Multi-Granularity Self-Attention for Neural Machine Translation

Jie Hao[*]
Florida State University
haoj8711@gmail.com

Xing Wang
Tencent AI Lab
brightxwang@tencent.com

Shuming Shi
Tencent AI Lab
shumingshi@tencent.com

Jinfeng Zhang
Florida State University
jinfeng@stat.fsu.edu

Zhaopeng Tu
Tencent AI Lab
zptu@tencent.com

Abstract

Current state-of-the-art neural machine translation (NMT) uses a deep multi-head self-attention network with no explicit phrase information. However, prior work on statistical machine translation has shown that extending the basic translation unit from words to phrases has produced substantial improvements, suggesting the possibility of improving NMT performance from explicit modeling of phrases. In this work, we present *multi-granularity self-attention* (MG-SA): a neural network that combines multi-head self-attention and phrase modeling. Specifically, we train several attention heads to attend to phrases in either n-gram or syntactic formalism. Moreover, we exploit interactions among phrases to enhance the strength of structure modeling – a commonly-cited weakness of self-attention. Experimental results on WMT14 English-to-German and NIST Chinese-to-English translation tasks show the proposed approach consistently improves performance. Targeted linguistic analysis reveals that MG-SA indeed captures useful phrase information at various levels of granularities.

1 Introduction

Recently, TRANSFORMER (Vaswani et al., 2017), implemented as deep multi-head self-attention networks (SANs), has become the state-of-the-art neural machine translation (NMT) model in recent years. The popularity of SANs lies in its high parallelization in computation, and flexibility in modeling dependencies regardless of distance by explicitly attending to all the signals.

More recently, an in-depth study (Raganato and Tiedemann, 2018) reveals that SANs generally focus on disperse words and ignore continuous phrase patterns, which have proven essential in both statistical machine translation (SMT, Koehn et al., 2003; Chiang, 2005; Liu et al., 2006) and NMT (Eriguchi et al., 2016; Wang et al., 2017; Yang et al., 2018; Zhao et al., 2018).

To alleviate this problem, in this work we propose *multi-granularity self-attention* (MG-SA), which offers SANs the ability to model phrases and meanwhile maintain their simplicity and flexibility. The starting point for our approach is an observation: the power of multiple heads in SANs is not fully exploited. For example, Li et al. (2018) show that different attention heads generally attend to the same positions, and Voita et al. (2019) reveal that only specialized attention heads do the heavy lifting while the rest can be pruned. Accordingly, we spare several attention heads for modeling phrase patterns for SANs.

Specifically, we use two representative types of phrases that are widely-used in SMT models: *n-gram phrases* (Koehn et al., 2003) to use surface of adjacent words, and *syntactic phrases* (Liu et al., 2006) induced from syntactic trees to represent well-formed structural information. We first partition the input sentence into phrase fragments at different levels of granularity. For example, we can split a sentence into 2-grams or 3-grams. Then, we assign an attention head to attend over phrase fragments at each granularity. In this way, MG-SANs provide a lightweight strategy to explicitly model phrase structures. Furthermore, we also model the interactions among phrases to enhance structure modeling, which is one commonly-cited weakness of SANs (Tran et al., 2018; Hao et al., 2019b).

We evaluate the proposed model on two widely-used translation tasks: WMT14 English-to-German and NIST Chinese-to-English. Experimental results demonstrate that our approach consistently improves translation performance over strong TRANSFORMER baseline model (Vaswani et al., 2017) across language pairs, while

[*]Work done when interning at Tencent AI Lab.

Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, pages 887–897, Hong Kong, China, November 3–7, 2019.

speeds marginally decrease. Analysis on multi-granularity label prediction tasks reveals that MG-SA indeed captures and stores the information of different granularity phrases as expected.

2 Background

Multi-Head Self-attention Instead of performing a single attention, Multi-Head Self-attention Networks (MH-SA), which are the defaults setting in TRANSFORMER (Vaswani et al., 2017), project the queries, keys and values into multiple subspaces and performs attention on the projected queries, keys and values in each subspace. In the standard MH-SA, it jointly attends to information from different representation subspaces at different positions. Specifically, MH-SA transform input layer $\mathbf{H} = h_1, ..., h_n \in \mathbb{R}^{n \times d}$ into h-th subspace with different linear projections:

$$\mathbf{Q}^h, \mathbf{K}^h, \mathbf{V}^h = \mathbf{H}\mathbf{W}_Q^h, \mathbf{H}\mathbf{W}_K^h, \mathbf{H}\mathbf{W}_V^h, \quad (1)$$

where $\{\mathbf{Q}^h, \mathbf{K}^h, \mathbf{V}^h\} \in \mathbb{R}^{n \times d_h}$ are respectively the query, key, and value representations of the h-th head, $\{\mathbf{W}_Q^h, \mathbf{W}_K^h, \mathbf{W}_V^h \in \mathbb{R}^{d \times d_h}\}$ denote parameter matrices associated with the h-th head, d and d_h represent the dimensionality of the model and h-th head subspace. Moreover, N attention functions are applied to generate the output states $\{\mathbf{O}^1, ..., \mathbf{O}^N\}$ in parallel, among them:

$$\mathbf{O}^h = \text{ATT}(\mathbf{Q}^h, \mathbf{K}^h)\ \mathbf{V}^h. \quad (2)$$

Finally, the output states are concatenated to produce the final state. Here ATT denotes attention models, which can be implemented as either additive attention or dot-product attention. In this work, we use dot-product attention which is efficient and effective compared with its additive counterpart (Vaswani et al., 2017):

$$\text{ATT}(\mathbf{Q}^h, \mathbf{K}^h) = softmax(\frac{\mathbf{Q}^h \mathbf{K}^{h^T}}{\sqrt{d_h}}), \quad (3)$$

where $\sqrt{d_h}$ is the scaling factor.

