Improving Neural Machine Translation with Pre-trained Representation

Rongxiang Weng[1], Heng Yu[1], Shujian Huang[2], Weihua Luo[1], and Jiajun Chen[2]

[1]Machine Intelligence Technology Lab, Alibaba DAMO Academy
[2]National Key Laboratory for Novel Software Technology, Nanjing University
wengrongxiang@gmail.com, {yuheng.yh,weihua.luowh}@alibaba-inc.com,
{huangsj,chenjj}@nju.edu.cn

Abstract

Monolingual data has been demonstrated to be helpful in improving the translation quality of neural machine translation (NMT). The current methods stay at the usage of word-level knowledge, such as generating synthetic parallel data or extracting information from word embedding. In contrast, the power of sentence-level contextual knowledge which is more complex and diverse, playing an important role in natural language generation, has not been fully exploited. In this paper, we propose a novel structure which could leverage monolingual data to acquire sentence-level contextual representations. Then, we design a framework for integrating both source and target sentence-level representations into NMT model to improve the translation quality. Experimental results on Chinese-English, German-English machine translation tasks show that our proposed model achieves improvement over strong Transformer baselines, while experiments on English-Turkish further demonstrate the effectiveness of our approach in the low-resource scenario. [1]

1 Introduction

Neural machine translation (NMT) based on an encoder-decoder framework (Sutskever et al., 2014; Cho et al., 2014; Bahdanau et al., 2014; Luong et al., 2015) has obtained state-of-the-art performances on many language pairs (Deng et al., 2018). Various advanced neural architectures have been explored for NMT under this framework, such as recurrent neural network (RNN) (Bahdanau et al., 2014; Luong et al., 2015, RNNSearch), convolutional neural network (CNN) (Gehring et al., 2016, Conv-S2S), self-attention network (Vaswani et al., 2017, Transformer).

Currently, most NMT systems only utilize the sentence-aligned parallel corpus for model training. Monolingual data, which is larger and easier to collect, is not fully utilized limiting the capacity of the NMT model. Recently, several successful attempts have been made to improve NMT by incorporating monolingual data (Gulcehre et al., 2015; Sennrich et al., 2016; Zhang and Zong, 2016; Poncelas et al., 2018), and reported promising improvements.

However, these studies only focus on the usage of word-level information, e.g. extracting information from word embedding. Meanwhile, several methods such as ELMo (Peters et al., 2018), GPT (Radford et al., 2018) and BERT (Devlin et al., 2018) have gained tremendous success by modeling sentence representation. Thus, the upcoming question is whether the sentence representation is useful in machine translation, and these methods of acquiring and integrating could be applied directly to it.

In this paper, we compare these methods and demonstrate it is non-trivial to propose new methods to acquire sentence representation and integrate it into NMT for three main reasons: First, due to the limited amount of parallel data, compared with word level representation, the NMT models may not generate appropriate sentence representation, which is more complex and diverse. However, sentence representation is important in text generation. So, the pre-trained sentence representation could be an excellent complement to provide proper information for NMT. Second, the current methods of acquiring pre-trained representations don't fully model the sequential information in the sentence, which is a fundamental factor for modeling sentence. Last, machine translation, which is a *bilingual language generation* task, has not been well studied for how to integrate external knowledge into it. For example, the gen-

---

[1]In Progress

eration process is on the fly during the inference stage, most current methods could not benefit the decoder with the partially translated inputs.

To address the challenges above, we propose a solution to better leverage the pre-trained sentence representation to improve NMT as follows: First, we propose a *bi-directional self-attention language model* (BSLM) to acquire sentence representation which is trained by the monolingual data. The BSLM could capture the forward and backward sequential information from a sentence to build better representations.

Then, we design a framework to integrate effectively task-specific information from pre-trained representation into NMT on both source and target sides. We propose a *weighted-fusion mechanism* for fusing the task-specific representation into the encoder. For the decoder, we use a *knowledge transfer paradigm* to learn task-specific knowledge from the pre-trained representation. This framework could be applied to various neural structures based on the encoder-decoder framework (Bahdanau et al., 2014; Gehring et al., 2016; Vaswani et al., 2017) and language generation tasks.

