Exploiting Source-side Monolingual Data in Neural Machine Translation

Jiajun Zhang[†] and **Chengqing Zong**[†‡]
[†]University of Chinese Academy of Sciences, Beijing, China
National Laboratory of Pattern Recognition, CASIA, Beijing, China
[‡]CAS Center for Excellence in Brain Science and Intelligence Technology, Shanghai, China
{jjzhang,cqzong}@nlpr.ia.ac.cn

Abstract

Neural Machine Translation (NMT) based on the encoder-decoder architecture has recently become a new paradigm. Researchers have proven that the target-side monolingual data can greatly enhance the decoder model of NMT. However, the source-side monolingual data is not fully explored although it should be useful to strengthen the encoder model of NMT, especially when the parallel corpus is far from sufficient. In this paper, we propose two approaches to make full use of the source-side monolingual data in NMT. The first approach employs the self-learning algorithm to generate the synthetic large-scale parallel data for NMT training. The second approach applies the multi-task learning framework using two NMTs to predict the translation and the reordered source-side monolingual sentences simultaneously. The extensive experiments demonstrate that the proposed methods obtain significant improvements over the strong attention-based NMT.

1 Introduction

Neural Machine Translation (NMT) following the encoder-decoder architecture proposed by (Kalchbrenner and Blunsom, 2013; Cho et al., 2014) has become the novel paradigm and obtained state-of-the-art translation quality for several language pairs, such as English-to-French and English-to-German (Sutskever et al., 2014; Bahdanau et al., 2014; Luong et al., 2015b; Sennrich et al., 2015). This end-to-end NMT typically consists of two recurrent neural networks. The encoder network maps the source sentence of variable length into the context vector representation; and the decoder network generates the target translation word by word starting from the context vector.

Currently, most NMT methods utilize only the sentence aligned parallel corpus for model training, which limits the capacity of the model. Recently, inspired by the successful application of target monolingual data in conventional statistical machine translation (SMT) (Koehn et al., 2007; Chiang, 2007), Gulcehre et al. (2015) and Sennrich et al. (2015) attempt to enhance the decoder network model of NMT by incorporating the target-side monolingual data so as to boost the translation fluency. They report promising improvements by using the target-side monolingual data. In contrast, the source-side monolingual data is not fully explored. Luong et al. (2015a) adopt a simple autoencoder or skip-thought method (Kiros et al., 2015) to exploit the source-side monolingual data, but no significant BLEU gains are reported. Note that, in parallel to our efforts, Cheng et al. (2016b) have explored the usage of both source and target monolingual data using a similar semi-supervised reconstruction method, in which two NMTs are employed. One translates the source-side monolingual data into target translations, and the other reconstructs the source-side monolingual data from the target translations.

In this work, we investigate the usage of the source-side large-scale monolingual data in NMT and aim at greatly enhancing its encoder network so that we can obtain high quality context vector representations. To achieve this goal, we propose two

approaches. Inspired by (Ueffing et al., 2007; Wu et al., 2008) handling source-side monolingual corpus in SMT and (Sennrich et al., 2015) exploiting target-side monolingual data in NMT, the first approach adopts the self-learning algorithm to generate adequate synthetic parallel data for NMT training. In this method, we first build the baseline machine translation system with the available aligned sentence pairs, and then obtain more synthetic parallel data by translating the source-side monolingual sentences with the baseline system.

The proposed second approach applies the multi-task learning framework to predict the target translation and the reordered source-side sentences at the same time. The main idea behind is that we build two NMTs: one is trained on the aligned sentence pairs to predict the target sentence from the source sentence, while the other is trained on the source-side monolingual corpus to predict the reorderd source sentence from original source sentences[1]. It should be noted that the two NMTs share the same encoder network so that they can help each other to strengthen the encoder model.

In this paper, we make the following contributions:

- To fully investigate the source-side monolingual data in NMT, we propose and compare two methods. One attempts to enhance the encoder network of NMT by producing rich synthetic parallel corpus using a self-learning algorithm, and the other tries to perform machine translation and source sentence reordering simultaneously with a multi-task learning architecture.

- The extensive experiments on Chinese-to-English translation show that our proposed methods significantly outperform the strong NMT baseline augmented with the attention mechanism. We also find that the usage of the source-side monolingual data in NMT is more effective than that in SMT. Furthermore, we find that more monolingual data does not always improve the translation quality and only relevant monolingual data helps.

[1]We reorder all the source-side monolingual sentences so as to make them close to target language in word order.