Motivation We demonstrate our motivation from two aspects. On the one hand, the conventional MH-SA model the individual word dependencies, in such scenario the query directly attends all words in memory without considering the latent structure of the input sentence. We argue that self-attention can be further improved by taking phrase pattern into account. On the other hand, recent study (Vaswani et al., 2017) implicitly hint that attention heads are underutilized as increasing number of heads from 4 to 8 or even 16 can hardly improve the translation performance. Several attention heads can be further exploited under specific guidance to improve the performance (Strubell et al., 2018). We expect the inductive bias for multi-granularity phrase can further improve the performance of SANs and meanwhile maintain its simplicity and flexibility.

3 Multi-Granularity Self-Attention

We first introduce the framework of the proposed MG-SA. Then we describe the approaches of generating multi-granularity representation on a certain granularity representation. Finally, we introduce the training objective of our model with auxiliary supervision.

3.1 Framework

The proposed MG-SA aims at improving the capability of MH-SA by modeling both word and phrase. We introduce various phrase granularity over the conventional word-level memory to generate phrase level memory.

Specifically, we first transform the input layer $\mathbf{H}$ to a phrase level memory by function F_h in certain attention head:

$$\mathbf{H}_g = F_h(\mathbf{H}), \quad (4)$$

where $\mathbf{H}_g$ is the generated phrase level memory, h denotes the h-th head which is used to generate a certain granularity of phrase memory, and F_h is a representation function with its own trainable parameters. The details for F_h will be described in Section 3.2.

Then we perform attention on phrase level memory $\mathbf{H}^g$:

$$\mathbf{Q}^h, \mathbf{K}^h, \mathbf{V}^h = \mathbf{H}\mathbf{W}_Q^h, \mathbf{H}_g\mathbf{W}_K^h, \mathbf{H}_g\mathbf{W}_V^h \quad (5)$$

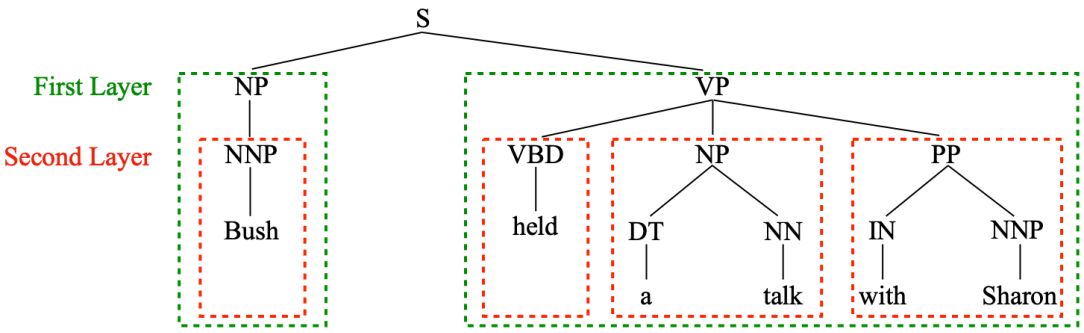
$$\mathbf{O}^h = \text{ATT}(\mathbf{Q}^h, \mathbf{K}^h)\ \mathbf{V}^h, \quad (6)$$

where $\mathbf{Q}^h \in \mathbb{R}^{n \times d_h}, \mathbf{K}^h \in \mathbb{R}^{p \times d_h}, \mathbf{V}^h \in \mathbb{R}^{p \times d_h}$, the p means the length of the key and value vectors which is decided by the granularity of phrase.

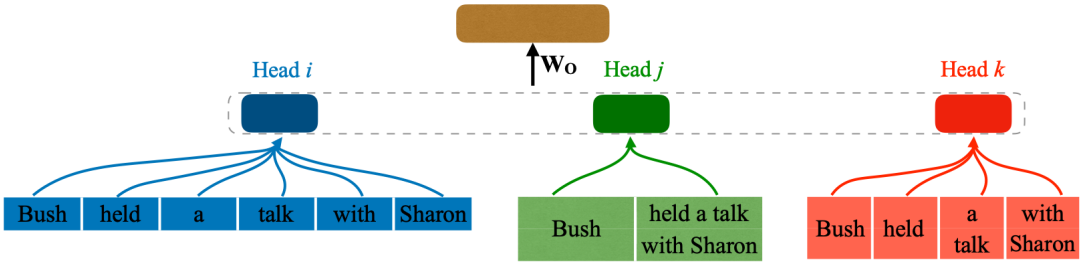
Based on the single head self-attention, the final output of MG-SA can be expressed as follows:

$$\text{MG-SA}(\mathbf{H}) = [\mathbf{O}^1, ..., \mathbf{O}^N], \quad (7)$$

where N denotes the number of heads. One head conducts either conventional word level attention or a certain granularity of phrase attention.



