Context-Aware Self-Attention Networks

Baosong Yang[1] **Jian Li**[2] **Derek F. Wong**[1] **Lidia S. Chao**[1] **Xing Wang**[3] **Zhaopeng Tu**[3*]
[1]NLP[2]CT Lab, Department of Computer and Information Science, University of Macau
nlp2ct.baosong@gmail.com, {derekfw,lidiasc}@umac.mo
[2]The Chinese University of Hong Kong
jianli@cse.cuhk.edu.hk
[3]Tencent AI Lab
{brightxwang,zptu}@tencent.com

Abstract

Self-attention model has shown its flexibility in parallel computation and the effectiveness on modeling both long- and short-term dependencies. However, it calculates the dependencies between representations without considering the contextual information, which has proven useful for modeling dependencies among neural representations in various natural language tasks. In this work, we focus on improving self-attention networks through capturing the richness of context. To maintain the simplicity and flexibility of the self-attention networks, we propose to contextualize the transformations of the query and key layers, which are used to calculate the relevance between elements. Specifically, we leverage the internal representations that embed both global and deep contexts, thus avoid relying on external resources. Experimental results on WMT14 English⇒German and WMT17 Chinese⇒English translation tasks demonstrate the effectiveness and universality of the proposed methods. Furthermore, we conducted extensive analyses to quantify how the context vectors participate in the self-attention model.

Introduction

Self-attention networks (SANs) (Lin et al. 2017) have shown promising empirical results in various NLP tasks, such as machine translation (Vaswani et al. 2017), nature language inference (Shen et al. 2018), and acoustic modeling (Sperber et al. 2018). One strong point of SANs is the strength of capturing long-range dependencies by explicitly attending to all the signals, which allows the model to build a direct relation with another long-distance representation.

However, SANs treat the input sequence as a bag-of-word tokens and each token individually performs attention over the bag-of-word tokens. Consequently, the contextual information is not taken into account in the calculation of dependencies between elements. Several researchers have shown that contextual information can enhance the ability of modeling dependencies among neural representations, especially for the attention models. For example, Tu et al. (2017) and Zhang et al. (2017) respectively enhanced the query and memory of a standard attention model (Bahdanau, Cho, and Bengio 2015) with internal contextual representations. Wang et al. (2017) and Voita et al. (2018) enhanced the two components with external contextual representations that summarizes previous source sentences.

In this work, we propose to strengthen SANs through capturing the richness of context, and meanwhile maintain their simplicity and flexibility. To this end, we employ the internal representations as context vectors, thus avoid relying on external resources, e.g. the embeddings of previous sentences. Specifically, we contextualize the transformations from the input layer to the query and key layers, which are used to calculate the relevance between elements. We exploit several strategies for the contextualization, including: 1) *global context* that represents the global information of a sequence; 2) *deep context* that embeds syntactic and semantic information summarized by multiple-layer representations; and 3) *deep-global context* that combines information of both the above two context vectors.

Some researchers may doubt that, for a multi-layer self-attentive model (e.g. TRANSFORMER (Vaswani et al. 2017)), each input state has summarized the global information from its lower layer through the weighted sum operation. Our study dispels the doubt by showing that such summarization does not fully captures the richness of contextual information. We conducted experiments on two widely-used WMT14 English⇒German and WMT17 Chinese⇒English translation tasks. The proposed approach consistently improves translation performance over the strong TRANSFORMER baseline, while only marginally decreases the speed. Extensive analyses reveal that there exists separate requirement of contextual information for different representations, e.g., the representation of the function words distinctly require more contexts than that of the content words.

Background

Recently, as a variant of attention model, self-attention networks (Lin et al. 2017) have attracted a lot of interests due to their flexibility in parallel computation and modeling both long- and short-term dependencies. SANs calculate attention weights between each pair of tokens in a single se-

*Zhaopeng Tu is the corresponding author. Work was done when Baosong Yang and Jian Li were interning at Tencent AI Lab.

quence, thus can capture long-range dependency more directly than their RNN counterpart.

Formally, given an input layer $\mathbf{H} = \{h_1, \ldots, h_n\}$, the hidden states in the output layer are constructed by attending to the states of input layer. Specifically, the input layer $\mathbf{H} \in \mathbb{R}^{n \times d}$ is first transformed into queries $\mathbf{Q} \in \mathbb{R}^{n \times d}$, keys $\mathbf{K} \in \mathbb{R}^{n \times d}$, and values $\mathbf{V} \in \mathbb{R}^{n \times d}$:

$$\begin{bmatrix} \mathbf{Q} \\ \mathbf{K} \\ \mathbf{V} \end{bmatrix} = \mathbf{H} \begin{bmatrix} \mathbf{W}_Q \\ \mathbf{W}_K \\ \mathbf{W}_V \end{bmatrix}, \quad (1)$$

where $\{\mathbf{W}_Q, \mathbf{W}_K, \mathbf{W}_V\} \in \mathbb{R}^{d \times d}$ are trainable parameter matrices with d being the dimensionality of input states. The output layer $\mathbf{O} \in \mathbb{R}^{n \times d}$ is constructed by

$$\mathbf{O} \quad = \quad \textsc{Att}(\mathbf{Q}, \mathbf{K})\ \mathbf{V}, \quad (2)$$

where $\textsc{Att}(\cdot)$ is an attention model, which can be implemented as either additive attention (Bahdanau, Cho, and Bengio 2015) or dot-product attention (Luong, Pham, and Manning 2015). In this work, we use the latter, which achieves similar performance with its additive counterpart while is much faster and more space-efficient in practice (Vaswani et al. 2017):

$$\textsc{Att}(\mathbf{Q}, \mathbf{K}) \quad = \quad softmax(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d}}), \quad (3)$$

