Local Translation Prediction with Global Sentence Representation

Jiajun Zhang
National Laboratory of Pattern Recognition, CASIA, Beijing, P.R. China
jjzhang@nlpr.ia.ac.cn

Abstract

Statistical machine translation models have made great progress in improving the translation quality. However, the existing models predict the target translation with only the source- and target-side local context information. In practice, distinguishing good translations from bad ones does not only depend on the local features, but also rely on the global sentence-level information. In this paper, we explore the source-side global sentence-level features for target-side local translation prediction. We propose a novel bilingually-constrained chunk-based convolutional neural network to learn sentence semantic representations. With the sentence-level feature representation, we further design a feed-forward neural network to better predict translations using both local and global information. The large-scale experiments show that our method can obtain substantial improvements in translation quality over the strong baseline: the hierarchical phrase-based translation model augmented with the neural network joint model.

1 Introduction

In the recent years, statistical machine translation (SMT) models, such as phrase-based models [Koehn *et al.*, 2007], hierarchical phrase-based models [Chiang, 2007], and linguistically syntax-based models [Liu *et al.*, 2006; Galley *et al.*, 2006], have achieved great progress in improving the translation performance. In these translation models, the target sentence is generated by compositing several local translations with reordering models or synchronous grammars, and the local translations are rendered with the help of the source- and target-side local context information. In most cases, the translation of source language words can be determined with local context features. However, there are many cases in which the target translation does not only depend on the local context, but also rely on the global sentence-level information.

Take the Chinese sentence and its English translation in Figure 1 as an example. For the Chinese word in red color, *molecule* is the most possible translation. Even with the help of the local context information, it cannot figure out the correct translation. Given the sentence-level semantics talking about *preventing illegal immigrants and unsafe people from entering Japan*, we can make sure that the best target translation for the Chinese word in red color should be *people*. Obviously, the global sentence-level semantic information plays an important role in local translation prediction.

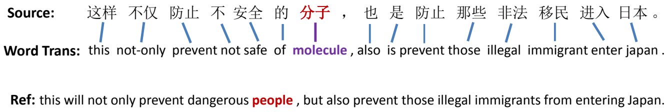
Source: 这样 不仅 防止 不 安全 的 分子 ， 也 是 防止 那些 非法 移民 进入 日本 。
Word Trans: this not-only prevent not safe of molecule , also is prevent those illegal immigrant enter japan .
Ref: this will not only prevent dangerous people , but also prevent those illegal immigrants from entering Japan.

Figure 1: An example for Chinese to English translation in which the translation of the Chinese word in red needs the global sentence information of the Chinese sentence.

Two questions arise: 1) how to represent the global sentence-level semantics? 2) how to make full use of the sentence semantic representation in statistical machine translation models?

For the first problem, the neural network methods are proposed recently to learn sentence representations. These methods include recurrent neural networks [Kalchbrenner and Blunsom, 2013; Sutskever *et al.*, 2014; Bahdanau *et al.*, 2014], recursive neural networks [Socher *et al.*, 2011; Socher *et al.*, 2013a; Socher *et al.*, 2013b], sentence to vector [Le and Mikolov, 2014] and convolutional neural networks [Kim, 2014; Kalchbrenner *et al.*, 2014; Hu *et al.*, 2014; Zeng *et al.*, 2014]. It should be noted that most of the above approaches learn the distributed sentence representations for specific tasks, such as classification, sequence labelling, and structure prediction. The semantic meaning of the sentence is not fully contained by the sentence representation. In this paper, we focus on learning the sentence semantic representation. Although we have no gold sentence semantic representation available for supervised learning, we have a large amount of parallel sentence pairs which share the same semantic meaning. Accordingly, sentence translation equivalents can supervise each other to learn their semantic representations. Furthermore, we design the chunk-based convolutional neural network in order to well handle the sentence length variation and retain as much information as possible at the same time. In this network, we just need to decide how many chunks we want to segment a sentence. Therefore, combining the two ideas together, we propose the bilingually-constrained chunk-based convolutional neural network (BC-