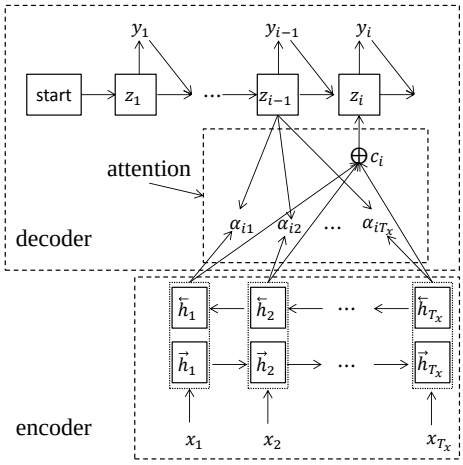

Figure 1: The encoder-decoder NMT with attention.

2 Neural Machine Translation

Our approach on using source-side monolingual corpora can be applied in any neural machine translation as long as it employs the encoder-decoder framework. Without loss of generality, we use the attention-based NMT proposed by (Bahdanau et al., 2014), which utilizes recurrent neural networks for both encoder and decoder as illustrated in Fig. 1.

The encoder-decoder NMT first encodes the source sentence $X = (x_1, x_2, \cdots, x_{T_x})$ into a sequence of context vectors $C = (\mathbf{h}_1, \mathbf{h}_2, \cdots, \mathbf{h}_{T_x})$ whose size varies with respect to the source sentence length. Then, the encoder-decoder NMT decodes from the context vectors C and generates target translation $Y = (y_1, y_2, \cdots, y_{T_y})$ one word each time by maximizing the probability of $p(y_i|y_{<i}, C)$. Note that x_j (y_i) is word embedding corresponding to the j_{th} (i_{th}) word in the source (target) sentence. Next, we briefly review the encoder introducing how to obtain C and the decoder addressing how to calculate $p(y_i|y_{<i}, C)$.

Encoder: The context vectors C are generated by the encoder using a pair of recurrent neural networks (RNN) which consists of a forward RNN and a backward RNN. The forward RNN operates left-to-right over the source sentence from the first word, resulting in the forward context vectors $C_f = (\overrightarrow{h}_1, \overrightarrow{h}_2, \cdots, \overrightarrow{h}_{T_x})$, in which

$$\overrightarrow{h}_j = RNN(\overrightarrow{h}_{j-1}, x_j) \quad (1)$$

$\overleftarrow{h}_j$ can be calculated similarly.

RNN can be a Gated Recurrent Unit (GRU) (Cho et al., 2014) or a Long Short-Term Memory Unit (LSTM) (Hochreiter and Schmidhuber, 1997). At each position j of the source sentence, the context vector $\mathbf{h}_j$ is defined as the concatenation of the forward and backward context vectors.

Decoder: The conditional probability $p(y_i|y_{<i}, C)$ is computed in different ways according to the choice of the context C at time i. In (Cho et al., 2014), the authors choose $C = \mathbf{h}_{T_x}$, while Bahdanau et al. (2014) use different context c_i at different time step and the conditional probability will become:

$$p(y_i|y_{<i}, C) = p(y_i|y_{<i}, c_i) = g(y_{i-1}, z_i, c_i) \quad (2)$$

where z_i is the i_{th} hidden state of the decoder and is calculated conditioning on the previous hidden state z_{i-1}, previous output y_{i-1} and and the source context vector c_i at time i:

$$z_i = RNN(z_{i-1}, y_{i-1}, c_i) \quad (3)$$

In attention-based NMT, c_i is computed as the weighted sum of the source-side context vectors, just as illustrated in the top half of Fig. 1.

All the parameters of the encoder-decoder NMT are optimized to maximize the following conditional log-likelihood of the ***sentence aligned bilingual data***:

$$\mathcal{L}(\theta) = \frac{1}{N}\sum_{n=1}^{N}\sum_{i=1}^{T_y} logp(y_i^{(n)}|y_{<i}^{(n)}, X^{(n)}, \theta) \quad (4)$$

3 Incorporating Source-side Monolingual Data in NMT

We can see from the above objective function that all the network parameters are only optimized on the sentence aligned parallel corpus. It is well known that more related data of high quality leads to better and more robust network models. However, bilingual data is scarce in many languages (or domains). It becomes a key issue how to improve the encoder and decoder networks using other data besides the parallel sentence pairs. Gulcehre et al. (2015) and Sennrich et al. (2015) have tried to fine-tune the decoder neural network with target-side large-scale monolingual data and they report remarkable performance improvement with the enhanced decoder. In contrast, we believe that the encoder part of NMT can also be greatly strengthened with the source-side monolingual data.

To investigate fully the source-side monolingual data in improving the encoder network of NMT, we propose two approaches: the first one employs the self-learning algorithm to provide synthetic parallel data in which the target part is obtained through automatically translating the source-side monolingual data, which we refer to as *self-learning method*. The second one applies the multi-task learning framework that consists of two NMTs sharing the same encoder network to simultaneously train one NMT model on bilingual data and the other sentence reordering NMT model[2] on source-side monolingual data, which we refer to as *sentence reordering method*.