To demonstrate the effectiveness of our approach, we implement the proposed approach on the current state-of-the-art Transformer model (Vaswani et al., 2017). Experimental results on Chinese-English and German-English translation tasks show that our approach improves over the Transformer baseline on the standard data-sets. Moreover, we show that on low resource language pair like English-Turkish, our approach leads to more improvements.

2   Approach

In this section, we will introduce our proposed solution for acquiring and integrating sentence representation in detail. First, we propose a *bi-directional self-attention language model* (BSLM) trained on large scale unlabeled monolingual sentences to get *sentence representation*. Next, we introduce two individual methods to employ the generated representation: a *weighted-fusion mechanism* and a *knowledge transfer paradigm* to enhance the encoder and decoder by fusing and learning the information from the sentence representation, respectively.
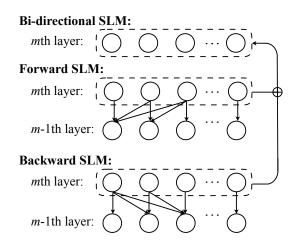

Figure 1: Overview of the bidirectional self-attention language model (BSLM).

2.1   Bi-directional Self-Attention Language Model

Previous studies (Peters et al., 2018; Radford et al., 2018; Devlin et al., 2018) indicate that sentence representation generated by the pre-trained model could help the downstream tasks. Furthermore, compared with traditional RNN (Jozefowicz et al., 2016; Melis et al., 2017; Peters et al., 2018) structure, self-attention network (SAN), which could achieve wider and deeper information, has a stronger ability to extract features (Tang et al., 2018).

However, most of these SAN based pre-trained methods ignore the sequential information which is a fundamental factor for sentence modeling. To fully acquire sentence representation with sequential information, we propose a bi-directional self-attention language model (BSLM). Our BSLM model consists of a forward and a backward self-attention language model (SLM) which can capture the forward and backward sequential information by using the directional matrix (Shen et al., 2017) augments sequential relation (see Figure 1). Moreover, inspired by Wang et al. (2018) and Dou et al. (2018), we gather all layers' representations from BSLM which contain different aspects of information for a sentence.

Specifically, the forward SLM has M layers and the representation in the mth layer is defined as:

$$\overrightarrow{\mathbf{R}}_m^L = (\overrightarrow{\mathbf{r}}_{m,1}^L, \overrightarrow{\mathbf{r}}_{m,2}^L, \cdots, \overrightarrow{\mathbf{r}}_{m,k}^L, \cdots, \overrightarrow{\mathbf{r}}_{m,K}^L),$$

where K is the number of vectors in the $\overrightarrow{\mathbf{R}}_m^L$. The representation matrix is computed by:

$$\overrightarrow{\mathbf{R}}_m^L = \text{LN}(\text{FFN}(\overrightarrow{\mathbf{H}}_m) + \overrightarrow{\mathbf{R}}_{m-1}^L), \qquad (1)$$

The $\overrightarrow{\mathbf{H}}_m$ is calculated by:

$$\overrightarrow{\mathbf{H}}_m = \text{MultiHead}(\mathbf{Q}, \mathbf{K}, \mathbf{V})$$
$$= \text{Concat}(\mathbf{head}_1, \cdots, \mathbf{head}_H), \quad (2)$$

where $\mathbf{Q}$, $\mathbf{K}$, $\mathbf{V}$ are query, key and value matrixes that are equal to $\overrightarrow{\mathbf{R}}_{m-1}^L$.

We introduce a *directional matrix* proposed by Shen et al. (2017) in vanilla self-attention network (Vaswani et al., 2017), in which a mask matrix is used to get the sequential information by covering the rear words for each input word. So a single attention head is:

$$\mathbf{head}_\text{h} = \text{DirAtt}(\mathbf{Q}, \mathbf{K}, \mathbf{V})$$
$$= \text{softmax}(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{d_k}} + \mathbf{Mask})\mathbf{V}. \quad (3)$$

The **Mask** is calculated by:

$$\mathbf{Mask}_{c,v} = \begin{cases} -\infty, & c < v \\ 0, & otherwise, \end{cases} \quad (4)$$

the row and column of the **Mask** are equal to K.

