, 4$ and encoder layers $l = 1, 3, 6$.

value of the j-th row and k-th column of $\bar{s}^l_{i,*}$. $attn\_s_{w,l,*}$ measures the overall contribution of these neighbor tokens within the window size w to the query tokens' semantic modeling. We take $\mathcal{D}$ as the test set of IWSLT14 De-En and compute $attn\_s_{w,l,*}$ with $w = 1, 2, 4$ and $l = 1, 3, 6$.

The result is shown in Table 5. We see that in layer#1, #3 and #6, the sum attention scores of DMAN within the window size 2 are 50% more than those of SAN, especially in layer#1 where the gap is as much as five times between SAN and DMAN. This phenomenon validates that the attention scores of DMAN in neighbors are larger than those of SAN, thus DMAN is more specialized in localness modeling than SAN.

5 Related Work

Recently, there is a large body of work on improving Transformer (Vaswani et al., 2017) for various issues. For recurrence modeling, Hao et al. (2019) introduces a novel attentive recurrent network to leverage the strengths of both attention and recurrent networks. For context modeling, Yang et al. (2019a) focuses on improving self-attention through capturing the richness of context and proposes to contextualize the transformations of the query and key layers. Wu et al. (2019) introduces dynamic convolutions to predict separate convolution kernels solely based on the current time-step in order to determine the importance of context elements. In order to adjust attention weights beyond SAN, Shaw et al. (2018) extends the self-attention mechanism to efficiently consider representations of the relative positions or distances between sequence elements through adding a relative position embedding to the key vectors; Bugliarello and Okazaki (2019) transfers the distance between two nodes in dependency trees with a pre-defined Gaussian weighting function and multiply the distance with the key-query inner product value; Dai et al. (2019) presents a relative position encoding scheme that adds additional relative position representation to the key-query computation. Sukhbaatar et al. (2019a) proposes a parameterized linear function over self-attention to learn the optimal attention span in order to extend significantly the maximum context size used in Transformer. To merge FFN to SAN, Sukhbaatar et al. (2019b) proposes a new model that solely consists of attention layers and augments the self-attention layer with persistent memory vectors that play a similar role as the feed-forward layer. As for the collaboration of SAN and FFN, Lu et al. (2019) introduces Macaron layer that split the FFN into two half-steps based on Strang-Marchuk splitting scheme in ODE. For localness modeling, Yang et al. (2018) casts localness modeling as a learnable Gaussian bias according to relative distance to external energy in softmax function as a new self-attention network. Zhao et al. (2019) explores parallel multi-scale representation learning to capture both long-range and short-range language structures with combination of convolution and self-attention. In our work, DMAN, SAN and FFN are unified in Mask Attention Networks, where DMAN is a supplement of SAN and FFN that specializes in localness modeling. Moreover, we investigate different collaboration mechanisms.

6 Conclusion

In this paper, we introduce Mask Attention Networks and reformulate SAN and FFN to point out they are two special cases with static mask in MANs. We analyze the the deficiency of SAN and FFN in localness modeling. Dynamic Mask Attention Network is derived from MANs for better local structure modeling. Considering the different specialities of SAN, FFN, and DMAN, we investigate a sequential layered structure DMAN→SAN→FFN for their collaboration. Compared with original Transformer, our proposed model achieves better performance in neural machine translation and abstract summarization. For future work, we consider adding structure information or external knowledge, e.g., dependency tree, with mask matrices in MANs.

7 Acknowledgement

This work was supported by China National Key R&D Program (No.2018YFC0831105).

References

Karim Ahmed, Nitish Shirish Keskar, and Richard Socher. 2017. Weighted transformer network for machine translation. *arXiv preprint arXiv:1711.02132*.

Emanuele Bugliarello and Naoaki Okazaki. 2019. Improving neural machine translation with parent-scaled self-attention. *arXiv preprint arXiv:1909.03149*.

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc Le, and Ruslan Salakhutdinov. 2019. Transformer-XL: Attentive language models beyond a fixed-length context. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2978–2988, Florence, Italy. Association for Computational Linguistics.