(a) Syntactic phrase partition



(b) *Multi-Granularity Self-Attention* on syntactic phrase partition

Figure 1: Illustration of the proposed MG-SA model for syntactic phrase partition. In this example, we partition the sentence with top two layers in the constituent parse tree and obtain the syntactic phrase partitions ("Bush", "held a talk with Sharon"), ("Bush", "held", "a talk", "with Sharon"). Under the syntactic partition, multi-head attention in MG-SA attends the phrase memory (heads j and k) as well as the conventional word memory (head i). The approach of phrase memory representation is described in Section 3.2. Best viewed in colour.

3.2 Multi-Granularity Representation

As seen in Figure 1, multi-granularity phrases are simultaneously modeled by different heads. To obtain the multi-granularity phrase representation, we first introduce phrase partition and composition strategies. Then, we describe phrase tag supervision and phrase interaction to further enhance the structure modeling on phrase representation.

Phrase Partition Partially inspired by Shen et al. (2018), we split the entire sequence into N-grams without overlaps. Such N-gram phrases are expressed as structurally adjacent and continuous items in the sequence. Formally, let $\mathbf{x} = (x_1, ..., x_T)$ be a sequence, the phrases sequence of $\mathbf{x}$ can be denoted as is $P_\mathbf{x} = (p_1, ..., p_M), M = T/n$, where $p_m = (x_{n\times(m-1)}, ..., x_{n\times m}), 1 \leq m \leq M$, and n denotes the length of the phrase which is a hyper-parameter. Padding is applied to the last phrase if necessary.

In addition, syntactic information has proven helpful in both SMT and NMT. We further introduce a syntactic phrase partition to represent well-formed structural information. Syntactic phrases organize words into nested constituents by using constituent parse tree. To obtain phrases in the view syntax, we break down the nodes at top K layers in the parse tree to capture top K levels of granularity for phrases, as illustrated in Figure 1 (a). Formally, one phrase in a certain layer of the parse tree can be defined as $p_m = (x^1, ..., x^l)$, l is the length of the phrase which is decided by the parse tree. The phrase sequence of the given input $\mathbf{x}$ is $P_\mathbf{x} = (p_1, ..., p_M)$, M is the number of phrase in the sequence.

Composition Strategies Given phrase sequence $P_\mathbf{x} = (p_1, ..., p_M)$ of input sequence, to capture local structure and context dependency inside each phrase and further generate phrase representation Q_M, we adopt phrase composition function to each phrase in the phrase sequence:

$$g_m = \text{COM}(p_m), \tag{8}$$

where COM is the composition function with shared parameters to all phrases, $g_m \in \mathbb{R}^{1\times d_h}$ is the phrase representation after composition. There general choices of composition function are Convolution Neural Networks (CNNs), Recurrent Neural Networks (RNNs) and Self-attention Net-

works (SANs). For CNNs we only apply the Max-pooling layer. For RNNs, we use the last hidden state of Long Short-term Memory Networks (LSTM) as phrase representation. For SANs, we use max pooling vector of the phrase to serve as the query for extracting inside phrase features to generate phrase representation. Then the phrase level memory of the input sequence can be denoted as $\mathbf{G_x} = (g_1, ..., g_M)$.

Phrase Tag Supervision Recent study shows auxiliary supervision on heads of SANs can further improve semantic role labeling performance (Strubell et al., 2018). In this work, we leverage tag information as the auxiliary supervision on syntactic phrase representation. We argue that the proposed framework provide a natural way to incorporate syntactic tag signal of phrase representation. In detail, given phrase level memory $\mathbf{G_x} = (g_1, ..., g_M)$ after phrase composition, we predict the phrase tag of each phrase representation. We extract the node of each phrase in the constituent parsing tree to generate the phrase tag sequence $\mathbf{T_x} = (t_1, ..., t_M)$. t_i denotes the tag for each phrase. For example, "NP" is the tag of the phrase "a talk" in second layer of parse tree, as shown in Figure 1 (a). We use the phrase representation to compute the probability of phrase tags:

$$p_{\theta_i} = softmax(W_t g_i + b_t), i = 1, ..., M, \quad (9)$$

where W_t and b_t are parameters of tag generator. Formally, the phrase tag loss can be written as:

$$\mathcal{L}_{tag} = -\sum_{i=1}^{M} t_i \log p_{\theta_i}(t_i). \quad (10)$$

The loss is equivalent to maximizing the conditional probability of tag sequence $\mathbf{T_x}$ given phrase representation $\mathbf{G_x}$.