3.1 Self-learning Method

Given the sentence aligned bitext $\mathcal{D}_b = \{(X_b^{(n)}, Y_b^{(n)})\}_{n=1}^{N}$ in which N is not big enough, we have the source-side large-scale monolingual data $\mathcal{D}_{sm} = \{X_{sm}^{m}\}_{m=1}^{M}$ which is related to the bitext and $M \gg N$.

Our goal is to generate much more bilingual data using $\mathcal{D}_b$ and $\mathcal{D}_{sm}$. From the view of machine learning, we are equipped with some labelled data $\mathcal{D}_b$ and plenty of unlabelled data $\mathcal{D}_{sm}$, and we aim to obtain more labelled data for training better models. Self-learning is a simple but effective algorithm to tackle this issue. It first establishes a baseline with labelled data and then adopts the baseline to predict the labels of the unlabelled data. Finally, the unlabelled data together with the predicted labels become new labelled data.

In our scenario, the self-learning algorithm perform the following three steps . First, a baseline machine translation (MT) system (can use any translation model, SMT or NMT) is built with the given bilingual data $\mathcal{D}_b$. Second, the baseline MT sys-

[2]NMT is essentially a sequence-to-sequence prediction model. In most cases, the input sequence is different from the output sequence. In the sentence reordering NMT, we require that output sequence to be the reordered input sentences which are close to English word order.

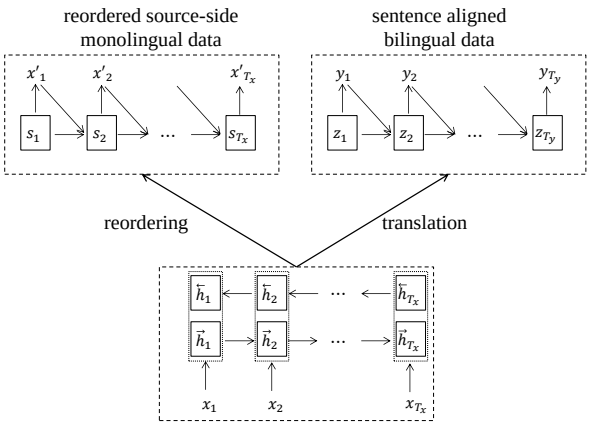

Figure 2: Multi-task learning framework to use source-side monolingual data in NMT, which includes a translation model and a sentence reordering model.

tem automatically translates the source-side monolingual sentences $\mathcal{D}_{sm}$ into target translations $\mathcal{D}_{tt} = \{(Y_{tt}^m)\}_{m=1}^M$, and further pairs $\mathcal{D}_{sm}$ with $\mathcal{D}_{tt}$ resulting in the synthetic parallel corpus $\mathcal{D}_{syn} = \{(X_{sm}^m, Y_{tt}^m)\}_{m=1}^M$. Third, the synthetic parallel corpus $\mathcal{D}_{syn}$ plus the original bitext $\mathcal{D}_b$ are combined together to train the new NMT model.

In principle, we can apply any MT system as the baseline to generate the synthetic bilingual data. In accordance with the translation model we focus on in this work, we employ NMT as the baseline MT system. Note that the synthetic target parts may negatively influence the decoder model of NMT. To address this problem, we can distinguish original bitext from the synthetic bilingual sentences during NMT training by freezing the parameters of the decoder network for the synthetic data.

It is worthy to discuss why self-learning algorithm can improve the encoder model of NMT. Even though we require $\mathcal{D}_{sm}$ to share the same source language vocabulary as $\mathcal{D}_b$ and no new word translations can be generated, the source-side monolingual data provides much more permutations of words in the vocabulary. Our RNN encoder network model will be optimized to well explain all of the word permutations. Thus, the encoder model of NMT can be enhanced for better generalization.

3.2 Sentence Reordering Method

The self-learning algorithm needs to translate first the large-scale source-side monolingual data. A natural question arises that whether can we improve the encoder model of NMT using just source-side monolingual corpora rather than the synthetic parallel data. Luong et al. (2015a) attempt to leverage source-side monolingual data in NMT using a simple autoencoder and skip-thought vectors. However, no promising results are reported. We believe that the reason lies in two aspects: 1) the large-scale monolingual data is not carefully selected; and 2) the adopted model is relatively simple. In this work, we propose to apply the multi-task learning method which designs a parameter sharing neural network framework to perform two tasks: machine translation and source sentence reordering. Fig.2 illustrates the overview of our framework for source-side monolingual data usage.

As shown in Fig. 2, our framework consists of two neural networks that shares the same encoder model but employs two different decoder models for machine translation and sentence reordering respectively. For the machine translation task trained on the sentence aligned parallel data $\mathcal{D}_b$, the network parameters are optimized to maximize the conditional probability of the target sentence $Y_b^{(n)}$ given a source sentence $X_b^{(n)}$, namely argmax$p(Y_b^{(n)}|X_b^{(n)})$.