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

Motivation

The strength of SANs lies in the ability of directly capturing dependencies between layer hidden states (Vaswani et al. 2017). However, the calculation of similarity between query and key in the self-attention model is merely controlled by two trained parameter matrices:

$$\mathbf{Q}\mathbf{K}^T = (\mathbf{H}\mathbf{W}_Q)(\mathbf{H}\mathbf{W}_K)^T = \mathbf{H}(\mathbf{W}_Q\mathbf{W}_K^T)\mathbf{H}^T, \quad (4)$$

which miss the opportunity to take advantage of useful contexts. For example, as seen in Figure 1 (a), self-attention model individually calculate the relevance between the word pair ("talk" and "Sharon") without considering the contextual information. We expect that modeling context can further improve the performance of SANs.

Approach

In this study, we propose a context-aware self-attention model. We describe several types of context vectors and introduce how to incorporate the context vectors into the SAN-based sequence-to-sequence models.

Context-Aware Self-Attention Model

In order to alleviate the lack of contextual information and to maintain the flexibility on parallel computation for self-attention networks, we propose to contextualize the transformations from the input layer $\mathbf{H}$ to the query and key layers. Specifically, we follow Shaw, Uszkoreit, and Vaswani (2018) to propagate contextual information to transformation using addition, which avoids significantly increasing computation:

$$\begin{bmatrix} \widehat{\mathbf{Q}} \\ \widehat{\mathbf{K}} \end{bmatrix} = (1 - \begin{bmatrix} \lambda_Q \\ \lambda_K \end{bmatrix}) \begin{bmatrix} \mathbf{Q} \\ \mathbf{K} \end{bmatrix} + \begin{bmatrix} \lambda_Q \\ \lambda_K \end{bmatrix} (\mathbf{C} \begin{bmatrix} \mathbf{U}_Q \\ \mathbf{U}_K \end{bmatrix}), \quad (5)$$

where $\mathbf{C} \in \mathbb{R}^{n \times d_c}$ is the context vector, and $\{\mathbf{U}_Q, \mathbf{U}_K\} \in \mathbb{R}^{d_c \times d}$ are the associated trainable parameter matrices. To effectively leverage these hierarchical representations, $\{\lambda_Q, \lambda_K\} \in \mathbb{R}^{n \times 1}$ are assigned to weight the expected importance of the context representations.

Britz et al. (2017) and Vaswani et al. (2017) noted that a large magnitude of Q and K may push the softmax function (Equation 3) into regions where it has extremely small gradients. To counteract this effect, $\{\lambda_Q, \lambda_K\}$ can also be treated as factors to regulate the magnitude of $\widehat{Q}$ and $\widehat{K}$.[1] Inspired by the prior studies on multi-modal networks (Xu et al. 2015; Calixto, Liu, and Campbell 2017; Yang et al. 2017), we assign a gating scalar to learn the factors:

$$\begin{bmatrix} \lambda_Q \\ \lambda_K \end{bmatrix} = \sigma(\begin{bmatrix} \mathbf{Q} \\ \mathbf{K} \end{bmatrix} \begin{bmatrix} \mathbf{V}_Q^H \\ \mathbf{V}_K^H \end{bmatrix} + \mathbf{C} \begin{bmatrix} \mathbf{U}_Q \\ \mathbf{U}_K \end{bmatrix} \begin{bmatrix} \mathbf{V}_Q^C \\ \mathbf{V}_K^C \end{bmatrix}), \quad (6)$$

where $\{\mathbf{V}_Q^H, \mathbf{V}_K^H\} \in \mathbb{R}^{d \times 1}$ and $\{\mathbf{V}_Q^C, \mathbf{V}_K^C\} \in \mathbb{R}^{d_c \times 1}$ are trainable parameters. $\sigma(\cdot)$ denotes the logistic sigmoid function. The gating scalar enables the model to explicitly quantify how much each representation and the context vector contribute to the prediction of attention weight.

Accordingly, the output representation is constructed based on the contextualized query and key representations:

$$\mathbf{O} \quad = \quad \textsc{Att}(\widehat{\mathbf{Q}}, \widehat{\mathbf{K}})\ \mathbf{V}. \quad (7)$$

As seen, the proposed approach does not require specific attention functions, thus is applicable to all attention models.

Choices of Context Vectors

One principle of our approach is to maintain the simplicity and flexibility of the self-attention model. With this in mind, we employ the internal states as context vectors, thus avoid relying on external resources. Specifically, we exploit several types of context vectors, which can either be used individually or combined together.