Phrase Interaction We introduce phrase interaction approach to better model dependencies between phrase representation. Since recurrence has proven important for capturing structure information (Tran et al., 2018; Hao et al., 2019b), we propose to introduce recurrence to interact phrases and further model latent structure among phrases. Specifically, we apply the recurrence function $\text{REC}(\cdot)$ on the output of phrase composition $\mathbf{G_x} = (g_1, ..., g_M)$ in order to model the latent structure of the phrase sequence.

$$\mathbf{H}_g = \text{REC}(\mathbf{G_x}), \quad (11)$$

where $\mathbf{H}_g$ is the final phrase level memory for the input layer $\mathbf{H}$. One general choice for $\text{REC}(\cdot)$ is Long Short-term Memory Networks (LSTM). Recently, Shen et al. (2018) introduce a new syntax-oriented inductive bias, namely ordered neurons, which enables LSTM models to perform tree-like composition without breaking its sequential form, and propose an advanced LSTM variant – *Ordered Neurons* LSTM (ON-LSTM). Hao et al. (2019a) demonstrate the effectiveness of ON-LSTM on modeling structure in NMT. Accordingly, we further use ON-LSTM for $\text{REC}(\cdot)$, and expect ON-LSTM can capture the latent structure under such syntax-oriented inductive bias between phrases.

Finally, the representation function F_h in Equation 4 of the framework can be summarized by the following components: 1). Phrase partition. 2). Phrase composition. 3). Phrase interaction.

3.3 Training

The training loss for a single training instance $\mathbf{x} = (x_1, ..., x_T), \mathbf{y} = (y_1, ..., y_L)$ is defined as a weighted sum of the negative conditional log likelihood and the phrase tag loss. The total loss function can be written as:

$$\mathcal{L} = -\sum_{i=1}^{L} y_i log P_\omega(y_i) + \lambda \mathcal{L}_{tag}, \quad (12)$$

where λ is the coefficient to balance two loss functions and $\mathcal{L}_{tag}$ follows Equation 10. The hyperparameter λ is empirically set to 0.001 in this work.

4 Experiments

In this section, we conduct experiments and make analysis to answer the following three questions:

Q1. Does the integration of the proposed MG-SA into the state-of-the-art TRANSFORMER improve the translation quality in terms of the BLEU score?

Q2. Does the proposed MG-SA promote the generation of the target phrases?

Q3. Does MG-SA capture more phrase information at the various granularity levels?

In Section 4.1, we demonstrate that integrating the proposed MG-SA into the TRANSFORMER consistently improves the translation quality on both WMT14 English⇒German and

NIST Chinese⇒English (Q1). Further analysis reveals that our approach has stronger ability of capturing the phrase information and promoting the generation of the target phrases (Q2).

In Section 4.2, we conduct experiments on the multi-granularity label prediction tasks (Shi et al., 2016), and investigate the representations of NMT encoders trained on both translation data and the training data of the label prediction tasks. Experimental results show that the proposed MG-SA indeed captures useful phrase information at various levels of granularities in both scenarios (Q3).

4.1 Machine Translation

Implementation Detail We conduct the experiments on the WMT14 English-to-German (En⇒De) and NIST Chinese-to-English (Zh⇒En) translation tasks.

For En⇒De, the training dataset consists of 4.56M sentence pairs. We use the newstest2013 and newstest2014 as development set and test set respectively. For Zh⇒En, the training dataset consists of about 1.25M sentence pairs. We used NIST MT02 dataset as development set, and MT 03-06 datasets as test sets. Byte pair encoding (BPE) toolkit[1] (Sennrich et al., 2016) is used with 32K merge operations. We used case-sensitive NIST BLEU score (Papineni et al., 2002) as the evaluation metric, and bootstrap resampling (Koehn et al., 2003) for statistical significance test. We use the Stanford parser (Klein and Manning, 2003) to parse the sentences and obtain the relevant tags.

We test both *Base* and *Big* models, which differ at hidden size (512 vs. 1024), filter size (2048 vs. 4096) and the number of attention heads (8 vs. 16). All models are trained on eight NVIDIA Tesla P40 GPUs where each is allocated with a batch size of 4096 tokens. We implement the proposed approaches on top of TRANSFORMER (Vaswani et al., 2017) – a state-of-the-art SANs-based model on machine translation, and followed the setting in previous work (Vaswani et al., 2017) to train the models.

We incorporate the proposed model into the encoder. In each of our model variant, we maintain a quarter of heads for vanilla word level self-attention. For N-gram phrase models, we arrange the rest 3 quarters of heads for 2-gram, 3-gram and 4-gram respectively. For syntactic based models, we use the top 3 levels of granularity for phrases generated from constituent parse tree, each granularity of phrase modeled in a quarter of heads. There are many possible ways to implement the general idea of MG-SA. The aim of this paper is not to explore this whole space but simply to show that some fairly straightforward implementations work well.

[1] https://github.com/rsennrich/subword-nmt

Phrase Modeling	# Para.	Speed	BLEU
n/a	88.0M	1.28	27.31
MAX-POOLING	88.0M	1.27	27.56
SANs	90.4M	1.26	**27.69**
LSTM	96.1M	1.14	27.58

Table 1: Evaluation of various phrase composition strategies under N-gram phrase partition. "# Para" denotes the trainable parameter size of each model (M=million), "Speed" denotes the training speed (steps/second).

