Bilingually-constrained Phrase Embeddings for Machine Translation

Jiajun Zhang[1], **Shujie Liu**[2], **Mu Li**[2], **Ming Zhou**[2] **and Chengqing Zong**[1]
[1]National Laboratory of Pattern Recognition, CASIA, Beijing, P.R. China
{jjzhang,cqzong}@nlpr.ia.ac.cn
[2]Microsoft Research Asia, Beijing, P.R. China
{shujliu,muli,mingzhou}@microsoft.com

Abstract

We propose Bilingually-constrained Recursive Auto-encoders (BRAE) to learn semantic phrase embeddings (compact vector representations for phrases), which can distinguish the phrases with different semantic meanings. The BRAE is trained in a way that minimizes the semantic distance of translation equivalents and maximizes the semantic distance of non-translation pairs simultaneously. After training, the model learns how to embed each phrase semantically in two languages and also learns how to transform semantic embedding space in one language to the other. We evaluate our proposed method on two end-to-end SMT tasks (phrase table pruning and decoding with phrasal semantic similarities) which need to measure semantic similarity between a source phrase and its translation candidates. Extensive experiments show that the BRAE is remarkably effective in these two tasks.

1 Introduction

Due to the powerful capacity of feature learning and representation, Deep (multi-layer) Neural Networks (DNN) have achieved a great success in speech and image processing (Kavukcuoglu et al., 2010; Krizhevsky et al., 2012; Dahl et al., 2012).

Recently, statistical machine translation (SMT) community has seen a strong interest in adapting and applying DNN to many tasks, such as word alignment (Yang et al., 2013), translation confidence estimation (Mikolov et al., 2010; Liu et al., 2013; Zou et al., 2013), phrase reordering prediction (Li et al., 2013), translation modelling (Auli et al., 2013; Kalchbrenner and Blunsom, 2013) and language modelling (Duh et al., 2013; Vaswani et al., 2013). Most of these works attempt to improve some components in SMT based on *word embedding*, which converts a word into a dense, low dimensional, real-valued vector representation (Bengio et al., 2003; Bengio et al., 2006; Collobert and Weston, 2008; Mikolov et al., 2013).

However, in the conventional (phrase-based) SMT, phrases are the basic translation units. The models using word embeddings as the direct inputs to DNN cannot make full use of the whole syntactic and semantic information of the phrasal translation rules. Therefore, in order to successfully apply DNN to model the whole translation process, such as modelling the decoding process, learning compact vector representations for the basic phrasal translation units is the essential and fundamental work.

In this paper, we explore the phrase embedding, which represents a phrase (sequence of words) with a real-valued vector. In some previous works, phrase embedding has been discussed from different views. Socher et al. (2011) make the phrase embeddings capture the sentiment information. Socher et al. (2013a) enable the phrase embeddings to mainly capture the syntactic knowledge. Li et al. (2013) attempt to encode the reordering pattern in the phrase embeddings. Kalchbrenner and Blunsom (2013) utilize a simple convolution model to generate phrase embeddings from word embeddings. Mikolov et al. (2013) consider a phrase as an indivisible n-gram. Obviously, these methods of learning phrase embeddings either focus on some aspects of the phrase (e.g. reordering pattern), or impose strong assumptions (e.g. bag-of-words or indivisible n-gram). Therefore, these phrase embeddings are not suitable to fully represent the phrasal translation units in SMT due to the lack of semantic meanings of the phrase.

Instead, we focus on learning phrase embeddings from the view of semantic meaning, so that our phrase embedding can fully represent the phrase and best fit the phrase-based SMT. Assuming the phrase is a meaningful composition

Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, pages 111–121,
Baltimore, Maryland, USA, June 23-25 2014. ©2014 Association for Computational Linguistics

of its internal words, we propose Bilingually-constrained Recursive Auto-encoders (BRAE) to learn semantic phrase embeddings. The core idea behind is that a phrase and its correct translation should share the same semantic meaning. Thus, they can supervise each other to learn their semantic phrase embeddings. Similarly, non-translation pairs should have different semantic meanings, and this information can also be used to guide learning semantic phrase embeddings.

In our method, the standard recursive auto-encoder (RAE) pre-trains the phrase embedding with an unsupervised algorithm by minimizing the reconstruction error (Socher et al., 2010), while the bilingually-constrained model learns to fine-tune the phrase embedding by minimizing the semantic distance between translation equivalents and maximizing the semantic distance between non-translation pairs.