Typically, $\overrightarrow{\mathbf{r}}_{1,k}^L$ from $\overrightarrow{\mathbf{R}}_1^L$ is defined as $\overrightarrow{\mathbf{r}}_{1,k}^L = \text{emb}^L(w_k)$, where w_k is a input word and $\text{emb}^L(w_k)$ is the word embedding of w_k. Therefore, we have $\overrightarrow{\mathbf{R}}_1^L$ with the given input: $\mathbf{w} = (w_1, w_2, \cdots, w_k, \cdots, w_K)$.

In the training stage, the kth output word is computed by maximizing the conditional probability, which is defined as:

$$P(w_k|w_{<k}) = \text{softmax}(\overrightarrow{\mathbf{r}}_{M,k}), \quad (5)$$

The SLM is optimized by maximizing the likelihood, defined as:

$$\mathcal{L}_\text{L} = \frac{1}{K} \sum_{k=1}^{K} \log P(w_k|w_{<k}). \quad (6)$$

Then, the representation from the forward SLM is:

$$\overrightarrow{\mathbf{R}}^L = (\overrightarrow{\mathbf{R}}_1^L, \cdots, \overrightarrow{\mathbf{R}}_m^L, \cdots, \overrightarrow{\mathbf{R}}_M^L).$$

The structure and training process of the backward SLM are similar to the forward one. At last, we can get the sentence representation $\mathbf{R}^L$ containing forward and backward sequential information by adding the $\overrightarrow{\mathbf{R}}^L$ and $\overleftarrow{\mathbf{R}}^L$, which are from the forward and backward SLMs, respectively.

2.2 Integrating Sentence Representation in Transformer

Due to the limited amount of parallel data, it is hard for the Transformer to generate appropriate sentence representation. The pre-trained BSLM could be an excellent complement to provide Transformer with proper sentence level information.

However, the previous integration methods like initializing parameters from pre-trained model may not suit for the machine translation which is a bilingual generation task. The general representation is quite different from the task-specific representation of NMT model. Thus, we propose a novel integration framework to fully utilize pre-trained sentence representation in the NMT model.

Weighted-fusion Mechanism. On the source side, partially inspired by Peters et al. (2018), we propose to use a *weighted-fusion mechanism* to generate *task-specific representation* from the pre-trained sentence representations, and fuse them into the encoder of Transformer through a gating network.

Different from previous works (Ramachandran et al., 2017; Peters et al., 2018; Radford et al., 2018), the proposed feature-based method can make a *deep fusion* which could incorporate appropriate information into each layer of the encoder, that is, Transformer can access the specific surface information in lower layers and the latent representation for its higher layers.

Formally, for the nth layer of the encoder, the vanilla representation $\mathbf{R}_n^S$ is computed by Equation 1-3, in which the **Mask** is a zero matrix. The sentence representation $\mathbf{R}^L$ from BSLM will be fused into $\mathbf{R}_n^S$ by:
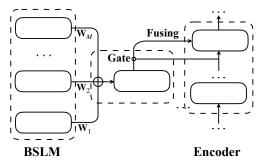
$$\mathbf{R}_n^S = \mathbf{R}_n^S + \theta_n * \mathbf{R}_n^W, \quad (7)$$

θ_n is a dynamic gate which means how much the current layer needs the extra information. The gate is computed by each state from the $\mathbf{R}_n^S$:
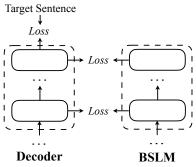
$$\theta_n = \text{sigmoid}(\bar{\mathbf{r}}_n^S), \quad (8)$$

$$\bar{\mathbf{r}}_n^S = \frac{1}{I} \sum_{i=1}^{I} \mathbf{r}_{n,i}^S, \quad (9)$$

I is the length of the source sentence $\mathbf{x}$. We denote $\mathbf{r}_{n,i}^S$ to be the word embedding of the corresponding word x_i in the $\mathbf{x}$. The task-specific representa-

(a) The weighted-fusion mechanism used in the encoder.

(b) The knoledge transfer paradigm used in the decoder.

Figure 2: Overview of the proposed integration framework.

tion $\mathbf{R}_n^W$ is calculated by:

$$\mathbf{R}_n^W = \sum_{m=1}^{M} (\mathbf{W}_{n,m} * \mathbf{R}_m^L). \qquad (10)$$

$\mathbf{W}_{n,m}$ is a trainable weight representing the importance of $\mathbf{R}_m^L$ learned in the training process.