Global Context Global context is a function of the entire input layer, which represents the global meaning of a sequence. In this work, we use mean operation to summarize the representations of the input layer, which is commonly used in Seq2Seq models (Cho et al. 2014):

$$\mathbf{c} = \overline{\mathbf{H}} \qquad \in \mathbb{R}^d \quad (8)$$

[1] We conduct experiments on the effectiveness of the factors. The experimental results reveal that without $\{\lambda_Q, \lambda_K\}$, there is a big drop (-5.23 BLEU) on the final translation qualities. This indicates that the large magnitude of $\mathbf{Q}$ and $\mathbf{K}$ exactly hinder the convergence of SANs, and the trainable linear projections (Equation 5) insufficiently learn to regular the magnitude.

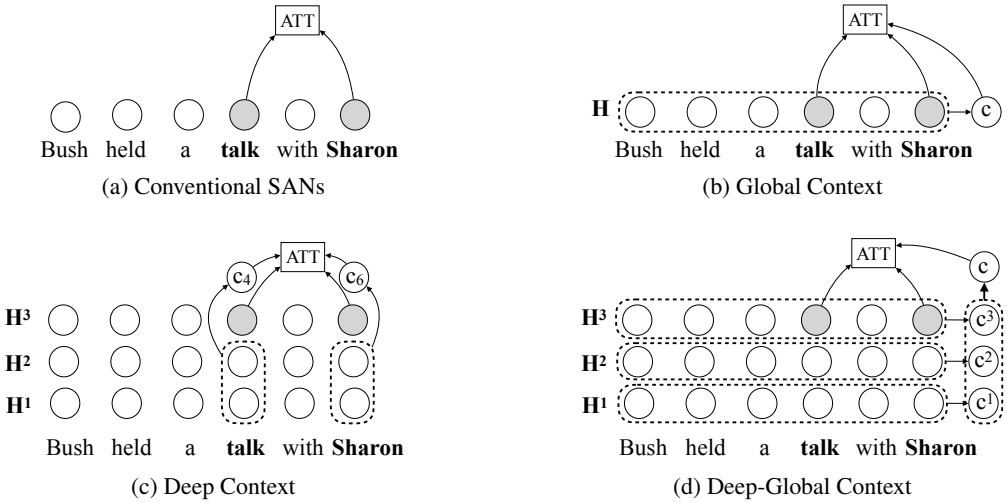

Figure 1: Illustration of the proposed models. As seen, the conventional self-attention networks (a) individually calculate the attention weight of two items ("talk" and "sharon") without covering the contextual information. The global context strategy (b) and the deep-context strategy (c) capture the global meaning of a sentence and the syntactic information from the lower layers respectively. Figure (d) shows a deep-global context model which summarizes the historical global context vectors.

Note that the global context is a vector instead of a matrix, which is shared across layer states.

Intuitively, the global context can be regarded as an instance-specific bias (Hariharan et al. 2015) for the self-attention model, which is expected to complement the unified parameters $\{\mathbf{W}_Q, \mathbf{W}_K\}$ shared across instances in the training data. The pair-wised features in conjunction with the global features produce an instance-specific prior which has been shown its effectiveness on several recognition and detection tasks (Hariharan et al. 2015; Gkioxari, Girshick, and Malik 2015; Zhu, Porikli, and Li 2016).

Deep Context Deep context is a function of the internal layers stacked below the current input layer. Advanced neural models are generally implemented as multiple layers, which are able to capture different types of syntactic and semantic information (Shi, Padhi, and Knight 2016; Peters et al. 2018; Anastasopoulos and Chiang 2018). For example, Peters et al. (2018) show that higher-level layer states capture the context-dependent aspects of word meaning while lower-level states model the aspects of syntax, and simultaneously exposing all of these signals is highly beneficial.

Formally, let $\mathbf{H}^l$ be the current input layer at the l-th level, the deep context is a concatenation of the layers underneath the input layer:

$$\mathbf{C} = [\mathbf{H}^1, \ldots, \mathbf{H}^{l-1}] \qquad \in \mathbb{R}^{n \times (l-1)d} \tag{9}$$

The deep context enables the self-attention model to fuse different types of syntactic and semantic information captured by different layers.

Note that we employ a dense connection strategy (Huang et al. 2017) instead of linear combination (Peters et al. 2018). We believe the former is a more suitable strategy in this scenario, since the weight matrices $\{\mathbf{U}_Q, \mathbf{U}_K\} \in \mathbb{R}^{(l-1)d \times d}$ in Equation 5 plays the role of combination. Our strategy differs from Peters et al. (2018) at: (1) they use normalized weights while we directly use parameters that could be either positive or negative numbers, which may benefit from more modeling flexibility; (2) they use a scalar that is shared by all elements in the layer states, while we assign a distinct "scalar" to each element. The latter offers a more precise control of the combination by allowing the model to be more expressive than scalars (Tu et al. 2017).

Deep-Global Context Intuitively, we can combine the concepts of global and deep context, and fuse global context across layers:

$$\mathbf{c} = [\mathbf{c}^1, \ldots, \mathbf{c}^l] \qquad \in \mathbb{R}^{ld} \tag{10}$$

where $\mathbf{c}^l$ is the global context of the l-th layer $\mathbf{H}^l$, which is calculated via Equation 8. We expect the deep-global context to provide different levels of linguistic biases, ranging from lexical, through syntactic, to semantic levels.