As for the sentence reordering task trained on source-side monolingual data $\mathcal{D}_{sm}$, we regard it as a special machine translation task in which the target output is just the reordered source sentence, $Y_{sm}^{(m)} = X_{sm}^{'(m)}$. $X_{sm}^{'(m)}$ is obtained from $X_{sm}^{(m)}$ by using the pre-ordering rules proposed by (Wang et al., 2007), which can permutate the words of the source sentence so as to approximate the target language word order[3]. In this way, the sentence reordering NMT is more powerful than an autoencoder. Using the NMT paradigm, the shared encoder network is leveraged to learn the deep representation $C_{sm}^{(n)}$ of the source sentence $X_{sm}^{(n)}$, and the decoder network is employed to predict the reordered source sentence from the deep representation $C_{sm}^{(n)}$ (here $X_{sm}^{(n)} \in \mathcal{D}_{sm}$) by maximizing $p(X_{sm}^{'(n)}|X_{sm}^{(n)})$. Note that the above two

[3]The pre-ordering rules are obtained from the parsed source trees which heavily depend on the accuracy and efficiency of the parser. In fact, it takes us lots of time (even longer than synthetic parallel data generation) to parse all the source-side monolingual data. In the future, we attempt to design a more efficient pre-ordering method relying only on the bilingual training data.

tasks share the same encoder model to obtain the encoding of the source sentences. Accordingly, the overall objective function of this multi-task learning is the summation of log probabilities of machine translation and sentence reordering:

$$\mathcal{L}(\theta) = \frac{1}{N}\sum_{n=1}^{N}\sum_{i=1}^{T_y} logp(y_i^{(n)}|y_{<i}^{(n)}, X^{(n)}, \theta) + \frac{1}{M}\sum_{m=1}^{M}\sum_{i=1}^{T_X} logp(X_i^{'(m)}|X_{<i}^{'(m)}, X^{(m)}, \theta) \quad (5)$$

where $(\theta = \theta_{enc}, \theta_{dec_T}, \theta_{dec_R})$. θ_{enc} is the parameter collection of source language encoder network, θdec_T denotes the parameter set of the decoder network for translation, and θ_{dec_R} represents the parameters of the decoder network for sentence reordering.

Intuitively, the sentence reordering task is easier than the translation task. Furthermore, in this paper, we pay much more attention on the translation task compared to the sentence reordering task. Considering these, we distinguish these two tasks during the parameter optimization process. It is performed using an alternate iteration strategy. For each iteration, we first optimize the encoder-decoder network parameters in the reordering task for one epoch. The learnt encoder network parameters are employed to initialize the encoder model for the translation task. Then, we learn the encoder-decoder network parameters in the translation task for several epochs[4]. The new encoder parameters are then used to initialize the encoder model for the reordering task. We continue the iteration until the constraint (e.g. iteration number or no parameter change) is satisfied. The weakness is that this method is less efficient than the self-learning approach.

4 Experimental Settings

In this section we describe the data set used in our experiments, data preprocessing, the training and evaluation details, and all the translation methods we compare in experiments.

[4]We rune four epochs for the translation task in each iteration.

4.1 Dataset

We perform two tasks on Chinese-to-English translation: one for small data set and the other for large-scale data set. Our small training data includes 0.63M sentence pairs (after data cleaning) extracted from LDC corpora[5]. The large-scale data set contains about 2.1M sentence pairs including the small training data. For validation, we choose NIST 2003 (MT03) dataset. For testing, we use NIST 2004 (MT04), NIST 2005 (MT05) and NIST 2006 (MT06) datasets. As for the source-side monolingual data, we collect about 20M Chinese sentences from LDC and we retain the sentences in which more than 50% words should appear in the source-side portion of the bilingual training data, resulting in 6.5M monolingual sentences for small training data set (12M for large-scale training data set) ordered by the word hit rate.

4.2 Data Preprocessing

We apply word-level translation in experiments. The Chinese sentences are word segmented using Stanford Word Segmenter[6]. To pre-order the Chinese sentences using the syntax-based reordering method proposed by (Wang et al., 2007), we utilize the Berkeley parser (Petrov et al., 2006). The English sentences are tokenized using the tokenizer script from the Moses decoder[7]. To speed up the training procedure, we clean the training data and remove all the sentences of length over 50 words. We limit the vocabulary in both Chinese and English to the most 40K words and all the out-of-vocabulary words are replaced with *UNK*.

4.3 Training and Evaluation Details

Each NMT model is trained on GPU K40 using stochastic gradient decent algorithm AdaGrad (Duchi et al., 2011). We use mini batch size of 32. The word embedding dimension of source and target language is 500 and the size of hidden layer is set to 1024. The training time for each model ranges from 5 days to 10 days for small training data set and ranges from 8 days to 15 days for large training data

[5]LDC2000T50, LDC2002L27, LDC2002T01, LDC2002E18, LDC2003E07, LDC2003E14, LDC2003T17, LDC2004T07.