Knowledge Transfer Paradigm. Compared with the encoder, the decoder is difficult to exploit the pre-trained knowledge whenever using feature-based or fune-tuning methods (Devlin et al., 2018; Peters et al., 2018).

For example, the *exposure bias* (Lee et al., 2018; Wu et al., 2018) leads to that the ground-truth sentence representation which is generated by the reference is not available in the inference stage. On the other hand, the decoding process involves reading the source side representations, these parameters are hard to initialized by pre-trained models. It leads to the fine-tuning methods do not work well here (Radford et al., 2018).

Therefore, we propose a simple *knowledge transfer paradigm* to capture the bi-directional sequential information by learning the pre-trained sentence representation in the training stage. This method could transfer the knowledge from monolingual data to the NMT model and could avoid the problems mentioned above.

We design an auxiliary learning objective besides traditional translation objective, which is meant to transfer bi-directional sentence knowledge from BSLM to the NMT model. Formally, the auxiliary knowledge transfer objective of our proposed paradigm is:

$$\mathcal{L}_E = \frac{1}{J} \sum_{n=1}^{N} \sum_{j=1}^{J} ||\mathbf{r}_{n,j}^T - \mathbf{r}_{n,j}^L||_2^2, \qquad (11)$$

where the J is the length of given target sentence $\mathbf{y}$, the N is number of layers. The $\mathbf{r}_{n,j}^T$ and $\mathbf{r}_{n,j}^L$ are from the decoder and BSLM, respectively. Note that the number of layers of BSLM and the decoder should be the same to utilize the knowledge transfer method.

The translation model is optimized by maximizing the likelihood of the $\mathbf{y}$ given source sentence $\mathbf{x}$:

$$\mathcal{L}_M = \frac{1}{J} \sum_{j=1}^{J} \log P(y_j|y_{<j}, \mathbf{x}). \qquad (12)$$

The $P(y_j|y_{<j}, \mathbf{x})$ is computed by Equation 5. Finally, the loss function of our model is:

$$\mathcal{L}_T = \mathcal{L}_M + \mathcal{L}_E. \qquad (13)$$

3 Experiment

3.1 Implementation Detail

Data-sets. We conduct experiments on Chinese→English (ZH→EN), German→English (DE→EN) and English→Turkish (EN→TR) translation tasks.

On the ZH→EN tasks, training set consists of about 1 million sentence pairs from LDC.[2] We use `MT02` as our validation set, and `MT03`, `MT04` and `MT05` as our test sets. We use 8 million monolingual sentences from LDC[3]. On the

---

#	Model	MT02	MT03	MT04	MT05	Average	Δ
1	RNNSearch (Luong et al., 2015)	N/A	28.38	30.85	26.78	–	–
2	(Sennrich et al., 2016)	36.95	36.80	37.99	35.33	–	–
3	(Zhang and Zong, 2016)	N/A	33.38	34.30	31.57	–	–
4	(Cheng et al., 2016)	38.78	38.32	38.49	36.45	–	–
5	(Zhang et al., 2018)	N/A	43.26	N/A	41.61	–	–
6	Transformer	44.77	44.93	45.81	43.04	44.59	–
7	+ ELMo (Peters et al., 2018)	45.23	45.60	46.26	43.61	45.16	+0.57
8	+ GPT (Radford et al., 2018)	44.89	45.22	45.99	43.31	44.84	+0.25
9	+ BERT (Devlin et al., 2018)	45.02	45.53	46.02	43.52	45.02	+0.43
Effectiveness of weighted-fusion mechanism used in the different layers							
10	+ Weighted-fusion (*shallow*)	44.97	45.21	46.19	43.23	44.88	+0.29
11	+ Weighted-fusion (*deep*)	45.46	45.62	46.57	43.82	45.34	+0.75
Effectiveness of knowledge transfer paradigm used in the different layers							
12	+ Knowledge Transfer (*shallow*)	45.61	45.63	46.54	43.86	45.34	+0.75
13	+ Knowledge Transfer (*deep*)	45.71	45.78	46.62	43.94	45.45	+0.85
Our proposed model							
14	+ Our Approach	**45.82**	**45.86**	**46.83**	**44.13**	**45.61**	**+1.01**

Table 1: Translation qualities on the ZH→EN experiments. *deep* and *shallow* mean employing proposed methods on the all layers or the first layer, respectively.