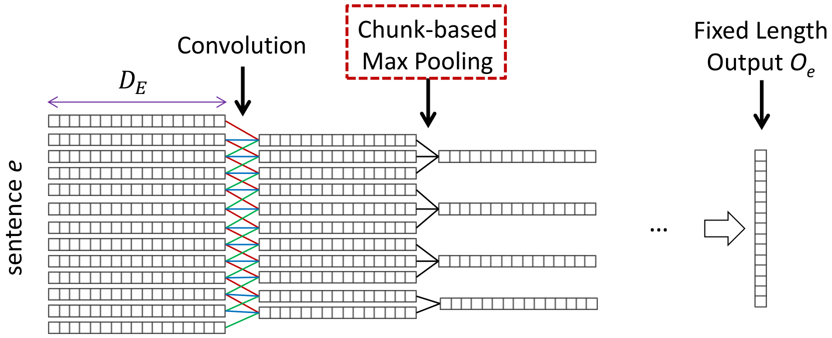

Figure 2: Chunk-based convolutional neural network architecture for modelling sentence representation.

CNN) for sentence semantic representation.

For the second problem, we incorporate the sentence semantic representation during the decoding process to better generate the target translation. Following the idea in [Devlin *et al.*, 2014], we design a feed-forward neural network which takes the learnt sentence semantic representation as the new input feature to predict the conditional probability of the target word given both the local and global information. As an informative feature, this conditional probability is integrated into the log-linear model of the hierarchical phrase-based translation system [Chiang, 2007].

In this paper, we make the following contributions:

- Our idea of the bilingually-constrained method circumvents the problem of lacking gold labelled data and provides a good way to learn sentence semantic representations.
- To deal with the variable length of the sentences and meanwhile retain as much semantics as possible, we propose the chunk-based convolutional neural network in which we can choose the number of the chunks.
- When integrating the sentence semantic representation into the decoding process, we can achieve significant improvements in translation quality over a strong baseline.

2 Sentence Semantic Representation

Convolutional neural network (CNN) consisting of the convolution and pooling layers provides a standard architecture [Collobert *et al.*, 2011] which maps variable-length sentences into fixed-length distributed vectors. This section starts with introducing a new variant of the standard CNN called chunk-based CNN in order to keep more semantics of the sentence.

2.1 Chunk-based Convolutional Neural Network

The model architecture is illustrated in Figure 2. The data flow is similar to the standard CNN: the model takes as input the sequence of word embeddings in a sentence, summarizes the sentence meaning by convolving the sliding window and pooling the saliency through the sentence, and yields the fixed-length distributed vector with other layers, such as dropout layer, more convolution and pooling layers, linear and non-linear layers.

Specifically, assuming we are equipped with a word embedding matrix $\mathbb{L} \in \mathbb{R}^{k\times|V|}$ [1] trained on the large-scale monolingual data using unsupervised algorithm (e.g. word2vec [Mikolov *et al.*, 2013]). Given a sentence $w_1 w_2 \cdots w_n$, each word w_i is first projected into a vector X_i through the word embedding matrix. Then, we concatenate all the vectors in order to form the input of the model:

$$X = [X_1, X_2, \cdots, X_n] \tag{1}$$

Convolution Layer involves a number of *filters* $W \in \mathbb{R}^{h\times k}$ which summarize the information of h-word window and produce a new feature. For the window of h words $X_{i:i+h-1}$, a *filter* F_l ($1 \le l \le L$, and L denotes the number of filters) generates the feature y_i^l as follows:

$$y_i^l = \sigma(W \cdot X_{i:i+h-1} + b) \tag{2}$$

Where σ is a non-linear activation function (e.g. Relu or Sigmoid), and b is a bias term. When a *filter* traverses each window in the sentence from $X_{1:h-1}$ to $X_{n-h+1:n}$, we get the output of the feature map corresponding to the *filter* F_l:

$$y^l = [y_1^l, y_2^l, \cdots, y_{n-h+1}^l] \tag{3}$$

Here, $y^l \in \mathbb{R}^{n-h+1}$. It should be noted that the sentences differ with each other in length n (from several words to more than 100 words), and then y^l has different dimensions for different sentences. It becomes a key question how to transform the variable-length vector y^l into a fixed-length vector.

Pooling Layer is designed to perform this task. At most cases, we apply a standard *max-over-time* pooling operation [Collobert *et al.*, 2011; Kim, 2014] over y^l and choose the maximum value $\hat{y}^l = max\{y^l\}$ as the most important feature of the filter F_l. This idea is simple and easy to implement. However, its disadvantage lies in two-fold: 1) most of the information in the sentence is lost, and 2) the word order information is missing. Kalchbrenner *et al.* [2014] proposed to take the top-K maximum values over y^l to keep more information, but the word order information is still missing. Hu *et al.* [2014] designed a max-pooling over every two-unit, but they require the fixed-length inputs. In this paper, we design a simple but effective **chunk-based max-pooling** so as to retain more semantics of the sentence and keep the order information as well.