[6]http://nlp.stanford.edu/software/segmenter.shtml

[7]http://www.statmt.org/moses/

Method	MT03	MT04	MT05	MT06
Moses	30.30	31.04	28.19	30.04
RNNSearch	28.38	30.85	26.78	29.27
RNNSearch-Mono-SL (25%)	29.65	31.92	28.65	29.86
RNNSearch-Mono-SL (50%)	**32.43**	**33.16**	**30.43**	**32.35**
RNNSearch-Mono-SL (75%)	30.24	31.18	29.33	28.82
RNNSearch-Mono-SL (100%)	29.97	30.78	26.45	28.06
RNNSearch-Mono-MTL (25%)	31.68	32.51	29.8	31.29
RNNSearch-Mono-MTL (50%)	**33.38**	**34.30**	**31.57**	**33.40**
RNNSearch-Mono-MTL (75%)	31.69	32.83	28.17	30.26
RNNSearch-Mono-MTL (100%)	30.31	30.62	27.23	28.85
RNNSearch-Mono-Autoencoder (50%)	31.55	32.07	28.19	30.85
RNNSearch-Mono-Autoencoder (100%)	27.81	30.32	25.84	27.73

Table 1: Translation results (BLEU score) for different translation methods. For our methods exploring the source-side monolingual data, we investigate the performance change as we choose different scales of monolingual data (e.g. from top 25% to 100% according to the word coverage of the monolingual sentence in source language vocabulary of bilingual training corpus).

set[8]. We use case-insensitive 4-gram BLEU score as the evaluation metric (Papineni et al., 2002).

4.4 Translation Methods

In the experiments, we compare our method with conventional SMT model and a strong NMT model. We list all the translation methods as follows:

- **Moses**: It is the state-of-the-art phrase-based SMT system (Koehn et al., 2007). We use its default configuration and train a 4-gram language model on the target portion of the bilingual training data.
- **RNNSearch**: It is an attention-based NMT system (Bahdanau et al., 2014).
- **RNNSearch-Mono-SL**: It is our NMT system which makes use of the source-side large-scale monolingual data by applying the self-learning algorithm.
- **RNNSearch-Mono-MTL**: It is our NMT system that exploits the source-side monolingual data by using our multi-task learning framework which performs machine translation and sentence reordering at the same time.
- **RNNSearch-Mono-Autoencoder**: It also applies the multi-task learning framework in which a simple autoencoder is adopted on source-side monolingual data (Luong et al., 2015a).

5 Translation Results on Small Data

For translation quality evaluation, we attempt to figure out four questions: 1) Can the source-side monolingual data improve the neural machine translation? 2) Could the improved NMT outperform the state-of-the-art phrase-based SMT? 3) Whether it is true that the more the source-side monolingual data the better the translation quality? 4) Which MT model is more suitable to incorporate source-side monolingual data: SMT or NMT?

5.1 Effects of Source-side Monolingual Data in NMT

Table 1 reports the translation quality for different methods. Comparing the first two lines in Table 1, it is obvious that the NMT method *RNNSearch* performs much worse than the SMT model *Moses* on Chinese-to-English translation. The gap is as large as approximately 2.0 BLEU points (28.38 vs. 30.30). We speculate that the encoder-decoder network models of NMT are not well optimized due to insufficient bilingual training data.

The focus of this work is to figure out whether

[8]It needs another 5 to 10 days when adding millions of monolingual data.

the encoder model of NMT can be improved using source-side monolingual data and further boost the translation quality. The four lines (3-6 in Table 1) show the BLEU scores when applying self-learning algorithm to incorporate the source-side monolingual data. Clearly, *RNNSearch-Mono-SL* outperforms *RNNSearch* in most cases. The best performance is obtained if the top 50% monolingual data is used. The biggest improvement is up to 4.05 BLEU points (32.43 vs. 28.38 on MT03) and it also significantly outperforms *Moses*.

When employing our multi-task learning framework to incorporate source-side monolingual data, the translation quality can be further improved (Lines 7-10 in Table 1). For example, *RNNSearch-Mono-MTL* using the top 50% monolingual data can remarkably outperform the baseline *RNNSearch*, with an improvement up to 5.0 BLEU points (33.38 vs. 28.38 on MT03). Moreover, it also performs significantly better than the state-of-the-art phrase-based SMT *Moses* by the largest gains of 3.38 BLEU points (31.57 vs. 28.19 on MT05). The promising results demonstrate that source-side monolingual data can improve neural machine translation and our multi-task learning is more effective.

From the last two lines in Table 1, we can see that *RNNSearch-Mono-Autoencoder* can also improve the translation quality by more than 1.0 BLEU points when using the most related monolingual data. However, it underperforms *RNNSearch-Mono-MTL* by a large gap. It indicates that sentence reordering model is better than sentence reconstruction model for exploiting the source-side monolingual data.