DE→EN tasks, We use WMT-16 as our training set, which consists of about 4.5 million sentence pairs. We use `newstest2015` (NST15) as our validation set, and `newstest2016` (NST16) as test sets [4]. We use 40 million monolingual sentences from WMT-16 Common Crawl data-set. On the EN→TR tasks, We use WMT-16 as our training set, which consists of about 0.2 million sentence pairs. We use `newsdev2016` (NSD16) as our validation set and `newstest2016` (NST16) as test sets. We use 5 million monolingual sentences from WMT-16 Common Crawl data-set.

Setting. We apply byte pair encoding (BPE) (Sennrich et al., 2015) to encode all sentences from the parallel and monolingual data-sets and limit the vocabulary size to 32K. Out-of-vocabulary words are denoted to the special token *UNK*.

we set the dimension of input and output of all layers for the bi-directional self-attention language model (BSLM) and Transformer as 512, and that of the feed-forward layer as 2048. We employ eight parallel attention heads. The number of layers for BSLM and each side of the Transformer is set to 6. Sentence pairs are batched together by approximate sentence length. Each batch has approximately 4096 tokens. We use Adam (Kingma

and Ba, 2014) optimizer to update parameters, and the learning rate was varied under a warm-up strategy with 4000 steps (Vaswani et al., 2017).

After training, we use the beam search for heuristic decoding, and the beam size is set as 4. We measure the translation quality with the IBM-BLEU score (Papineni et al., 2002). We implement our model upon our in-house Transformer system.

3.2 Results on Standard Data-sets

Chinese→English. The results on ZH→EN task are shown in Table 1. We report several results from previous studies about using monolingual data in NMT. For fair comparison, we also implement ELMo (Peters et al., 2018), GPT (Radford et al., 2018) and BERT (Devlin et al., 2018) in our Transformer system. The implement details are as follows:

- ELMo: Peters et al. (2018) proposed a contextual representation pre-trained from Bi-LSTM based language model. We train a three layers Bi-LSTM language model[5] by using our monolingual corpus. Then, the contextual representation is concatenated on the source side of the Transformer.

- GPT: Radford et al. (2018) proposed to use a pre-trained self-attention language model

---

Model	NST15	NST13	NST14	NST16	Average	Δ
Transformer	30.33	29.14	28.87	35.74	31.25	−
+ Weighted-fusion (*deep*)	31.43	29.72	29.46	36.48	31.89	+0.64
+ Knowledge Transfer (*deep*)	31.32	29.61	29.73	36.60	31.98	+0.73
+ Our Approach	**31.72**	**30.32**	**30.29**	**36.91**	**32.51**	**+1.26**

Table 2: Translation qualities on the DE→EN experiments.

to initialize the downstream tasks. We implement it by using a forward SLM initialize the parameters of the Transformer's source side.

- BERT: Devlin et al. (2018) proposed to use parameters from a pre-trained bi-directional encoder optimized by special objectives to initialize the other tasks. Following them, we initialize the encoder of our translation model with the parameters from the pre-trained BERT[6] with our monolingual data.

In Table 1, we can see that the weighted-fusion mechanism in all layers contributes to 0.75 improvements (line 11), while employing knowledge transfer paradigm for the decoder of NMT can achieve 0.85 BLEU score improvement (line 13). Our proposed model, combining both methods, significantly improve 1.01 BLEU score (line 14) over a strong Transformer baseline. To further test our integration framework, we also try the shallow integration (*shallow*) by using proposed methods only in the first layer of the NMT network (line 10 and 12). The results show that deep integration (*deep*), which utilizes the proposed integration framework in all layers of the Transformer, can achieve a noticeable improvement compared to their shallow counterparts.

Compared with the related work on model pre-training, ELMo (Peters et al., 2018) doesn't yield a significant improvement (+0.57) when integrated with Transformer (line 6). We conjecture that there are two possible reasons: first, the representation from the Bi-LSTM based language model can't be deeply fused with Transformer network due to the structural difference, so the relatively shallow integration leads to the performance degradation. Second, the char-CNN (Peters et al., 2018) in their work can't capture enough information when the input is subword (Sennrich et al., 2015), which limits the performance of ELMo.