We use an example to explain our model. As illustrated in Fig. 1, the Chinese phrase on the left and the English phrase on the right are translations with each other. If we learn the embedding of the Chinese phrase correctly, we can regard it as the gold representation for the English phrase and use it to guide the process of learning English phrase embedding. In the other direction, the Chinese phrase embedding can be learned in the same way. This procedure can be performed with an co-training style algorithm so as to minimize the semantic distance between the translation equivalents [1]. In this way, the result Chinese and English phrase embeddings will capture the semantics as much as possible. Furthermore, a transformation function between the Chinese and English semantic spaces can be learned as well.

With the learned model, we can accurately measure the semantic similarity between a source phrase and a translation candidate. Accordingly, we evaluate the BRAE model on two end-to-end SMT tasks (phrase table pruning and decoding with phrasal semantic similarities) which need to check whether a translation candidate and the source phrase are in the same meaning. In phrase table pruning, we discard the phrasal translation rules with low semantic similarity. In decoding with phrasal semantic similarities, we apply the semantic similarities of the phrase pairs as new features during decoding to guide translation candidate selection. The experiments show that up to 72% of the phrase table can be discarded without significant decrease on the translation quality, and in decoding with phrasal semantic similarities up to 1.7 BLEU score improvement over the state-of-the-art baseline can be achieved.

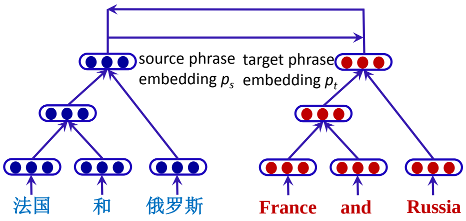

Figure 1: A motivation example for the BRAE model.

In addition, our semantic phrase embeddings have many other potential applications. For instance, the semantic phrase embeddings can be directly fed to DNN to model the decoding process. Besides SMT, the semantic phrase embeddings can be used in other cross-lingual tasks (e.g. cross-lingual question answering) and monolingual applications such as textual entailment, question answering and paraphrase detection.

2 Related Work

Recently, phrase embedding has drawn more and more attention. There are three main perspectives handling this task in monolingual languages.

One method considers the phrases as bag-of-words and employs a convolution model to transform the word embeddings to phrase embeddings (Collobert et al., 2011; Kalchbrenner and Blunsom, 2013). Gao et al. (2013) also use bag-of-words but learn BLEU sensitive phrase embeddings. This kind of approaches does not take the word order into account and loses much information. Instead, our bilingually-constrained recursive auto-encoders not only learn the composition mechanism of generating phrases from words, but also fine tune the word embeddings during the model training stage, so that we can induce the full information of the phrases and internal words.

Another method (Mikolov et al., 2013) deals with the phrases having a meaning that is not a simple composition of the meanings of its individual words, such as *New York Times*. They first find the phrases of this kind. Then, they regard these phrases as indivisible units, and learn their embeddings with the context information. How-

[1] For simplicity, we do not show non-translation pairs here.

ever, this kind of phrase embedding is hard to capture full semantics since the context of a phrase is limited. Furthermore, this method can only account for a very small part of phrases, since most of the phrases are compositional. In contrast, our method attempts to learn the semantic vector representation for any phrase.

The third method views any phrase as the meaningful composition of its internal words. The recursive auto-encoder is typically adopted to learn the way of composition (Socher et al., 2010; Socher et al., 2011; Socher et al., 2013a; Socher et al., 2013b; Li et al., 2013). They pre-train the RAE with an unsupervised algorithm. And then, they fine-tune the RAE according to the label of the phrase, such as the syntactic category in parsing (Socher et al., 2013a), the polarity in sentiment analysis (Socher et al., 2011; Socher et al., 2013b), and the reordering pattern in SMT (Li et al., 2013). This kind of semi-supervised phrase embedding is in fact performing phrase clustering with respect to the phrase label. For example, in the RAE-based phrase reordering model for SMT (Li et al., 2013), the phrases with the similar reordering tendency (e.g. monotone or swap) are close to each other in the embedding space, such as the prepositional phrases. Obviously, this kind methods of semi-supervised phrase embedding do not fully address the semantic meaning of the phrases. Although we also follow the composition-based phrase embedding, we are the first to focus on the semantic meanings of the phrases and propose a bilingually-constrained model to induce the semantic information and learn transformation of the semantic space in one language to the other.

3 Bilingually-constrained Recursive Auto-encoders

This section introduces the Bilingually-constrained Recursive Auto-encoders (BRAE), that is inspired by two observations. First, the recursive auto-encoder provides a reasonable composition mechanism to embed each phrase. And the semi-supervised phrase embedding (Socher et al., 2011; Socher et al., 2013a; Li et al., 2013) further indicates that phrase embedding can be tuned with respect to the label. Second, even though we have no correct semantic phrase representation as the gold label, the phrases sharing the same meaning provide an indirect but feasible way.