Note that we sort the source-side monolingual data according to the word coverage [9] in the bilingual training data. Sentences in the front have more shared words with the source-side vocabulary of bilingual training data. We can clearly see from Table 1 that monolingual data cannot always improve NMT. By adding closely related corpus (25% to 50%), the methods can achieve better and better performance. However, when adding more unrelated monolingual data (75% to 100%) which shares fewer and fewer words in common with the bilingual data, the translation quality becomes worse and worse, and even worse than the baseline *RNNSearch*. Both self-learning algorithm *RNNSearch-Mono-SL* and multi-task learning framework *RNNSearch-Mono-MTL* have the same trend. This indicates that only closely related source-side monolingual data can lead to performance improvement.

[9]In current work, the simple word coverage is applied to indicate the similarity. In the future, we plan to use phrase embedding (Zhang et al., 2014) or sentence embedding (Zhang et al., 2015; Wang et al., 2016a; Wang et al., 2016b) to select the relevant monolingual data.

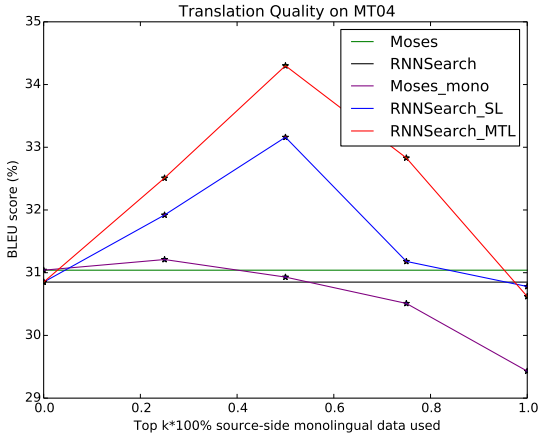

Figure 3: Effects of source-side monolingual data on MT04.

5.2 NMT vs. SMT on Using Source-side Monolingual Data

Although the proposed multi-task learning framework cannot fit SMT because of no shared deep information between the two tasks in SMT, self-learning algorithm can also be applied in SMT as done by (Ueffing et al., 2007; Wu et al., 2008). We may want to know whether NMT is more effective in using source-side monolingual data than SMT.

We apply the self-learning algorithm in SMT by incorporating top 25%, 50%, 75% and 100% synthetic sentence pairs to retrain baseline Moses. Fig. 3 shows the effect of source-side monolingual data in different methods on test set MT04. The figure reveals three similar phenomena. First, related monolingual data can boost the translation quality no matter whether NMT or SMT is used, but mixing more unrelated monolingual corpus will decrease the performance. Second, integrating closely related source-side monolingual data in NMT (*RNNSearch-SL* and *RNNSearch-MTL*) is much more effective than that in SMT (e.g. results for top 50%). It is because that SMT relies on the translation rules

Method	MT03	MT04	MT05	MT06
RNNSearch	35.18	36.20	33.21	32.86
RNNSearch-Mono-MTL (50%)	**36.32**	**37.51**	**35.08**	**34.26**
RNNSearch-Mono-MTL (100%)	35.75	36.74	34.23	33.52

Table 2: Translation results (BLEU score) for different translation methods in large-scale training data.

learnt from the bilingual training data and the synthetic parallel data is obtained by these rules, and thus the synthetic parallel data cannot generate much more information. In contrast, NMT provides a encoder-decoder mechanism and depends heavily on the source language semantic vector representations which facilitate the information sharing. Third, the translation quality changes much more dramatically in NMT methods than that in SMT. It indicates that the neural network models incline to be more affected by the quality of the training data.

6 Translation Results on Large-scale Data

A natural question arises that is the source-side monolingual data still very helpful when we have much more bilingual training data. We conduct the large-scale experiments using our proposed multi-task framework *RNNSearch-Mono-MTL*. Table 2 reports the results.

We can see from the table that closely related source-side monolingual data (the top 50%) can also boost the translation quality on all of the test sets. The performance improvement can be more than 1.0 BLEU points. Compared to the results on small training data, the gains from source-side monolingual data are much smaller. It is reasonable since large-scale training data can make the parameters of the encoder-decoder parameters much stable. We can also observe the similar phenomenon that adding more unrelated monolingual data leads to decreased translation quality.

7 Related Work

As a new paradigm for machine translation, the encoder-decoder based NMT has drawn more and more attention. Most of the existing methods mainly focus on designing better alignment mechanisms (attention model) for the decoder network (Cheng et al., 2016a; Luong et al., 2015b; Cohn et al., 2016; Feng et al., 2016; Tu et al., 2016; Mi et al., 2016a; Mi et al., 2016b), better objective functions for BLEU evaluation (Shen et al., 2016) and better strategies for handling unknown words (Luong et al., 2015c; Sennrich et al., 2015; Li et al., 2016) or large vocabularies (Jean et al., 2015; Mi et al., 2016c).