The GPT (Radford et al., 2018) and BERT (Devlin et al., 2018) methods improve 0.25 and 0.43

BLEU score, respectively. Compared with the results in other monolingual tasks, the improvement is relatively small. The main reason is that these methods for parameter initialization can't fully exploit the pre-trained model and may not suit for the bilingual task as machine translation.

German→English. To show the generalization ability of our approach, we carry out experiments on DE→EN translation task. The results are shown in Table 2. With the weighted-fusion mechanism, the BLEU gain is 0.64, while the knowledge transfer paradigm leads to 0.73 improvements. The combination of both yields a further improvement (+1.26). It shows that our model can achieve improvements in different types of language pairs on large scale data-sets.

3.3 Results on Low-resource Data-sets

We report the results on the EN→TR task which is a relatively low-resource language pair. Furthermore, we compare with the back-translation method (Sennrich et al., 2016) which is predominant in the low-resource scenario. We translate the target language (TR) to source language (EN) by a reversely-trained NMT system, and copy the parallel data set to the same size. At last, we shuffle the training data which includes pseudo and parallel sentence pairs. The different sizes of back-translated corpus are shown in Table 3. We translate 0.2M (*small size*), 0.4M (*medium size*) and 1M (*big size*) as pseudo data-sets. The ratios of pseudo and parallel sentence pairs are 1:1, 2:1 and 5:1.

The results are shown in Table 4, our method can improve the 0.71 BLEU score compared to the baseline system. Moreover, when combining with pseudo data-sets (+BT), our model can achieve further improvement. Typically, when using the large size pseudo corpus, our model can improve 1.77 BLEU score, which is a prominent gain in the low-resource scenario.

---

[6]code: https://github.com/codertimo/BERT-pytorch

Size	Pseudo	Parallel	Ratio
Small	0.2M	0.2M	1:1
Medium	0.4M	0.2M	2:1
Big	1.0M	0.2M	5:1

Table 3: The different sizes of pseudo corpus.

Model	NSD16	NST16	Δ
Transformer	15.19	13.72	−
+ Our Approach	15.55	14.43	+0.71
+ BT (*small*)	15.58	14.64	+0.92
+ Our Approach	15.84	14.99	+0.95
+ BT (*medium*)	15.70	14.86	+1.14
+ Our Approach	15.89	15.01	+1.29
+ BT (*big*)	16.04	15.19	+1.47
+ Our Approach	**16.36**	**15.49**	**+1.77**

Table 4: Translation qualities on the EN→TR experiments. (BT: back-translation; *small*, *medium* and *big*: the different size of pseudo corpus.)

3.4 Decoding Efficiency

Our BSLM is highly parallelizable which could be integrated into the Transformer without losing efficiency. In order to verify this, we compare the decoding efficiency of the baseline, our model and ELMo (Peters et al., 2018).

The results are shown in Table 5, our model leads to a very little drop in decoding efficiency compared to the baseline (15.06 sentences/second vs. 14.12 sentences/second) because the self-attention structure of our model can generate sentence representation in parallel. Whereas for ELMo, the efficiency decline drastically (8.41 sentences/second) due to the auto-regressive structure of LSTM.

4 Analysis

4.1 Uni-direction Vs. Bi-direction

We compare the effectiveness of uni-directional and bi-directional self-attention language model (SLM) used on the source side. The uni-directional SLM is the forward SLM described in Section 2.1. The results are shown in Table 6, the bi-directional SLM (+1.01) outperforms the uni-directional SLM (+0.83), which verifies the combination power of forward and backward linguistic information captured by bi-directional SLM.

Model	Speed
Transformer	15.06
+ ELMo (Peters et al., 2018)	8.41
+ Our Approach	14.12

Table 5: The comparison of decoding efficiency (sentences/second).

Model	BLEU	Δ
Transformer	44.59	−
+ Uni-directional SLM	45.42	+0.83
+ Bi-directional SLM	**45.61**	**+1.01**

Table 6: The effectiveness of uni-directional and bi-directional self-attention language model (SLM).