As seen, the above context vectors embed different types of information, either global or state-wise context, which may be complementary to each other. To exploit the advantages of all of them, an intuitive strategy is to concatenate multiple context vectors to form a new vector, which serves as $\mathbf{C}$ in Equation 5. The proposed approach can be easily integrated into the state-of-the-art SAN-based SEQ2SEQ models (Vaswani et al. 2017), in which both encoder and decoder are composed of a stack of L SAN layers.[2]

[2]For decoder-side SANs, global context vector is a summarization of the forward representations at each decoding step, since the subsequent representations are invisible and thus are masked during training.

#	Model	Applied to	Context Vectors	# Para.	Train	Decode	BLEU
1	BASE	n/a	n/a	88.0M	1.28	1.52	27.31
2	OURS	encoder	*global context*	91.0M	1.26	1.50	27.96
3			*deep-global context*	99.0M	1.25	1.48	28.15
4			*deep context*	95.9M	1.18	1.38	28.01
5			*deep-global context + deep context*	106.9M	1.16	1.36	**28.26**
6		decoder	*deep-global context*	99.0M	1.23	1.44	27.94
7			*deep-global context + deep context*	106.9M	1.15	1.35	28.02
8		both	5 + 7	125.8M	1.04	1.20	28.16

Table 1: Experimental results on WMT14 En⇒De translation task using TRANSFORMER-BASE. "# Para." denotes the trainable parameter size of each model (M = million). "Train" and "Decode" denote the training speed (steps/second) and the decoding speed (sentences/second), respectively.

Experiments

Setup

Following (Vaswani et al. 2017), we evaluated the proposed approach on machine translation tasks. To compare with the results reported by previous SAN-based NMT models (Vaswani et al. 2017; Hassan et al. 2018), we conducted experiments on both English⇒German (En⇒De) and Chinese⇒English (Zh⇒En) translation tasks. For the En⇒De task, we trained on the widely-used WMT14 dataset consisting of about 4.56 million sentence pairs. The models were validated on newstest2013 and examined on newstest2014. For the Zh⇒En task, the models were trained using all of the available parallel corpus from WMT17 dataset, consisting of about 20.62 million sentence pairs. We used newsdev2017 as the development set and newstest2017 as the test set. The English and German sentences were tokenized using the scripts provided in Moses. Then, all tokenized sentences were processed by byte-pair encoding (BPE) to alleviate the Out-of-Vocabulary problem (Sennrich, Haddow, and Birch 2016) with 32K merge operations for both language pairs. We used BLEU score (Papineni et al. 2002) as the evaluation metric.

We evaluated the proposed approaches on our re-implemented TRANSFORMER model (Vaswani et al. 2017). We followed (Vaswani et al. 2017) to set the configurations and reproduced their reported results on the En⇒De task. We tested both the *Base* and *Big* models, which differ at the layer size (512 vs. 1024) and the number of attention heads (8 vs. 16). All the models are trained on eight NVIDIA P40 GPUs, each of which is allocated a batch of 4096 tokens. In consideration of the computation cost, we studied the variations of the *Base* model on En⇒De task, and evaluated the overall performance with the *Big* model on both En⇒De and Zh⇒En translation tasks.

Ablation Study on the Context Vector

In this section, we conducted experiments to evaluate the impact of different types of context vector on the WMT14 En⇒De translation tasks using the TRANSFORMER-BASE. First, we investigated the effect of different context vectors for the encoder-side self-attention networks. Then, we examined whether modeling contextual information on the decoder-side SANs is able to gain consistent improvement. Finally, we checked whether the context-aware model on encoder-side and decoder-side SANs can be complementary to each other. To eliminate the influence of control variables, we conducted the first two ablation studies on encoder-side or decoder-side self-attention networks only.

Applied to Encoder As shown in Table 1, all the proposed context vector strategies consistently improve the model performance over the baseline, validating the importance of modeling contextual information in self-attention networks. Among them, global context (Model #2) and deep context (Model #4) gained comparable improvements. Deep-global context outperforms its global counterpart, showing that the different levels of global linguistic biases benefit to the accuracy of translation. Moreover, we evaluated whether the global and deep manners are complementary to each other. By simply summing them to the final context vectors, the model "deep-global context + deep context" (Model #5) gains further improvements. According to the results, we argue that the two types of context vectors are able to improve the SANs in different aspects.

Applied to Decoder In this group of experiments, we investigated the question of which types of context vector should be applied to the decoder-side self-attention networks. As shown in Table 1, both "deep-global context" (Model #6) and "deep-global context + deep context" (Model #7) consistently improve the SANs. Still, the later outperforms the former one, which is same to the phenomenon appeared in the experiments in terms of encoder. Noted that, Zhang, Xiong, and Su (2018) pointed out that the decoder-side SANs tends to only focus on its nearby representation. However, our improvements show that all the forward (global) and lower layer (deep) representations are still necessary for the decoder-side SANs.