Encoder Layers	# Para.	Speed	BLEU
$[1-6]$	90.4M	1.26	27.69
$[1-3]$	89.2M	1.27	27.74
$[1]$	88.4M	1.28	**27.83**

Table 2: Evaluation of different layers in the encoder, which are implemented as self-attention with SANs phrase composition under N-gram partition. "1" denotes the bottom layer.

Table 1, 2 and 3 show the results on WMT14 English⇒German translation task with TRANSFORMER-BASE. These results show the evaluation on the impact of different components.

Phrase Composition We investigate the effect of different phrase composition strategies with N-gram phrase partition. As seen in Table 1, all proposed phrase composition methods consistently outperform TRANSFORMER-BASE baseline, validating the importance of introducing multi-granularity phrase in TRANSFORMER. Compared with other two models, SANs achieve best performance with its strong representational powers inside the phrase, while only marginally increase the parameters and decrease the speed. We use SANs phrase composition strategy as the default setting in subsequent experiments.

Encoder Layers Recent works (Shi et al., 2016; Peters et al., 2018) show that different layers in encoder tend to capture different syntax and semantic features. Hence, there may have different needs

#	Model Architecture	# Para.	Speed	BLEU	Δ
1	TRANSFORMER-BASE	88.0M	1.28	27.31	-
2	+ N-gram Phrase	88.4M	1.28	27.83	+0.52
3	+ Syntactic Phrase	88.4M	1.24	28.01	+0.70
4	+ Syntactic Phrase + $\mathcal{L}_{tag}$	88.4M	1.23	28.07	+0.76
5	+ LSTM Interaction	89.5M	1.20	28.14	+0.83
6	+ ON-LSTM Interaction	89.9M	1.19	**28.28**	+0.97

Table 3: Evaluation of phrase partition, tag supervision and interaction strategies.

for modeling phrase structure in each layer. In this experiment, we investigate the question of which layers should be applied with MG-SA. We apply MG-SA on different combination of layers. As shown in Table 2, reducing the applied layers from high-level to low-level consistently increase translation quality in terms of BLEU score as well as the training speed. The results reveal that the bottom layer in encoder, which is directly taking word embedding as input, benefits more from modeling phrase structure. This phenomena verifies it is unnecessary to apply the phrase structure modeling to all layers. Accordingly, we only apply MG-SA in the bottom layer in the following experiments.

Phrase Partition and Tag Supervision As seen in Table 3, syntactic phrase partition (Row 3) improves the model performance over the N-gram phrase partition (Row 2), showing that the syntactic phrase benefits to translation quality. In addition, incorporating tag loss (Row 4) in training stage can further boost the translation performance. This indicates the auxiliary syntax objective is necessary, which is consistent with the results in other NLP task (Strubell et al., 2018). We use syntactic phrase partition with tag supervision as the default setting for subsequent experiments unless otherwise stated.

Phrase Interaction As observed in Table 3, phrase interaction (Row 5-6) consistently improves performance of translation, proving the effectiveness and necessity of enhancing phrase level dependencies on phrase representation. ON-LSTM based interaction (Row 6) outperforms its LSTM counterpart (Row 5). We attribute the improvement of ON-LSTM to the stronger ability to perform syntax-oriented dependencies on phrase representation. We apply ON-LSTM as the default setting for phrase interaction.

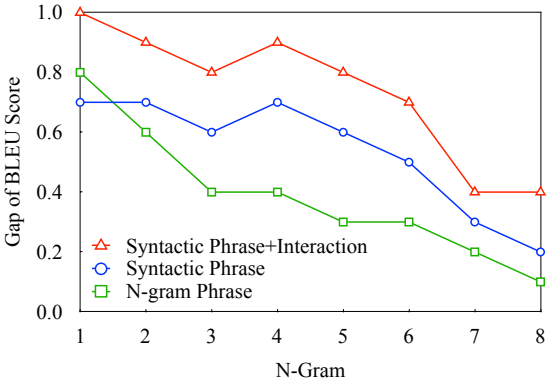


Figure 2: Performance improvement according to N-gram. Y-axis denotes the gap of BLEU score between our models and the baseline.

Main Results Table 4 lists the results on WMT14 En⇒De and NIST Zh⇒En translation tasks. Our baseline models, outperform the reported results on the same data (Vaswani et al., 2017; Zhang et al., 2019), which we believe make the evaluation convincing. As seen, in terms of BLEU score, the proposed MG-SA consistently improves translation performance across language pairs, which demonstrates the effectiveness and universality of the proposed approach.

Phrasal Pattern Evaluation As aforementioned, the proposed MG-SA aims to simultaneously model different granularities of phrases with different heads in SANs. To investigate whether the proposed MG-SA improves the generation of phrases in the output, we calculate the improvement of the proposed models over multiple N-grams, as shown in Figure 2. The results are reported on En⇒De validation set with TRANSFORMER-BASE.