4.2 Impact of Different Integration Schema

In this section, we analyze the different integration schema for our BSLM. Specifically, we switch the integration mechanism on source and target side by employing the *knowledge transfer paradigm* on the source side and the weighted-fusion mechanism on the target side. So the knowledge transfer paradigm is used to optimize the source representation directly. And on the target side, we generate the representation by feeding the *partially translated part* to the BSLM and fuse them into the decoder at each time step.

The results are shown in Table 7. The knowledge transfer paradigm on the source side can improve the performance to some extent by refining source side representation but the impact is relatively small comparing to the weighted-fusion mechanism. On the target side, the BLEU score decreases 0.66 when using weighted-fusion in all layers. The problem is that the representation generated by the partially translated part is incomplete and may contain wrong information. This will negatively influence the decoder, especially in higher layers. According to this phenomena, the category of methods which directly fuses pre-trained representation into the target side may not work well.

4.3 Visualization of Weights

We draw the heat map for the learned weights of the weighted-fusion mechanism (Equation 10). As shown in Figure 3, The column axis is the layer of Transformer and row axis is the layer of the BSLM. The number in each row is normalized. The map shows a good correlation between the fusion weights for different layers of Trans-

Model	Side	BLEU
Transformer	N/A	44.59
+ Weighted-fusion (*deep*)	*src.*	45.47
+ Weighted-fusion (*first*)	*trg.*	44.66
+ Weighted-fusion (*deep*)	*trg.*	43.93
+ Knowledge Transfer (*deep*)	*trg.*	45.45
+ Knowledge Transfer (*first*)	*src.*	44.92
+ Knowledge Transfer (*deep*)	*src.*	45.06

Table 7: The comparison of translation qualities for using the weighted-fusion and knowledge transfer methods in different sides (*src.*: source; *trg.*: target) on ZH-EN task.
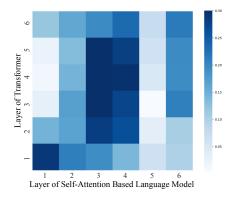

Figure 3: The heat map for the learned weights of the weighted-fusion mechanism.

former decoder and BSLM. It illustrates the advantage of BSLM in the integration with Transformer from another perspective: with the similar network structure, the lower layers of BSLM can be used to model specific surface information, while higher layers provide latent representation.

5 Related Work

Several successful attempts have been made to utilize monolingual data in NMT directly. Sennrich et al. (2016) propose to use back-translation to generate synthetic parallel data from monolingual data for NMT. Currey et al. (2017) propose a copy mechanism to copy fragments of sentences from monolingual data to translated outputs directly. Zhang et al. (2018) propose a joint learning of source-to-target and target-to-source NMT models, generating pseudo parallel data using monolingual data. However, directly using synthetic data suffers from potential noise propagation problem. Our model only learns sentence representation rely on the gold inputs, which can avoid this problem.

On applying the pre-trained model for NMT,

Di Gangi and Federico (2017) use source side pre-trained embedding and integrate it into NMT with a mix-sum/gating mechanism. They only focus on improving the source side's representation, leaving the target side information largely ignored. Ramachandran et al. (2017) propose to initialize the parameters of NMT by a pre-trained language model. However, these methods lack a lasting impact on training, leaving the information from pre-trained models less exploited. Our proposed solutions can generate sentence representations which can be applied on both sides of the NMT models, resulting in much better performance in terms of translation quality.

In the NLP field, sentence representation has been widely explored. Peters et al. (2018) introduce Embedding learned from LSTM based Language Models (ELMo) and successfully apply it in question answering, textual entailment, etc. Partially inspired by them, we propose to use the weighted-fusion mechanism to utilize the sentence representation. Learning information from monolingual data by knowledge transfer is widely used in the several NLP tasks (Tang et al., 2019). We are the first attempt in the neural machine translation. Radford et al. (2018) use the parameters from a pre-trained self-attention structure language model to initialize the downstream tasks. Devlin et al. (2018) extend this idea by using the bi-directional encoder representation from Transformer (BERT) and introducing several training objectives. Different from them, we carefully studied the application of pre-trained model in the bilingual setting of machine translation, our proposed BSLM and integration mechanism take account of both source and target side monolingual data, and can fully explore the information from pre-trained model.