Applied to Both Encoder and Decoder Finally, we integrated the strategies into both the encoder and decoder. As seen, this strategy (Model #8) even slightly harms the translation quality (compare to encoder-side models). We attribute the drop of BLEU score to the fact that the conventional encoder-decoder attention model in TRANSFORMER exploits the top-layer of encoder representations, which already embeds useful contextual information. The context-aware model may benefit more on encoder-side SAN under

System	Architecture	Zh⇒En			En⇒De		
		# Para.	Train	BLEU	# Para.	Train	BLEU
Existing NMT systems							
(Vaswani et al. 2017)	TRANSFORMER-BASE	n/a	n/a	n/a	65M	n/a	27.30
	TRANSFORMER-BIG	n/a	n/a	n/a	213M	n/a	28.40
(Hassan et al. 2018)	TRANSFORMER-BIG	n/a	n/a	24.20	n/a	n/a	n/a
Our NMT systems							
this work	TRANSFORMER-BASE	107.9M	1.21	24.13	88.0M	1.28	27.31
	+ Context-Aware SANs	126.8M	1.10	24.67⇑	106.9M	1.16	28.26⇑
	TRANSFORMER-BIG	303.9M	0.58	24.56	264.1M	0.61	28.58
	+ Context-Aware SANs	379.4M	0.41	25.15⇑	339.6M	0.44	28.89

Table 2: Comparing with the existing NMT systems on WMT17 Zh⇒En and WMT14 En⇒De test sets. "⇑": significant over the conventional self-attention counterpart ($p < 0.01$), tested by bootstrap resampling.

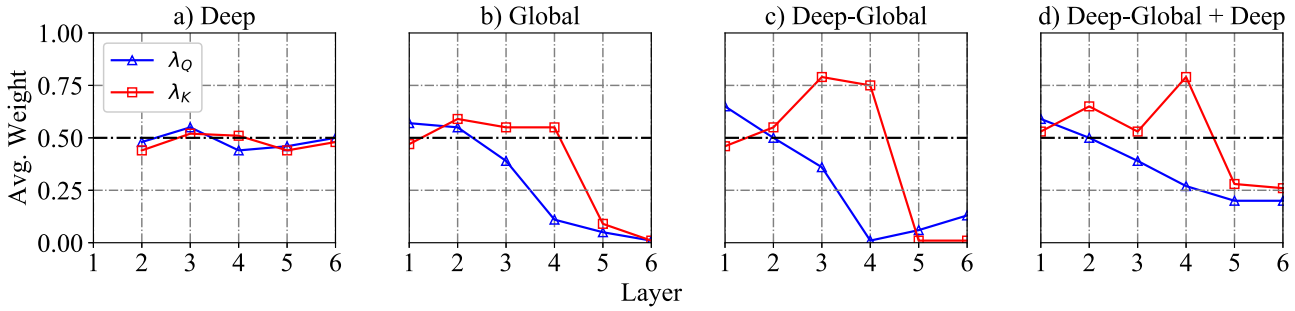

Figure 2: Visualization of the importance of each type of context vector on different layers. The importance is assessed by averaging the scalar factors in Equation 5 over the validation set and distinguished by λ_Q (blue) and λ_K (red).

the architecture of TRANSFORMER.

Unless otherwise stated, considering the training speed, we only applied the context-aware model to the encoder-side SANs in the following experiments, which employs a "deep-global context + deep context " strategy (Row #5 in Table 1).

Main Results

Table 2 lists the results on WMT17 Zh⇒En and WMT14 En⇒De translation tasks. Our baseline models, both TRANSFORMER-BASE and TRANSFORMER-BIG, outperform the reported results on the same data, which makes the evaluation convincing. As seen, modeling contextual information ("Context-Aware SANs") consistently improves the performance across language pairs and model variations, demonstrating the efficiency and universality of the proposed approach. It is encouraging to see that TRANSFORMER-BASE with context-aware model achieves comparable performance with TRANSFORMER-BIG, while the training speed is nearly twice faster and only requires half of parameters.

Analysis

We conducted extensive analyses to better understand our models in terms of their compatibility with self-attention networks. All the results are reported on En⇒De validation set with TRANSFORMER-BASE.

Deep Context vs. Global Context

In this section, we investigated the details of difference between deep context and global context to answer the question: how do they harmonically work with queries and keys in multiple layers?

Stable Necessity of Deep Context As seen in Figure 2 (a), in deep based context-aware models, the weights of scalar factors are consistently close to 0.5, meaning the equivalent importance of the information in the current layer and that of the historical layers. The improvements on the translation quality and the stable necessity jointly verified our claim that the conventional self-attention mechanism is insufficient to fully capture the richness of the context through weighted averaging its input layer. Opportunely utilizing the historical context benefits the performance of SANs.

The Lower Layer, The More Global Context Required Concerning the global-based approaches, obviously, the average weights of global context vectors in Figure 2 (b), (c) and (d) drop at high-level layers. The trends demonstrate that the higher layers require less global information, while the lower layers require more global context. The phenomenon confirms that a single SAN layer has limited ability in learning the global meaning, resulting in the high weights of global context vector in lower layers. However the global contextual information can be gradually accumulated with

the increasing number of layers, this is the reason why the higher layers hardly need the global information. We believe the global context vector is more beneficial to the lower layers on modeling semantic meanings.

Model	Query	Key	Dev
TRANSFORMER-BASE	-	-	25.84
+ Context-Aware	✓	✓	**26.42**
	×	✓	26.36
	✓	×	26.20

Table 3: BLEU scores on the En⇒De validation set with respect to integrate context vector into queries and keys.