Clearly, the proposed models consistently outperform the baseline model on all N-grams, indicating that the proposed MG-SA has stronger ability of capturing the phrase information and pro-

Architecture	En⇒De		Zh⇒En					
	# Para.	BLEU	# Para.	MT03	MT04	MT05	MT06	Avg
Existing NMT systems								
Vaswani et al. (2017)	213M	28.4	n/a	n/a	n/a	n/a	n/a	n/a
Zhang et al. (2019)	n/a	n/a	n/a	40.45	42.76	40.09	39.67	40.74
Our NMT systems								
TRANSFORMER-BASE	88.0M	27.31	73.4M	41.88	44.48	42.21	41.93	42.60
+MG-SA	89.9M	28.28$^{\Uparrow}$	75.3M	43.98$^{\Uparrow}$	45.60$^{\Uparrow}$	44.28$^{\Uparrow}$	44.00$^{\Uparrow}$	44.46
TRANSFORMER-BIG	264.1M	28.58	234.8M	45.30	46.49	45.21	44.87	45.47
+MG-SA	271.5M	29.01$^{\uparrow}$	242.2M	45.76$^{\uparrow}$	46.81$^{\uparrow}$	45.77$^{\uparrow}$	46.48$^{\Uparrow}$	46.21

Table 4: Comparing with the existing NMT systems on WMT14 En⇒De and NIST Zh⇒En test sets. "$\uparrow$ / $\Uparrow$": significant over the conventional self-attention counterpart ($p < 0.05/0.01$), tested by bootstrap resampling. "MG-SA" denotes "Syntactic Phrase + $\mathcal{L}_{tag}$ + ON-LSTM Interaction" in Table 3.

moting the generation of the target phrases. Concerning the variations of proposed models, two syntactic phrase models outperforms the N-gram phrase model on larger phrases (i.e. 4-8 grams). We attribute this to the fact that more syntactic information is beneficial for the translation performance. This is also consistent with the strengths of phrase-based and syntax-based SMT models.

Visualization of Attention In order to evaluate whether the proposed model is able to capture phrase patterns or not, we visualize the attention layers in the encoder [2]. As shown in Fig. 3, the vanilla model prefers to pay attention to the previous and next word and the end of the sentence, which is consistent with previous findings in Raganato and Tiedemann (2018). The proposed MG-SA successfully focuses on the phrases: 1) "三峡 工程", the 4th and the 5th rows in Fig. 3(b), its English translation is 'the Three Gorges Project'; 2) "首要 任务", the 7th and 8th rows in Fig. 3(b), its English translation is 'top priority'. By visualizing the attention distributions, we believe the proposed MG-SA can capture phrase patterns to improve the translation performance.

4.2 Multi-Granularity Phrases Evaluation

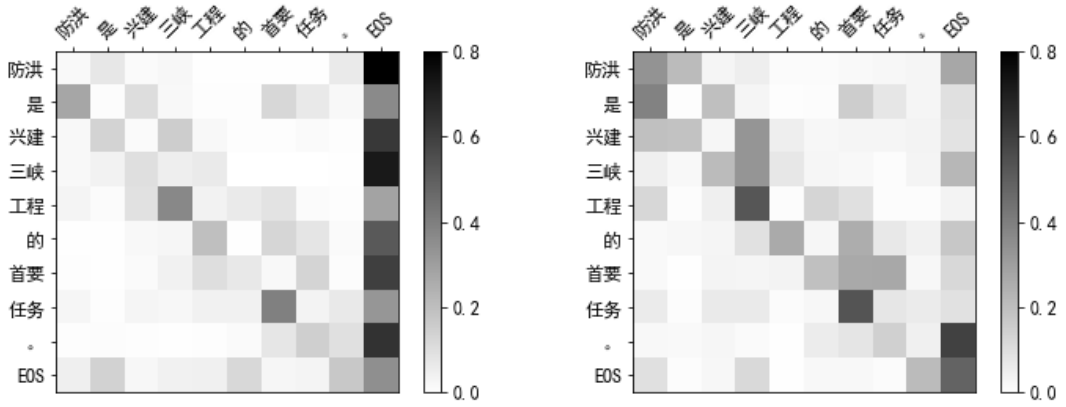
In this section, we conduct multi-granularity label prediction tasks to the proposed models in terms of whether the proposed model is effective as expected to capture different levels of granularity phrase information of sentences. We analyze the impact of multi-granularity self-attention based on two sets of experiments. The first set of experiments are probing the pre-trained NMT encoders, which aims to evaluate the linguistics knowledge embedded in the NMT encoder output in the machine translation section. Furthermore, to test the ability of the MG-SA itself, we conduct the second set of experiments, which are on the same tasks using encoder models trained from scratch.

Tasks Shi et al. (2016) propose 5 tasks to predict various granularity syntactic labels of from sentence to word in order to investigate whether an encoder can learn syntax information. These labels are: "Voice": active or passive, "Tense": past or non-past of main-clause verb, "TSS": top-level syntactic sequence of constituent tree, and two word-level syntactic label tasks, "SPC": the smallest phrase constituent that above each word, "POS": Part-of-Speech tags for each words. The tasks for predicting larger labels require models to capture and record larger granularity of phrase information of sentences (Shi et al., 2016). We conduct these tasks to study whether the proposed MG-SA benefits the multi-granularity phrase modeling to produce more useful and informative representation.

Data and Models We extracted the sentences from the Toronto Book Corpus (Zhu et al., 2015).We sample and pre-process 120k sentences for each task following Conneau et al. (2018). By instruction of Shi et al. (2016), we label these sentences for each task. The train/valid/test dataset ratios are set to 10/1/1.