[1] k is the embedding dimension and $|V|$ is the vocabulary size.

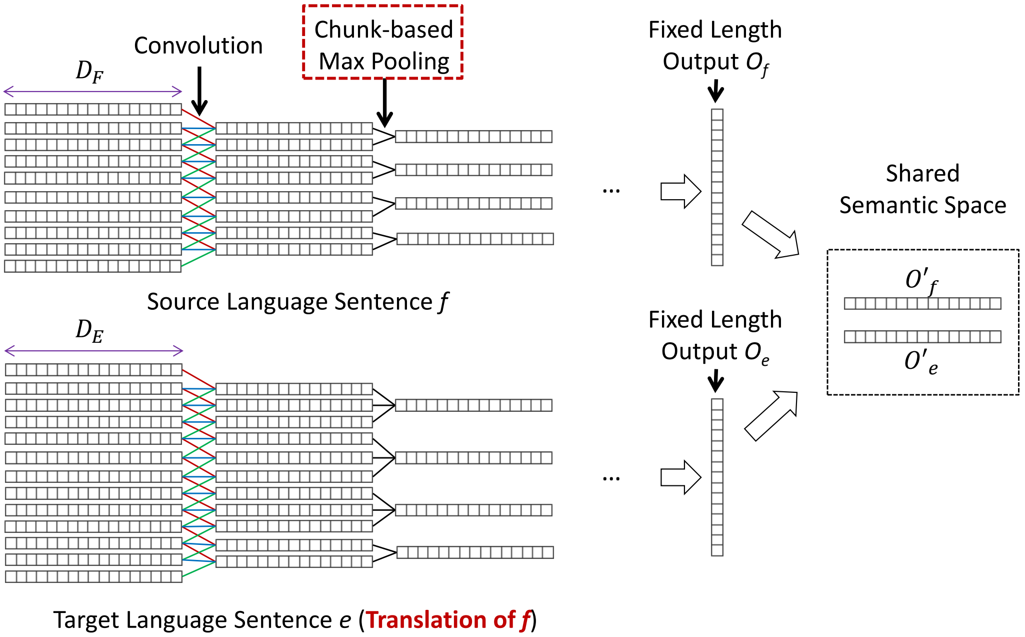

Figure 3: Bilingually-constrained Chunk-based convolutional neural network architecture for learning sentence semantic representation.

Given the predefined number of chunks C (e.g. $C = 4$ in Figure 2), we first divide evenly y^l into C segments, and then takes the maximum value from each segment. Note that $n-h+1$ does not have to be divisible by C and the last chunk can have the size of the modulus. Then, we can transform the variable-length vector y^l into a fixed-length vector with C values:

$$\begin{aligned} y^{lC} &= chunkMax\{y^l\} \\ &= chunkMax\{[y_1^l, y_2^l, \cdots, y_{n-h+1}^l]\} \\ &= [y_{c_1}^l, y_{c_2}^l, \cdots, y_C^l] \end{aligned} \tag{4}$$

If there are F_L *filters*, the output of the pooling layer will be a vector in $F_L \times C$ dimension. In our model, the pooling layer is followed by a dropout layer, two fully connected linear layers with non-linear activation Relu. Finally, we can obtain a fixed-length output vector for each sentence.

This idea of chunk-based CNN is inspired by the inherent structure of a sentence. From the perspective of shallow structures, a sentence is organized by subject-verb-object (English and Chinese word order). Then, we can assign C with a small number (e.g 2 or 3) to summarize this kind of semantic information. From the perspective of deep structures, a sentence can be described as a sequence of noun phrase (NP), verb phrase (VP), adjective phrase (ADJP), prepositional phrase (PP) and so on. Accordingly, we can set C to be a relatively large number to capture these kinds of information. Therefore, the chunk-based CNN can retain more semantics of the sentence and maintain as well the sentence structure to some extent.