Maosheng Guo, Yu Zhang, and Ting Liu. 2019. Gaussian transformer: a lightweight approach for natural language inference. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6489–6496.

Jie Hao, Xing Wang, Baosong Yang, Longyue Wang, Jinfeng Zhang, and Zhaopeng Tu. 2019. Modeling recurrence for transformer. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1198–1207, Minneapolis, Minnesota. Association for Computational Linguistics.

Chin-Yew Lin and Eduard Hovy. 2003. Automatic evaluation of summaries using N-gram co-occurrence statistics. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pages 71–78. Association for Computational Linguistics.

Yiping Lu, Zhuohan Li, Di He, Zhiqing Sun, Bin Dong, Tao Qin, Liwei Wang, and Tie-Yan Liu. 2019. Understanding and improving transformer from a multi-particle dynamic system point of view. *arXiv preprint arXiv:1906.02762*.

Xutai Ma, Juan Miguel Pino, James Cross, Liezl Puzon, and Jiatao Gu. 2020. Monotonic multihead attention. In *International Conference on Learning Representations*.

Ramesh Nallapati, Bowen Zhou, Cicero dos Santos, Çağlar Gulçehre, and Bing Xiang. 2016. Abstractive text summarization using sequence-to-sequence RNNs and beyond. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pages 280–290, Berlin, Germany. Association for Computational Linguistics.

Myle Ott, Sergey Edunov, David Grangier, and Michael Auli. 2018. Scaling neural machine translation. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 1–9, Brussels, Belgium. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics.

Romain Paulus, Caiming Xiong, and Richard Socher. 2018. A deep reinforced model for abstractive summarization. In *International Conference on Learning Representations*.

Alexander M. Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 379–389, Lisbon, Portugal. Association for Computational Linguistics.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1083, Vancouver, Canada. Association for Computational Linguistics.

Peter Shaw, Jakob Uszkoreit, and Ashish Vaswani. 2018. Self-attention with relative position representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 464–468, New Orleans, Louisiana. Association for Computational Linguistics.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2019. Mass: Masked sequence to sequence pre-training for language generation. In *International Conference on Machine Learning*, pages 5926–5936.

Sainbayar Sukhbaatar, Edouard Grave, Piotr Bojanowski, and Armand Joulin. 2019a. Adaptive attention span in transformers. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 331–335, Florence, Italy. Association for Computational Linguistics.

Sainbayar Sukhbaatar, Edouard Grave, Guillaume Lample, Herve Jegou, and Armand Joulin. 2019b. Augmenting self-attention with persistent memory. *arXiv preprint arXiv:1907.01470*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 5998–6008. Curran Associates, Inc.

Felix Wu, Angela Fan, Alexei Baevski, Yann Dauphin, and Michael Auli. 2019. Pay less attention with lightweight and dynamic convolutions. In *International Conference on Learning Representations*.

Baosong Yang, Jian Li, Derek F Wong, Lidia S Chao, Xing Wang, and Zhaopeng Tu. 2019a. Context-aware self-attention networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 387–394.

Baosong Yang, Zhaopeng Tu, Derek F. Wong, Fandong Meng, Lidia S. Chao, and Tong Zhang. 2018. Modeling localness for self-attention networks. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4449–4458, Brussels, Belgium. Association for Computational Linguistics.

Baosong Yang, Longyue Wang, Derek F. Wong, Lidia S. Chao, and Zhaopeng Tu. 2019b. Convolutional self-attention networks. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4040–4045, Minneapolis, Minnesota. Association for Computational Linguistics.

Haoyu Zhang, Jingjing Cai, Jianjun Xu, and Ji Wang. 2019. Pretraining-based natural language generation for text summarization. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 789–797, Hong Kong, China. Association for Computational Linguistics.

Guangxiang Zhao, Xu Sun, Jingjing Xu, Zhiyuan Zhang, and Liangchen Luo. 2019. MUSE: Parallel multi-scale attention for sequence to sequence learning. *arXiv preprint arXiv:1911.09483*.