For pre-trained NMT encoders, we use the pre-trained encoders of model variations in Table 3 followed by a MLP classifier, which are used to

[2]Since the attention weights of MG-SA cannot be visualized at word level, we visualize the attention weights in the subsequent layer after MH-SA and MG-SA.



(a) Vanilla Multi-Head Self-Attention

(b) Multi-Granularity Self-Attention

Figure 3: Visualization of attention examples of the same input sentence: (a) and (b) are produced by the vanilla multi-head self-attention and the proposed MG-SA models, respectively. Each row is the attention distribution over all the source tokens. The attention layer has 16 attention heads, and the attention weights in each row are the average of all the heads.

#	Model	Label Granularity: Large → Small					
		Voice	Tense	TSS	SPC	POS	Avg
Pre-Trained NMT Encoder							
1	BASE	73.38	73.73	72.72	92.81	93.73	81.27
2	N-Gram Phrase	73.06	72.83	72.11	96.42	96.34	82.15
3	Syntactic Phrase	73.37	73.62	75.60	96.72	96.68	83.19
4	Syntactic Phrase + Interaction	73.20	74.78	75.24	96.78	96.56	83.31
Train From Scratch							
5	BASE	83.46	85.39	83.44	96.35	96.12	88.95
6	N-Gram Phrase	83.55	85.62	85.21	96.23	96.17	89.36
7	Syntactic Phrase	84.70	87.52	97.42	96.95	96.24	92.57
8	Syntactic Phrase + Interaction	86.45	87.65	99.07	96.99	96.40	93.31

Table 5: Accuracies on multi-granularity label prediction tasks. "Pre-Trained NMT Encoder" denotes using the pre-trained NMT encoders of model variations in Table 3. "Train From Scratch" denotes using three encoder layers with proposed MG-SA variants, which are trained from scratch. For syntactic phrase based models, we only apply syntactic boundary of phrases and do not use any tag supervision for fair comparison.

carry out probing tasks.

For models trained from scratch, each of our model consists of 3 encoding layers followed by a MLP classifier. For each encoding layer, we employ a multi-head self-attention block and a feed-forward block as in TRANSFORMER, which have shown significant performance on several NLP tasks (Devlin et al., 2019). The difference between the compared models merely lies in the self-attention mechanism: "BASE" denotes standard MH-SA, "N-Gram Phrase" and "Syntactic Phrase" are the proposed MG-SA under N-gram phrase and syntactic phrase partition, and "Syntactic Phrase + Interaction" denotes MG-SA with phrase interaction by using ON-LSTM. We use same assignments of heads for multi-granularity phrases as machine translation task for all model variants.

Results Analysis Table 5 lists the prediction accuracies of five syntactic labels on test. Several observations can be made here. 1). Comparing the two set of experiments, the experimental results from models trained from scratch consistently outperform the results from NMT encoder probing on all tasks. 2). The models with syntactic information (Rows 3-4, 7-8) significantly perform better than those models without incorporating syntactic information (Rows 1-2, 5-6). 3). For NMT prob-

ing, the proposed models outperform the baseline model especially on relative small granularity of phrases information, such as 'SPC' and 'POS' tasks. 4). If trained from scratch, the proposed models achieve more improvements on predicting larger granularities of labels, such as 'TSS', 'Tense' and 'Voice' tasks, which require models to record larger phrase of sentences (Shi et al., 2016). The results show that the applicability of the proposed MG-SA is not limited to machine translation, but also on monolingual tasks.

5 Related Works

Phrase Modeling for NMT Several works have proven that the introduction of phrase modeling in NMT can obtain promising improvement on translation quality. Tree-based encoders, which explicitly take the constituent tree (Eriguchi et al., 2016) or dependency tree (Bastings et al., 2017) into consideration, are proposed to produce tree-based phrase representations. The difference of our work from these studies is that they adopt the RNN-based encoder to form the tree-based encoder while we explicitly introduce the phrase structure into the the state-of-the-art multi-layer multi-head SANs-based encoder, which we believe is more challenging.

Another thread of work is to implicitly promote the generation of phrase-aware representation, such as the integration of external phrase boundary (Wang et al., 2017; Nguyen and Joty, 2018; Li et al., 2019b), prior attention bias (Yang et al., 2018, 2019; Guo et al., 2019). Our work differs at that we explicitly model phrase patterns at different granularities, which is then attended by different attention heads.

Multi Granularity Representation Multi-granularity representation, which is proposed to make full use of subunit composition at different levels of granularity, has been explored in various NLP tasks, such as paraphrase identification (Yin and Schütze, 2015), Chinese word embedding learning (Yin et al., 2016), universal sentence encoding (Wu et al., 2018) and machine translation (Nguyen and Joty, 2018; Li et al., 2019b). The major difference between our work and Nguyen and Joty (2018); Li et al. (2019b) lies in that we successfully introduce syntactic information into our multi-granularity representation. Furthermore, it is not well measured how much phrase information are stored in multi-granularity representation. We conduct the multi-granularity label prediction tasks and empirically verify that the phrase information is embedded in the multi-granularity representation.