2.2 Bilingually-constrained Chunk-based CNN

The convolutional neural network is usually tuned to optimize an objective function for a specific task, such as sentiment and relation classification [Kim, 2014; Zeng *et al.*, 2014]. The result sentence representation is class sensitive but does not encode adequate semantic meaning of the sentence. Since our goal is to learn sentence semantic representations, we need to find a well-defined objective function.

However, there are no gold sentence semantic representations available in the real world. Fortunately, we know the fact that if two sentences share the same meaning, their semantic representations should be identical. As we know in machine translation that, there are lots of parallel sentence pairs for different languages, such as Chinese-English, Arabic-English. Thus, we can make an inference from this fact that if a model can learn the same representation for any parallel sentence pair sharing the same meaning, the learnt representation must encode the semantics of the sentences and the corresponding model is our desire. Inspired by the work on word and phrase embeddings using a bilingual method [Zou *et al.*, 2013; Zhang *et al.*, 2014a], we propose the Bilingually-constrained Chunk-based CNN (BC-CNN), whose basic goal is to minimize the semantic distance between the sentences and their translations.

As illustrated in Figure 3, given a source language sentence f and its translation e, the chunk-based CNN can generate respectively the fixed-length output vectors O_f and O_e. Then, O_f and O_e are projected into a shared semantic space, and become O'_f and O'_e (e.g. $O'_f = \sigma(W^t \cdot O_f + b^t)$, where W^t denotes transformation matrix). In the shared space, two representation vectors can calculate their distance easily with dot-product. The basic objective function is to minimize the distance $dis(f, e; \Theta) = dis(O'_f, O'_e)$ between O'_f and O'_e.

We know that a good model should not only make the representations of translation equivalents as similar as possible, but also should enforce the representations of non-translation pairs as different as possible. Therefore, our objective function is also designed to maximize the distance $dis(f, e^*; \Theta)$ if (f, e^*) is a non-translation pair. Then, we design our objective function to be a max-margin loss:

$$j(f, e, e^*; \Theta) = max(0, 1 + dis(f, e^*; \Theta) - dis(f, e; \Theta)) \tag{5}$$

Here, Θ includes all the parameters of the bilingually-constrained chunk-based CNN. For any translation equivalent (f, e), we can choose randomly a sentence e^* ($e^* \neq e$) from the target language monolingual data and obtain the non-translation pair (f, e^*). The finally objective function over the large-scale parallel sentence pairs (F, E) of size N will be:

$$J(F, E; \Theta) = \frac{1}{N} \sum_{(f,f) \in (F,E)} j(f, e, e^*; \Theta) + \frac{\lambda}{2} ||\Theta||^2 \tag{6}$$

3 Integrating Sentence Semantic Representations in Translation Models

With the trained chunk-based convolutional neural network, we can obtain the semantic representation for any sentence. In this section, we introduce how to make full use of the sentence semantic representations in statistical machine translation models.

3.1 Sentence Representation for Translation Probability Estimation

Formally, given a source sentence s, machine translation aims to find from the search space T the best target translation hypothesis t which has the highest conditional probability $p(t|s)$. If we focus on each target word, the conditional probability can be decomposed as follows:

$$p(t|s) = \prod_{i=1}^{|t|} p(t_i | t_1, t_2, \cdots, t_{i-1}, s) \tag{7}$$

Devlin *et al.* [2014] approximated the target single word probability $p(t_i | t_1, t_2, \cdots, t_{i-1}, s)$ following the target n-gram language model and using the source-side local context, called joint model:

$$p(t|s) \approx \prod_{i=1}^{|t|} p(t_i | t_{i-n+1}, \cdots, t_{i-1}, \mathcal{S}_i) \tag{8}$$

In which $\mathcal{S}_i$ includes the source-side local context associated with the current target word t_i. We know that machine translation models, such as hierarchical phrase-based model, generate the target hypothesis with translation rules [2] from which we can find the source word s_{a_i} that is aligned to t_i (s_{a_i} and t_i are translations with each other). Devlin *et al.* [2014] take the m-word source-side local context centered at s_{a_i}:

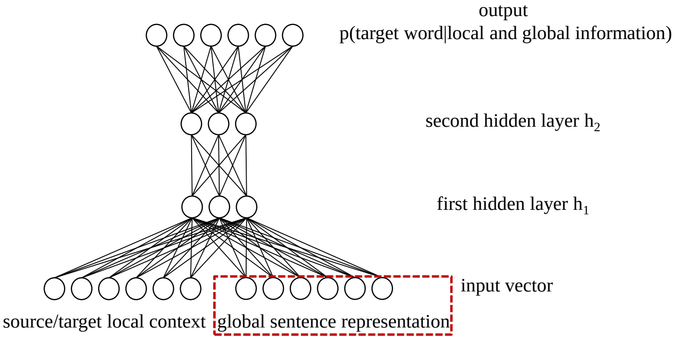
output
p(target word|local and global information)
second hidden layer h$_2$
first hidden layer h$_1$
input vector
source/target local context
global sentence representation

Figure 4: Local translation prediction with both the local context and the global sentence semantic representations.

$$\mathcal{S}_i = s_{a_i - \frac{m-1}{2}}, \cdots, s_{a_i}, s_{a_i + \frac{m-1}{2}} \tag{9}$$

However, just as we discussed in the Introduction section that besides the local context, the global sentence semantic information plays an indispensable role in accurate translation prediction. Therefore, we augment Equation 8 with the global sentence semantics:

$$p(t|s) \approx \prod_{i=1}^{|t|} p(t_i | t_{i-n+1}, \cdots, t_{i-1}, \mathcal{S}_i, \mathbf{s}) \tag{10}$$

In this way, translating every target word during decoding is aware of the source-side global sentence-level information.

In our experiments, following [Devlin *et al.*, 2014] we use $n = 4$ and $m = 11$. It is easy to see that the data sparsity will become a serious problem if we employ the traditional method to perform the probability estimation. Therefore, we resort to the neural networks that perform the probability estimation in a distributed continuous space.

The neural network architecture shown in Figure 4 is almost identical to the original feed-forward neural network described in [Bengio *et al.*, 2003; Vaswani *et al.*, 2013; Devlin *et al.*, 2014]. It consists of two hidden layers besides the input and output layer. The input includes two parts: 1) the local $m+n-1$ context vector used in [Devlin *et al.*, 2014] ($n-1$ target history words and m source-side context words), where each word is mapped to a 192-dimensional vector; 2) the global 192-dimensional sentence semantic representation vector obtained with the learnt chunk-based CNN.

Through two 512-dimensional hidden layers with rectified linear activation function (Relu, $\sigma(x) = max(0, x)$), we apply the softmax function in the output layer to calculate the probability for each target word in the vocabulary. Following [Devlin *et al.*, 2014], input vocabulary contains 16,000 source words and 16,000 target words. The output vocabulary contains 32,000 target words.

This feed-forward neural network will be trained to maximize the log-likelihood over the target side of the bilingual training data for machine translation.

[2] Word alignment information in the translation rules are retained during decoding.

3.2 Decoding with Neural Network Probability

To calculate the target word conditional probability with neural network, we need the information of $n-1$ target history words, m source context words and the whole source sentence, which are all easy to obtain during SMT decoding. Thus, the neural network probability can be integrated into any SMT model. In this paper, the hierarchical phrase-based model (HPB) [Chiang, 2007] is employed.

The HPB model translates the source sentences using synchronous context free grammars (SCFG). The SCFG translation rules are in the form of $X \to (\alpha, \gamma, \sim)$, where X is the non-terminal symbol, α and γ are sequences of lexical items and non-terminals in the source and target side respectively, and $\sim$ indicates one-to-one correspondence between non-terminals in α and γ in the standard version. Here, in order to retrieve easily the central source word aligned to each target word during decoding, we also retain the alignments between source- and target-side lexical terms in the rules. That is, $\sim$ contains all the correspondences between terminal or non-terminals in α and γ.

The HPB SMT adopts a log-linear model to search for the best translation candidate. The powerful features in the log-linear model includes: 1) forward and backward rule probability, 2) forward and backward lexical probability, 3) a 5-gram language model, 4) rule counts and translation length penalty, and 5) a glue rule reordering model. The conditional probability $p(t_i|t_{i-n+1}, \cdots, t_{i-1}, \mathcal{S}_i, s)$ calculated in the previous section will serve as the sixth kind of the informative feature to be integrated in the log-linear model.

4 Experiments

Before elaborating the experimental results, we first introduce the details of neural network training and the experimental settings.