Our focus in this work is aiming to make full use of the source-side large-scale monolingual data in NMT, which is not fully explored before. The most related works lie in three aspects: 1) applying target-side monolingual data in NMT, 2) targeting knowledge sharing with multi-task NMT, and 3) using source-side monolingual data in conventional SMT and NMT.

Gulcehre et al. (2015) first investigate the target-side monolingual data in NMT. They propose shallow and deep fusion methods to enhance the decoder network by training a big language model on target-side large-scale monolingual data. Sennrich et al. (2015) further propose a new approach to use target-side monolingual data. They generate the synthetic bilingual data by translating the target monolingual sentences to source language sentences and retrain NMT with the mixture of original bilingual data and the synthetic parallel data. It is similar to our self-learning algorithm in which we concern the source-side monolingual data. Furthermore, their method requires to train an additional NMT from target language to source language, which may negatively influence the attention model in the decoder network.

Dong et al. (2015) propose a multi-task learning method for translating one source language into multiple target languages in NMT so that the encoder network can be shared when dealing with several sets of bilingual data. Zoph et al. (2016), Zoph and Knight (2016) and Firat et al. (2016) further deal with more complicated cases (e.g. multi-source languages). Note that all these methods require bilingual training corpus. Instead, we adapt the multi-task learning framework to better accommodate the source-side monolingual data.

Ueffing et al. (2007) and Wu et al. (2008) explore

the usage of source-side monolingual data in conventional SMT with a self-learning algorithm. Although we apply self-learning in this work, we use it to enhance the encoder network in NMT rather than generating more translation rules in SMT and we also adapt a multi-task learning framework to take full advantage of the source-side monolingual data. Luong et al. (2015a) also investigate the source-side monolingual data in the multi-task learning framework, in which a simple autoencoder or skip-thought vectors are employed to model the monolingual data. Our sentence reordering model is more powerful than simple autoencoder in encoder enhancement. Furthermore, they do not carefully prepare the monolingual data for which we show that only related monolingual data leads to big improvements.

In parallel to our work, Cheng et al. (2016b) propose a similar semi-supervised framework to handle both source and target language monolingual data. If source-side monolingual data is considered, a reconstruction framework including two NMTs is employed. One NMT translates the source-side monolingual data into target language translations, from which the other NMT attempts to reconstruct the original source-side monolingual data. In contrast to their approach, we propose a sentence reordering model rather than the sentence reconstruction model. Furthermore, we carefully investigate the relationship between the monolingual data quality and the translation performance improvement.

8 Conclusions and Future Work

In this paper, we propose a self-learning algorithm and a new multi-task learning framework to use source-side monolingual data so as to improve the encoder network of the encoder-decoder based NMT. The self-learning algorithm generates the synthetic parallel corpus and enlarge the bilingual training data to enhance the encoder model of NMT. The multi-task learning framework performs machine translation on bilingual data and sentence reordering on source-side monolingual data by sharing the same encoder network. The experiments show that our method can significantly outperform the strong attention-based NMT baseline, and the proposed multi-task learning framework performs better than the self-learning algorithm at the expense of low efficiency. Furthermore, the experiments also demonstrate that NMT is more effective for incorporating the source-side monolingual data than conventional SMT. We also observe that more monolingual data does not always improve the translation quality and only relevant data does help.

In the future, we would like to design smarter mechanisms to distinguish real data from synthetic data in self-learning algorithm, and attempt to propose better models for handling source-side monolingual data. We also plan to apply our methods in other languages, especially for low-resource languages.

Acknowledgments

We thank the reviewers for their valuable comments and suggestions. This research work has been partially funded by the Natural Science Foundation of China under Grant No. 61333018, No. 91520204 and No. 61303181, and supported by the Strategic Priority Research Program of the CAS (Grant XDB02070007).

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Yong Cheng, Shiqi Shen, Zhongjun He, Wei He, Hua Wu, Maosong Sun, and Yang Liu. 2016a. Agreement-based joint training for bidirectional attention-based neural machine translation. In *Proceedings of AAAI 2016*.

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

David Chiang. 2007. Hierarchical phrase-based translation. *computational linguistics*, 33(2):201–228.

Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder-decoder for statistical machine translation. In *Proceedings of EMNLP 2014*.

Trevor Cohn, Cong Duy Vu Hoang, Ekaterina Vymolova, Kaisheng Yao, Chris Dyer, and Gholamreza Haffari. 2016. Incorporating structural alignment biases into an attentional neural translation model. In *Proceedings of NAACL 2016*.

Daxiang Dong, Hua Wu, Wei He, Dianhai Yu, and Haifeng Wang. 2015. Multi-task learning for multiple language translation. In *Proceedings of ACL 2015*.