6 Conclusion

In this paper, we propose a novel approach to better leveraging monolingual data for NMT by utilizing pre-trained sentence representations. The sentence representations are acquired through bi-directional self-attention language model (BSLM) and integrated into NMT network by a weighted-fusion mechanism and a knoledge transfer paradigm. Experiments on various languages show that proposed model achieves prominent improvements on the standard data-sets as well as in the low-resource scenario.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *CoRR*.

Yong Cheng, Wei Xu, Zhongjun He, Wei He, Hua Wu, Maosong Sun, and Yang Liu. 2016. Semi-supervised learning for neural machine translation. *arXiv*.

Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *EMNLP*.

Anna Currey, Antonio Valerio Miceli Barone, and Kenneth Heafield. 2017. Copied monolingual data improves low-resource neural machine translation. In *MT*.

Yongchao Deng, Shanbo Cheng, Jun Lu, Kai Song, Jingang Wang, Shenglan Wu, Liang Yao, Guchun Zhang, Haibo Zhang, Pei Zhang, et al. 2018. Alibaba's neural machine translation systems for wmt18. In *Conference on Machine Translation: Shared Task Papers*.

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

Mattia A Di Gangi and M Federico. 2017. Can monolingual embeddings improve neural machine translation? *Proc. of CLiC-it*.

Zi-Yi Dou, Zhaopeng Tu, Xing Wang, Shuming Shi, and Tong Zhang. 2018. Exploiting deep representations for neural machine translation. *arXiv*.

Jonas Gehring, Michael Auli, David Grangier, and Yann N Dauphin. 2016. A convolutional encoder model for neural machine translation. *arXiv*.

Caglar Gulcehre, Orhan Firat, Kelvin Xu, Kyunghyun Cho, Loic Barrault, Huei-Chi Lin, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2015. On using monolingual corpora in neural machine translation. *arXiv*.

Rafal Jozefowicz, Oriol Vinyals, Mike Schuster, Noam Shazeer, and Yonghui Wu. 2016. Exploring the limits of language modeling. *arXiv*.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization.

Jason Lee, Elman Mansimov, and Kyunghyun Cho. 2018. Deterministic non-autoregressive neural sequence modeling by iterative refinement. *arXiv*.

Minh-Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. In *EMNLP*.

Gábor Melis, Chris Dyer, and Phil Blunsom. 2017. On the state of the art of evaluation in neural language models. *arXiv*.

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. *arXiv*.

Alberto Poncelas, Dimitar Shterionov, Andy Way, Gideon Maillette de Buy Wenniger, and Peyman Passban. 2018. Investigating backtranslation in neural machine translation. *arXiv*.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training. *CoRR*.

Prajit Ramachandran, Peter Liu, and Quoc Le. 2017. Unsupervised pretraining for sequence to sequence learning. In *EMNLP*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2015. Neural machine translation of rare words with subword units. *arXiv*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Improving neural machine translation models with monolingual data. In *ACL*.

Tao Shen, Tianyi Zhou, Guodong Long, Jing Jiang, Shirui Pan, and Chengqi Zhang. 2017. Disan: Directional self-attention network for rnn/cnn-free language understanding. *arXiv*.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *NIPS*.

Gongbo Tang, Mathias Müller, Annette Rios, and Rico Sennrich. 2018. Why self-attention? a targeted evaluation of neural machine translation architectures. *arXiv*.

Raphael Tang, Yao Lu, Linqing Liu, Lili Mou, Olga Vechtomova, and Jimmy Lin. 2019. Distilling task-specific knowledge from BERT into simple neural networks. *CoRR*.

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*.

Qiang Wang, Fuxue Li, Tong Xiao, Yanyang Li, Yinqiao Li, and Jingbo Zhu. 2018. Multi-layer representation fusion for neural machine translation. In *COLING*.

Lijun Wu, Xu Tan, Di He, Fei Tian, Tao Qin, Jianhuang Lai, and Tie-Yan Liu. 2018. Beyond error propagation in neural machine translation: Characteristics of language also matter. *arXiv*.

Jiajun Zhang and Chengqing Zong. 2016. Exploiting source-side monolingual data in neural machine translation. In *EMNLP*.

Zhirui Zhang, Shujie Liu, Mu Li, Ming Zhou, and Enhong Chen. 2018. Joint training for neural machine translation models with monolingual data. *arXiv*.