Keys Required More Global Information Another common interesting phenomenon appears in all the global-based approaches is that the weights of global context vectors for keys are usually higher than that for queries, especially in the mid-level layers. We believe this is caused by the different usage of query and key. Considering the normalization in $softmax$ function (Equation 3) which is effected on the keys, each key should consider its relationships to other items. This is why the keys require more semantic information in SANs. The results in Table 3 show that self-attention networks indeed benefit more from incorporating global information into keys than that of queries. However, it should be noted that enhancing the queries with context representations can further improves the performance.

Source Context vs. Target Context

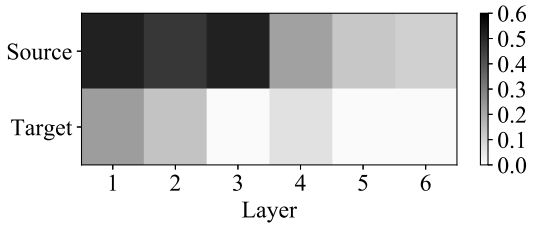

Figure 3: Visualization of weights learned for source-side context vectors and target-side context vectors when we integrate the context-aware model on the both sides of TRANSFORMER. Obviously, target context vectors are allocated with much lower weights than its source-side counterpart.

In this part, we investigated why integrating encoder-side and decoder-side context vectors fails to further improve the self-attention model. We took a deep look into the Model #8 (See Table 1), and averaged the gating scalar of the source-side and target-side context vectors, respectively. As observed in Figure 3, the target-side context vectors consistently gain minor weights, resulting in less contribution from the target-side contextual information. Concerning the source-side context vectors, the factors automatically allocate larger proportion. The result verified our claim that the top-layer of encoder representations has already embedded with useful contextual information, which has exploited to the target-side representations through the conventional encoder-decoder attention network. Thus, the decoder-side context-aware model does not further improve the translation quality as expected.

Linguistic Analysis

The last, we provided linguistic analyses to the proposed models in terms of: 1) whether the proposed model is flexibly as expected to utilize the contextual information for different words; and 2) how the proposed models perform on sentences of different lengths.

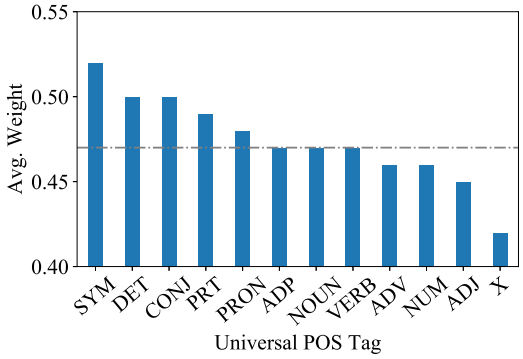

Figure 4: The weights of deep-global context vectors corresponding to different POS. Grey line indicates the average weight of all the words. As observed, the function words require more contextual information than content words.

Analysis on Part-of-Speech Since the context representations are element-wised added to the SAN model, an interesting question is whether different words are assigned with distinct weights. We categorized the words in validation set using the Part-of-Speech (POS) tag set.[3] Figure 4 shows the factors learned for controlling the weight of context vectors. The function words, which have very little substantive meaning (e.g, "SYM", 'DET", "CONJ", "PRT", "PRON" and "ADP"), require more contextual information than that of content words such as the nouns, verbs, adjectives, and adverbs. We attribute the phenomena to the fact that the representations of function words profit more from contextual information, which also noted by Wang et al. (2018) who suggested to reconstruct the function words (e.g. the pronouns) to alleviate the problem of dropped pronoun.

Analysis on Long Sentences We followed Tu et al. (2016) to evaluate the effect of context-aware models on long sentences. The sentences were divided into 10 disjoint groups according to their lengths and their BLEU scores were evaluated, as shown in Figure 5.

The proposed approaches outperform the baseline model in almost all the length segments. There are still consid-

[3]Including: "SYM"-symbols, "DET"- determiner, "CONJ"-conjuntion, "PRT"-partical, "PRON"-pronoun, "ADP"-adposition, "NOUN"-noun, "VERB"-verb, "ADV"-adverb, "Num"-number, "ADJ"-adjective, and "X"-others.

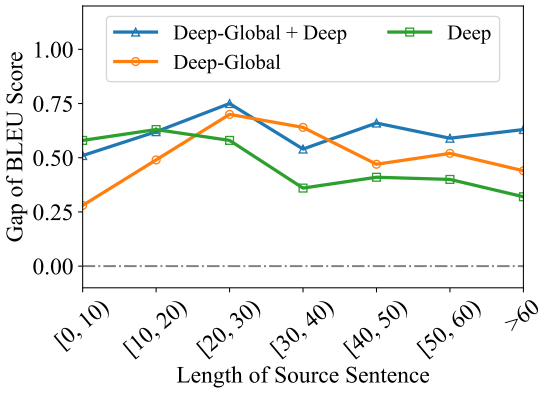

Figure 5: Performance improvement according to various input sentence lengths. Y-axis denotes the gap of BLEU score between our model and baseline (grey line).

erable differences between the global-based and the deep-based variations. Global-based strategies consistently outperform the deep-context model on sentences with more than 20 words, while the opposite situation appears on the shorter sentences. One possible reason is that translating long sentences require more long-distance dependency information which can be supplemented by the global contextual information. For short sentences, the effect of global context is relatively minor, while the complex syntactic and semantic dependencies from deep context provide more impact on the translation quality.