4.1 Neural Network Training Details

For the bilingually-constrained chunk-based CNN, the initial 192-dimensional word embeddings are trained with word2vec [Mikolov *et al.*, 2013]: the embedding for English words is learnt on ~1.1B data, while that for Chinese words is learnt on ~0.7B data. We set the context window $h = 3$ for convolution. We will test multiple settings of the chunk number ($C = 1, 2, 4, 8$) to see which one performs best. We apply $L = 100$ filters. The two fully connected linear layers both contain 192 neurons. The dropout ratio in the dropout layer is set 0.5 to prevent overfitting. Dot-product is employed to calculate the distance between source and target sentence representations in the shared semantic space. The standard back-propagation and stochastic gradient descent (SGD) algorithm is utilized to optimize this network.

For the feed-forward neural network, we also apply the SGD algorithm. A key issue of this neural network is that the computation in the softmax layer is too time consuming since normalization is required over the entire huge vocabulary. Inspired by [Vaswani *et al.*, 2013], we adopt the Noisy Contrastive Estimation (NCE) [Gutmann and Hyvärinen, 2010] to avoid the normalization in the output layer.

Data	Chinese Sent. Num.	English Sent. Num.
bilingual data	2,086,731	2,086,731
Xinhua News	~	10,912,683
NIST03	919	919×4
NIST05	1,082	1,082×4
NIST06	1,000	1,000×4
NIST08	691	691×4

Table 1: Data statistics of the SMT experiment.

4.2 Experimental Setup

The SMT evaluation is conducted on Chinese-to-English translation. The bilingual training data [3] from LDC contains about 2.1 million sentence pairs. This bilingual data is also utilized to train the two neural networks. The 5-gram language model is trained on the English part of the bilingual training data and the Xinhua portion of the English Gigaword corpus. NIST MT03 is used as the tuning data. MT05, MT06 and MT08 (news data) are used as the test data. Table 1 shows the detailed data statistics. Case-insensitive BLEU is employed as the evaluation metric. The statistical significance test is performed with the pairwise re-sampling approach [Koehn, 2004].

4.3 Experimental Results

To have a comprehensive understanding about the capacity of our proposed model, we compare our method with several baselines. The information of different systems is detailed as follows:

- **HPB**: the hierarchical phrase-based translation system.
- **+NNJM**: HPB system incorporating feed-forward neural network joint model in which the probability is predicted with 3 target history words and 11 source-side local context words.
- **+AVE_SENT**: it is similar to +NNJM, but the neural network probability is augmented with source-side global sentence representation which is obtained by averaging all the word embeddings in the sentence.
- **+BCCNN**: it is similar to +AVE_SENT. Instead of average embedding, the sentence semantic representation is learnt using the bilingually-constrained chunk-based convolutional neural networks.

Table 2 gives the detailed results. First, let us look at the performance of the neural network joint model using the local contexts (+NNJM). Compared to the hierarchical phrase-based model HPB, this model performs significantly better on test set MT05 and MT08. The biggest improvement can be up to 0.95 BLEU score. Devlin *et al.* [2014] has reported that this model can outperform HPB by more than 1.0 BLEU score on Chinese-to-English translation. Although our improvement is not so promising, we demonstrate that it is much helpful to apply the neural network joint model using source- and target-side local contexts.

[3] LDC2000T50, LDC2002L27, LDC2002T01, LDC2003E07, LDC2003E14, LDC2003T17, LDC2004T07, LDC2005T06, LDC2005T10 and LDC2005T34.

System	MT03	MT05	MT06	MT08
HPB	35.98	34.66	35.25	27.80
+NNJM	36.93	35.55^{+}	35.77	28.64^{+}
+AVE_SENT	37.16	35.88^{+}	36.07^{+}	29.19^{+}
+BCCNN-1	37.32	36.06^{+}	36.42^{+}	29.35^{+*}
+BCCNN-2	37.75	36.24^{+}	36.65^{+*}	29.97^{+*}
+BCCNN-4	**37.98**	36.22^{+}	$\mathbf{36.78}^{+*}$	$\mathbf{30.02}^{+*}$
+BCCNN-8	37.64	$\mathbf{36.29}^{+*}$	36.49^{+}	29.98^{+*}

Table 2: Experimental results of different translation systems. Significance test is performed on the test sets. "+" means that the model significantly outperforms the baseline HPB with $p < 0.01$. "$*$" indicates that the model is significantly better than +NNJM with $p < 0.01$. "+BCNN-4" denotes that this model adopts 4 chunks in the pooling layer.