Multi-Head Attention Multi-head attention mechanism has shown its effectiveness in machine translation (Vaswani et al., 2017) and generative dialog (Tao et al., 2018) systems. Recent studies shows that the modeling ability of multi-head attention has not been completely developed. Several specific guidance cues of different heads without breaking the vanilla multi-head attention mechanism can further boost the performance, e.g., disagreement regularization (Li et al., 2018; Tao et al., 2018), information aggregation (Li et al., 2019a), and functional specialization (Fan et al., 2019) on attention heads, the combination of multi-head attention with multi-task learning (Strubell et al., 2018). Our work demonstrates that multi-head attention also benefits from the integration of the phrase information.

6 Conclusion

In this paper, we propose multi-granularity self-attention model, a novel attention mechanism to simultaneously attend different granularity phrase. We study effective phrase representation for N-gram phrase and syntactic phrase, and find that a syntactic phrase based mechanism obtains the best result due to effectively incorporating rich syntactic information. To evaluate the effectiveness of the proposed model, we conduct experiments on widely-used WMT14 En⇒De and NIST Zh⇒En datasets. Experimental results on two language pairs show that the proposed model achieve significant improvements over the baseline TRANSFORMER. Targeted multi-granularity phrases evaluation shows that our model indeed capture useful phrase information.

As our approach is not limited to specific tasks, it is interesting to validate the proposed model in other tasks, such as reading comprehension, language inference, and sentence classification.

Acknowledgments

J.Z. was supported by the National Institute of General Medical Sciences of the National Institute of Health under award number R01GM126558. We thank the anonymous reviewers for their insightful comments.

References

Joost Bastings, Ivan Titov, Wilker Aziz, Diego Marcheggiani, and Khalil Simaan. 2017. Graph convolutional encoders for syntax-aware neural machine translation. In *EMNLP*.

David Chiang. 2005. A hierarchical phrase-based model for statistical machine translation. In *ACL*.

Alexis Conneau, German Kruszewski, Guillaume Lample, Loïc Barrault, and Marco Baroni. 2018. What you can cram into a single $\&!\#*$ vector: Probing sentence embeddings for linguistic properties. In *ACL*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*.

Akiko Eriguchi, Kazuma Hashimoto, and Yoshimasa Tsuruoka. 2016. Tree-to-sequence attentional neural machine translation. In *ACL*.

Angela Fan, Mike Lewis, and Yann Dauphin. 2019. Strategies for structuring story generation. In *ACL*.

Maosheng Guo, Yu Zhang, and Ting Liu. 2019. Gaussian transformer: a lightweight approach for natural language inference. In *AAAI*.

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

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

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

Philipp Koehn, Franz Josef Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *NAACL*.

Jian Li, Zhaopeng Tu, Baosong Yang, Michael R. Lyu, and Tong Zhang. 2018. Multi-Head Attention with Disagreement Regularization. In *EMNLP*.

Jian Li, Baosong Yang, Zi-Yi Dou, Xing Wang, Michael R Lyu, and Zhaopeng Tu. 2019a. Information aggregation for multi-head attention with routing-by-agreement. In *NAACL*.

Yang Li, Lukasz Kaiser, Samy Bengio, and Si Si. 2019b. Area attention. In *ICML*.

Yang Liu, Qun Liu, and Shouxun Lin. 2006. Tree-to-string alignment template for statistical machine translation. In *ACL*.

Phi Xuan Nguyen and Shafiq Joty. 2018. Phrase-based attentions. *arXiv preprint arXiv:1810.03444*.

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

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *NAACL*.

Alessandro Raganato and Jörg Tiedemann. 2018. An analysis of encoder representations in transformer-based machine translation. In *EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *ACL*.

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

Xing Shi, Inkit Padhi, and Kevin Knight. 2016. Does string-based neural mt learn source syntax. In *EMNLP*.

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

Chongyang Tao, Shen Gao, Mingyue Shang, Wei Wu, Dongyan Zhao, and Rui Yan. 2018. Get the point of my utterance! learning towards effective responses with multi-head attention mechanism. In *IJCAI*.

Ke Tran, Arianna Bisazza, and Christof Monz. 2018. The importance of being recurrent for modeling hierarchical structure. In *EMNLP*.

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

Elena Voita, David Talbot, Fedor Moiseev, Rico Sennrich, and Ivan Titov. 2019. Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned. In *ACL*.

Xing Wang, Zhaopeng Tu, Deyi Xiong, and Min Zhang. 2017. Translating phrases in neural machine translation. In *EMNLP*.

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

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

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

Rongchao Yin, Quan Wang, Peng Li, Rui Li, and Bin Wang. 2016. Multi-granularity chinese word embedding. In *EMNLP*.

Wenpeng Yin and Hinrich Schütze. 2015. Convolutional neural network for paraphrase identification. In *NAACL*.

Meishan Zhang, Zhenghua Li, Guohong Fu, and Min Zhang. 2019. Syntax-enhanced neural machine translation with syntax-aware word representations. In *NAACL*.

Yang Zhao, Yining Wang, Jiajun Zhang, and Chengqing Zong. 2018. Phrase table as recommendation memory for neural machine translation. In *IJCAI*.

Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *ICCV*.