Related Work

Neural representations embed complex characteristics of word use (e.g., syntax and semantic) (Choi, Cho, and Bengio 2017). Several researchers have shown that contextual information can enhance the ability of modeling dependencies among neural representations, especially for the attention models (Bahdanau, Cho, and Bengio 2015; Luong, Pham, and Manning 2015). For example, Tu et al. (2017) and Zhang et al. (2017) respectively enhanced the query and memory of a conventional attention model with internal contextual representations. Their studies verified the necessity of global contextual information for modeling the dependencies between representations. Moreover, Peters et al. (2018) pointed out that deeply modeling syntactic and semantic contexts from multiple layers benefits to the performance of multi-layer neural networks. Contrary to the prior studies explored on RNN-based approaches or required external resources, our work focuses on improving the self-attention networks with contextual information.

Although self-attention model has shown its strength in modeling discrete sequence on different tasks, e.g machine translation (Vaswani et al. 2017), natural language inference (Shen et al. 2018) and acoustic modeling (Sperber et al. 2018), several studies have mentioned the limitations in conventional self-attention networks (Chen et al. 2018). Among them, Yang et al. (2018a) noted that restricting the attention model to a local space may benefit to the performance, supporting that the conventional SAN model insufficiently fully capturing the context of a sequence. Tang et al. (2018) found that the conventional SANs fail to fully take the advantages of direct connections between elements. Moreover, Shaw, Uszkoreit, and Vaswani (2018) succeed on incorporating relative positions into the SAN models, supporting that the conventional model requires necessary information for modeling relations between the states. Bawden et al. (2018) and Voita et al. (2018) enhanced the attention network with external contextual representations that summarizes previous source sentences. Unlike their work which requires the embeddings of previous sentences, our approaches contextualize the transformations in SANs, thus avoid relying on external resources and maintain the simplicity and flexibility.

Conclusion

In this work, we improved the self-attention networks with contextual information. We proposed several simple but effective strategies to model the contexts, and found that the deep and global approaches are complementary to each other. Experimental results across language pairs demonstrate the effectiveness and universality of the proposed approach. Extensive analyses show that how the context-aware model enhances the original representations in the self-attention model, and our model is able to flexibly model the contextual information for different representations.

It is interesting to validate the model in other tasks, such as reading comprehension, language inference, and stance classification (Xu et al. 2018). Another promising direction is to design more linguistic context-aware techniques, such as incorporating the linguistic knowledge (e.g. phrases and syntactic categories). It is also interesting to combine with other techniques (Shaw, Uszkoreit, and Vaswani 2018; Li et al. 2018; Dou et al. 2018; Dou et al. 2019; Kong et al. 2019; Yang et al. 2018b) to further boost the performance of Transformer.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (Grant No. 61672555), the Joint Project of Macao Science and Technology Development Fund and National Natural Science Foundation of China (Grant No. 045/2017/AFJ) and the Multiyear Research Grant from the University of Macau (Grant No. MYRG2017-00087-FST). We thank the anonymous reviewers for their insightful comments.

References

Anastasopoulos, A., and Chiang, D. 2018. Tied Multitask Learning for Neural Speech Translation. In *NAACL*.

Bahdanau, D.; Cho, K.; and Bengio, Y. 2015. Neural Machine Translation by Jointly Learning to Align and Translate. In *ICLR*.

Bawden, R.; Sennrich, R.; Birch, A.; and Haddow, B. 2018. Evaluating Discourse Phenomena in Neural Machine Translation. In *NAACL*.

Britz, D.; Goldie, A.; Luong, M.-T.; and Le, Q. 2017. Massive Exploration of Neural Machine Translation Architectures. In *EMNLP*.

Calixto, I.; Liu, Q.; and Campbell, N. 2017. Doubly-Attentive Decoder for Multi-modal Neural Machine Translation. In *ACL*.

Chen, M. X.; Firat, O.; Bapna, A.; Johnson, M.; Macherey, W.; Foster, G.; Jones, L.; Schuster, M.; Shazeer, N.; Parmar, N.; Vaswani, A.; Uszkoreit, J.; Kaiser, L.; Chen, Z.; Wu, Y.; and Hughes, M. 2018. The Best of Both Worlds: Combining Recent Advances in Neural Machine Translation. In *ACL*.

Cho, K.; van Merrienboer, B.; Gülçehre, Ç.; Bahdanau, D.; Bougares, F.; Schwenk, H.; and Bengio, Y. 2014. Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. In *EMNLP*.

Choi, H.; Cho, K.; and Bengio, Y. 2017. Context-dependent word representation for neural machine translation. *CSL*.

Dou, Z.; Tu, Z.; Wang, X.; Shi, S.; and Zhang, T. 2018. Exploiting Deep Representations for Neural Machine Translation. In *EMNLP*.

Dou, Z.; Tu, Z.; Wang, X.; Wang, L.; Shi, S.; and Zhang, T. 2019. Dynamic Layer Aggregation for Neural Machine Translation. In *AAAI*.

Gkioxari, G.; Girshick, R.; and Malik, J. 2015. Contextual Action Recognition with R* CNN. In *ICCV*.