When the model +NNJM is augmented with the global sentence representations, we can obtain more gains (see last 5 lines in Table 2). Specifically, if the sentence representation is generated by just averaging all the word embeddings, it can get slight improvements over the model +NNJM. However, +AVE_SENT cannot perform significantly better than +NNJM. These results indicate that the sentence representation is beneficial to improving the translation quality. Due to the lack of adequate semantics, it does not lead to great improvements.

As we can see that if the sentence representation is learnt by the bilingually-constrained chunk-based CNN (BCCNN), the models can achieve much more BLEU score improvements no matter how many chunks we adopt. Note that using only one chunk is equivalent to the *max-over-time* pooling. From Table 2 we see that more chunks perform better than the *max-over-time* pooling. The results indicate that partitioning the sentence into several chunks and summarizing respectively their important semantics can result in better sentence semantic representations.

Overall, the model using 4 chunks (+BCCNN-4) performs best. It obtains three bests out of four sets and it significantly outperforms the model +NNJM on test sets MT06 and MT08. The largest gains can be up to 1.38 BLEU score on MT08. It is interesting that the model with 8 chunks just performs similarly to that with 2 chunks. We speculate that too many chunks may bring some noise. However, it deserves deep investigation. Nevertheless, we can say that the global sentence semantic representation much benefits the target translation prediction.

5 Related Work

Our work mainly includes two key issues: one is learning the sentence semantic representation, and the other is applying global sentence-level information to better predict target translations. We will discuss the related work from these two aspects.

On sentence representation learning, many researchers perform this task using the neural network methods. Sutskever *et al.* [2014] and Bahdanau *et al.* [2014] applied the recurrent neural networks to encode the source sentence and decode from the source sentence representation. And it takes long time to train their models. Socher *et al.*; Socher *et al.*; Socher *et al.* [2011; 2013a; 2013b] designed the recursive neural networks for syntactic parsing and sentiment analysis, in which sentence representation is the by-product. Le and Mikolov [2014] used a simple feed-forward neural network to learn sentence and paragraph representations, but one disadvantage is that test sentence representation must be learnt by performing the training process. Kim [2014], Kalchbrenner *et al.* [2014], Hu *et al.* [2014] and Zeng *et al.* [2014] adopted the convolutional neural networks to learn sentence representations for different classification tasks.

The sentence representations learnt with the above methods are mainly task dependent. For example they are sensitive to sentiment (relation or structure) of the sentence [Socher *et al.*, 2013b; Kim, 2014; Zeng *et al.*, 2014]. These sentence representations do not encode adequate semantics of the sentence. In contrast, we aim at encoding as much semantics as possible in the sentence representation by designing the bilingually-constrained chunk-based convolutional neural networks.

On applying more information for translation prediction, Devlin *et al.* [2014] developed a neural network joint model to make full use of the source- and target-side local contexts. However, they ignored the global sentence-level information. Recently, some researchers exploited the information beyond the sentence level. For example, Eidelman *et al.* [2012], Su *et al.* [2012] and Zhang *et al.* [2014b] attempted to apply the topic information of the document to distinguish good translation rules from bad ones during SMT decoding. But, their methods require the sentence's document information which is difficult to satisfy in practice. Instead, we just focus on the sentence-level features. We have designed two neural networks: one for sentence semantic representation learning, and the other for target word probability prediction.

6 Conclusions and Future Work

In this paper, we have explored the source-side global sentence representations for target translation prediction. We presented a new bilingually-constrained chunk-based convolutional neural network to learn sentence semantic representations. In order to integrate the sentence representation in SMT model, we further applied a feed-forward neural network joint model to better predict translation probability with both local and global information. The extensive experiments have shown that the proposed model can significantly outperform the strong hierarchical phrase-based translation model enriched with broader context features.

At present, the number of chunks in the chunk-based CNN is determined manually. We plan to analyse the sentence structure and design a theoretical way to choose the number of chunks. We also would like to apply our sentence semantic representations to other tasks such as question answering and paraphrase detection.

References

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

[Bengio *et al.*, 2003] Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. A neural probabilistic language model. *Journal of Machine Learning Research*, 3:1137–1155, 2003.

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

[Collobert *et al.*, 2011] Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. Natural language processing (almost) from scratch. *The Journal of Machine Learning Research*, 12:2493–2537, 2011.