John Duchi, Elad Hazan, and Yoram Singer. 2011. Adaptive subgradient methods for online learning and stochastic optimization. *The Journal of Machine Learning Research*, 12:2121–2159.

Shi Feng, Shujie Liu, Mu Li, and Ming Zhou. 2016. Implicit distortion and fertility models for attention-based encoder-decoder nmt model. *arXiv preprint arXiv:1601.03317*.

Orhan Firat, Kyunghyun Cho, and Yoshua Bengio. 2016. Multi-way, multilingual neural machine translation with a shared attention mechanism. *arXiv preprint arXiv:1601.01073*.

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 preprint arXiv:1503.03535*.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Sebastien Jean, Kyunghyun Cho, Roland Memisevic, and Yoshua Bengio. 2015. On using very large target vocabulary for neural machine translation. In *Proceedings of ACL 2015*.

Nal Kalchbrenner and Phil Blunsom. 2013. Recurrent continuous translation models. In *Proceedings of EMNLP 2013*.

Ryan Kiros, Yukun Zhu, Ruslan Salakhutdinov, Richard S. Zemel, Antonio Torralba, Raquel Urtasun, and Sanja Fidler. 2015. Skip-thought vectors. In *Proceedings of NIPS 2015*.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, et al. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of ACL 2007*, pages 177–180.

Xiaoqing Li, Jiajun Zhang, and Chengqing Zong. 2016. Towards zero unknown word in neural machine translation. In *Proceedings of IJCAI 2016*.

Minh-Thang Luong, Quoc V Le, Ilya Sutskever, Oriol Vinyals, and Lukasz Kaiser. 2015a. Multi-task sequence to sequence learning. *arXiv preprint arXiv:1511.06114*.

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

Minh-Thang Luong, Ilya Sutskever, Quoc V Le, Oriol Vinyals, and Wojciech Zaremba. 2015c. Addressing the rare word problem in neural machine translation. In *Proceedings of ACL 2015*.

Haitao Mi, Baskaran Sankaran, Zhiguo Wang, and Abe Ittycheriah. 2016a. A coverage embedding model for neural machine translation. In *Proceedings of EMNLP 2016*.

Haitao Mi, Zhiguo Wang, Niyu Ge, and Abe Ittycheriah. 2016b. Supervised attentions for neural machine translation. In *Proceedings of EMNLP 2016*.

Haitao Mi, Zhiguo Wang, and Abe Ittycheriah. 2016c. Vocabulary manipulation for large vocabulary neural machine translation. In *Proceedings of ACL 2016*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of ACL 2002*, pages 311–318.

Slav Petrov, Leon Barrett, Romain Thibaux, and Dan Klein. 2006. Learning accurate, compact, and interpretable tree annotation. In *Proceedings of COLING-ACL 2006*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2015. Improving neural machine translation models with monolingual data. *arXiv preprint arXiv:1511.06709*.

Shiqi Shen, Yong Cheng, Zhongjun He, Wei He, Hua Wu, Maosong Sun, and Yang Liu. 2016. Minimum risk training for neural machine translation. In *Proceedings of ACL 2016*.

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

Zhaopeng Tu, Zhengdong Lu, Yang Liu, Xiaohua Liu, and Hang Li. 2016. Coverage-based neural machine translation. In *Proceedings of ACL 2016*.

Nicola Ueffing, Gholamreza Haffari, and Anoop Sarkar. 2007. Transductive learning for statistical machine translation. In *Proceedings of ACL 2007*.

Chao Wang, Michael Collins, and Philipp Koehn. 2007. Chinese syntactic reordering for statistical machine translation. In *Proceedings of EMNLP 2007*.

Zhiguo Wang, Haitao Mi, and Abe Ittycheriah. 2016a. Semi-supervised clustering for short text via deep representation learning. In *Proceedings of CoNLL 2016*.

Zhiguo Wang, Haitao Mi, and Abe Ittycheriah. 2016b. Sentence similarity learning by lexical decomposition and composition. *arXiv preprint arXiv:1602.07019*.

Hua Wu, Haifeng Wang, and Chengqing Zong. 2008. Domain adaptation for statistical machine translation with domain dictionary and monolingual corpora. In *Proceedings of COLING 2008*, pages 993–1000.

Jiajun Zhang, Shujie Liu, Mu Li, Ming Zhou, and Chengqing Zong. 2014. Bilingually-constrained phrase embeddings for machine translation. In *Proceedings of ACL 2014*.

Jiajun Zhang, Dakun Zhang, and Jie Hao. 2015. Local translation prediction with global sentence representation. In *Proceedings of IJCAI 2015*.

Barret Zoph and Keven Knight. 2016. Transfer learning for low-resource neural machine translation. *arXiv preprint arXiv:1604.02201v1*.

Barret Zoph, Deniz Yuret, Jonathan May, and Kevin Knight. 2016. Multi-source neural translation. In *Proceedings of NAACL 2016*.