Hariharan, B.; Arbeláez, P.; Girshick, R.; and Malik, J. 2015. Hypercolumns for Object Segmentation and Fine-grained Localization. In *CVPR*.

Hassan, H.; Aue, A.; Chen, C.; Chowdhary, V.; Clark, J.; Federmann, C.; Huang, X.; Junczys-Dowmunt, M.; Lewis, W.; Li, M.; et al. 2018. Achieving Human Parity on Automatic Chinese to English News Translation. *arXiv:1803.05567*.

Huang, G.; Liu, Z.; van der Maaten, L.; and Weinberger, K. Q. 2017. Densely Connected Convolutional Networks. In *CVPR*.

Kong, X.; Tu, Z.; Shi, S.; Hovy, E.; and Zhang, T. 2019. Neural Machine Translation with Adequacy-Oriented Learning. In *AAAI*.

Li, J.; Tu, Z.; Yang, B.; Lyu, M. R.; and Zhang, T. 2018. Multi-Head Attention with Disagreement Regularization. In *EMNLP*.

Lin, Z.; Feng, M.; Santos, C. N. d.; Yu, M.; Xiang, B.; Zhou, B.; and Bengio, Y. 2017. A structured self-attentive sentence embedding. In *ICLR*.

Luong, T.; Pham, H.; and Manning, C. D. 2015. Effective Approaches to Attention-based Neural Machine Translation. In *EMNLP*.

Papineni, K.; Roukos, S.; Ward, T.; and Zhu, W.-J. 2002. BLEU: A Method for Automatic Evaluation of Machine Translation. In *ACL*.

Peters, M. E.; Neumann, M.; Iyyer, M.; Gardner, M.; Clark, C.; Lee, K.; and Zettlemoyer, L. 2018. Deep Contextualized Word Representations. In *NAACL*.

Sennrich, R.; Haddow, B.; and Birch, A. 2016. Neural Machine Translation of Rare Words with Subword Units. In *ACL*.

Shaw, P.; Uszkoreit, J.; and Vaswani, A. 2018. Self-Attention with Relative Position Representations. In *NAACL*.

Shen, T.; Zhou, T.; Long, G.; Jiang, J.; Pan, S.; and Zhang, C. 2018. DiSAN: Directional Self-Attention Network for RNN/CNN-Free Language Understanding. In *AAAI*.

Shi, X.; Padhi, I.; and Knight, K. 2016. Does String-based Neural MT Learn Source Syntax? In *EMNLP*.

Sperber, M.; Niehues, J.; Neubig, G.; Stüker, S.; and Waibel, A. 2018. Self-Attentional Acoustic Models. *arXiv:1803.09519*.

Tang, G.; Müller, M.; Rios, A.; and Sennrich, R. 2018. Why Self-Attention? A Targeted Evaluation of Neural Machine Translation Architectures. In *EMNLP*.

Tu, Z.; Lu, Z.; Liu, Y.; Liu, X.; and Li, H. 2016. Modeling Coverage for Neural Machine Translation. In *ACL*.

Tu, Z.; Liu, Y.; Lu, Z.; Liu, X.; and Li, H. 2017. Context Gates for Neural Machine Translation. *TACL*.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is All You Need. In *NIPS*.

Voita, E.; Serdyukov, P.; Sennrich, R.; and Titov, I. 2018. Context-Aware Neural Machine Translation Learns Anaphora Resolution. In *ACL*.

Wang, L.; Tu, Z.; Way, A.; and Liu, Q. 2017. Exploiting Cross-Sentence Context for Neural Machine Translation. In *EMNLP*.

Wang, L.; Tu, Z.; Way, A.; and Liu, Q. 2018. Learning to Jointly Translate and Predict Dropped Pronouns with a Shared Reconstruction Mechanism. In *EMNLP*.

Xu, K.; Ba, J.; Kiros, R.; Cho, K.; Courville, A. C.; Salakhutdinov, R.; Zemel, R. S.; and Bengio, Y. 2015. Show, Attend and Tell: Neural Image Caption Generation with Visual Attention. In *ICML*.

Xu, C.; Paris, C.; Nepal, S.; and Sparks, R. 2018. Cross-Target Stance Classification with Self-Attention Networks. In *ACL*.

Yang, B.; Wong, D. F.; Xiao, T.; Chao, L. S.; and Zhu, J. 2017. Towards Bidirectional Hierarchical Representations for Attention-based Neural Machine Translation. In *EMNLP*.

Yang, B.; Tu, Z.; Wong, D. F.; Meng, F.; Chao, L. S.; and Zhang, T. 2018a. Modeling Localness for Self-Attention Networks. In *EMNLP*.

Yang, B.; Wang, L.; Wong, D. F.; Chao, L. S.; and Tu, Z. 2018b. Convolutional Self-Attention Network. *arXiv:1810.13320*.

Zhang, B.; Xiong, D.; Su, J.; and Duan, H. 2017. A Context-Aware Recurrent Encoder for Neural Machine Translation. *TASLP*.

Zhang, B.; Xiong, D.; and Su, J. 2018. Accelerating Neural Transformer via an Average Attention Network. In *ACL*.

Zhu, G.; Porikli, F.; and Li, H. 2016. Beyond Local Search: Tracking Objects Everywhere with Instance-specific Proposals. In *CVPR*.