[Devlin *et al.*, 2014] Jacob Devlin, Rabih Zbib, Zhongqiang Huang, Thomas Lamar, Richard Schwartz, and John Makhoul. Fast and robust neural network joint models for statistical machine translation. In *Proceedings of ACL*, 2014.

[Eidelman *et al.*, 2012] Vladimir Eidelman, Jordan Boyd-Graber, and Philip Resnik. Topic models for dynamic translation model adaptation. In *Proceedings of ACL*, pages 115–119, 2012.

[Galley *et al.*, 2006] Michel Galley, Jonathan Graehl, Kevin Knight, Daniel Marcu, Steve DeNeefe, Wei Wang, and Ignacio Thayer. Scalable inference and training of context-rich syntactic translation models. In *Proceedings of ACL*, pages 961–968, 2006.

[Gutmann and Hyvärinen, 2010] Michael Gutmann and Aapo Hyvärinen. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *International Conference on Artificial Intelligence and Statistics*, pages 297–304, 2010.

[Hu *et al.*, 2014] Baotian Hu, Zhengdong Lu, Hang Li, and Qingcai Chen. Convolutional neural network architectures for matching natural language sentences. In *Proceedings of NIPS*, pages 2042–2050, 2014.

[Kalchbrenner and Blunsom, 2013] Nal Kalchbrenner and Phil Blunsom. Recurrent continuous translation models. In *Proceedings of EMNLP*, pages 1700–1709, 2013.

[Kalchbrenner *et al.*, 2014] Nal Kalchbrenner, Phil Blunsom, and Edward Grefenstette. A convolutional neural network for modelling sentences. In *Proceedings of ACL*, 2014.

[Kim, 2014] Yoon Kim. Convolutional neural networks for sentence classification. In *Proceedings of EMNLP 2014*, 2014.

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

[Koehn, 2004] Philipp Koehn. Statistical significance tests for machine translation evaluation. In *Proceedings of EMNLP*, pages 388–395, 2004.

[Le and Mikolov, 2014] Quoc V Le and Tomas Mikolov. Distributed representations of sentences and documents. In *International Conference on Machine Learning*, 2014.

[Liu *et al.*, 2006] Yang Liu, Qun Liu, and Shouxun Lin. Tree-to-string alignment template for statistical machine translation. In *Proceedings of ACL*, pages 609–616, 2006.

[Mikolov *et al.*, 2013] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. Distributed representations of words and phrases and their compositionality. In *Proceedings of NIPS*, 2013.

[Socher *et al.*, 2011] Richard Socher, Cliff C Lin, Chris Manning, and Andrew Y Ng. Parsing natural scenes and natural language with recursive neural networks. In *Proceedings of ICML-11*, pages 129–136, 2011.

[Socher *et al.*, 2013a] Richard Socher, John Bauer, Christopher D Manning, and Andrew Y Ng. Parsing with compositional vector grammars. In *Proceedings of ACL*, 2013.

[Socher *et al.*, 2013b] Richard Socher, Alex Perelygin, Jean Y Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of EMNLP*, 2013.

[Su *et al.*, 2012] Jinsong Su, Hua Wu, Haifeng Wang, Yidong Chen, Xiaodong Shi, Huailin Dong, and Qun Liu. Translation model adaptation for statistical machine translation with monolingual topic information. In *Proceedings of ACL*, pages 459–468, 2012.

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

[Vaswani *et al.*, 2013] Ashish Vaswani, Yinggong Zhao, Victoria Fossum, and David Chiang. Decoding with large-scale neural language models improves translation. In *Proceedings of EMNLP*, pages 1387–1392, 2013.

[Zeng *et al.*, 2014] Daojian Zeng, Kang Liu, Siwei Lai, Guangyou Zhou, and Jun Zhao. Relation classification via convolutional deep neural network. In *Proceedings of COLING*, pages 2335–2344, 2014.

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

[Zhang *et al.*, 2014b] Min Zhang, Xinyan Xiao, Deyi Xiong, and Qun Liu. Topic-based dissimilarity and sensitivity models for translation rule selection. *Journal of Artificial Intelligence Research*, 50(1):1–30, 2014.

[Zou *et al.*, 2013] Will Y Zou, Richard Socher, Daniel Cer, and Christopher D Manning. Bilingual word embeddings for phrase-based machine translation. In *Proceedings of EMNLP 2013*, pages 1393–1398, 2013.