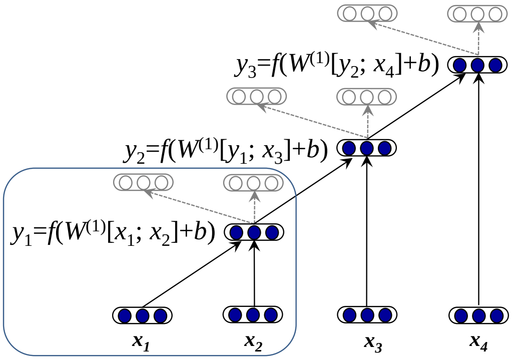
y_3=f($W^{(1)}$[y_2; x_4]+b)
y_2=f($W^{(1)}$[y_1; x_3]+b)
y_1=f($W^{(1)}$[x_1; x_2]+b)
x_1
x_2
x_3
x_4

Figure 2: A recursive auto-encoder for a four-word phrase. The empty nodes are the reconstructions of the input.

We will first briefly present the unsupervised phrase embedding, and then describe the semi-supervised framework. After that, we introduce the BRAE on the network structure, objective function and parameter inference.

3.1 Unsupervised Phrase Embedding

3.1.1 Word Vector Representations

In phrase embedding using composition, the word vector representation is the basis and serves as the input to the neural network. After learning word embeddings with DNN (Bengio et al., 2003; Collobert and Weston, 2008; Mikolov et al., 2013), each word in the vocabulary V corresponds to a vector $x \in \mathbb{R}^n$, and all the vectors are stacked into an embedding matrix $L \in \mathbb{R}^{n \times |V|}$.

Given a phrase which is an ordered list of m words, each word has an index i into the columns of the embedding matrix L. The index i is used to retrieve the word's vector representation using a simple multiplication with a binary vector e which is zero in all positions except for the ith index:

$$x_i = Le_i \in \mathbb{R}^n \tag{1}$$

Note that n is usually set empirically, such as $n = 50, 100, 200$. Throughout this paper, $n = 3$ is used for better illustration as shown in Fig. 1.

3.1.2 RAE-based Phrase Embedding

Assuming we are given a phrase $w_1 w_2 \cdots w_m$, it is first projected into a list of vectors $(x_1, x_2, \cdots, x_m)$ using Eq. 1. The RAE learns the vector representation of the phrase by recursively combining two children vectors in a bottom-up manner (Socher et al., 2011). Fig. 2 illustrates an instance of a RAE applied to a binary tree, in which a standard auto-encoder (in box) is re-used at each node. The standard auto-encoder aims at learning an abstract representation of its input. For two children $c_1 = x_1$ and $c_2 = x_2$, the auto-encoder computes the parent vector y_1 as follows:

$$p = f(W^{(1)}[c_1; c_2] + b^{(1)}) \tag{2}$$

Where we multiply the parameter matrix $W^{(1)} \in \mathbb{R}^{n\times 2n}$ by the concatenation of two children $[c_1; c_2] \in \mathbb{R}^{2n\times 1}$. After adding a bias term $b^{(1)}$, we apply an element-wise activation function such as $f = tanh(\cdot)$, which is used in our experiments. In order to apply this auto-encoder to each pair of children, the representation of the parent p should have the same dimensionality as the c_i's.

To assess how well the parent's vector represents its children, the standard auto-encoder reconstructs the children in a reconstruction layer:

$$[c'_1; c'_2] = f^{(2)}(W^{(2)}p + b^{(2)}) \tag{3}$$

Where c'_1 and c'_2 are reconstructed children, $W^{(2)}$ and $b^{(2)}$ are parameter matrix and bias term for reconstruction respectively, and $f^{(2)} = tanh(\cdot)$.

To obtain the optimal abstract representation of the inputs, the standard auto-encoder tries to minimize the reconstruction errors between the inputs and the reconstructed ones during training:

$$E_{rec}([c_1; c_2]) = \frac{1}{2}||[c_1; c_2] - [c'_1; c'_2]||^2 \tag{4}$$

Given $y_1 = p$, we can use Eq. 2 again to compute y_2 by setting the children to be $[c_1; c_2] = [y_1; x_3]$. The same auto-encoder is re-used until the vector of the whole phrase is generated.

For unsupervised phrase embedding, the only objective is to minimize the sum of reconstruction errors at each node in the optimal binary tree:

$$RAE_\theta(x) = \underset{y\in A(x)}{\operatorname{argmin}} \sum_{s\in y} E_{rec}([c_1; c_2]_s) \tag{5}$$

Where x is the list of vectors of a phrase, and $A(x)$ denotes all the possible binary trees that can be built from inputs x. A greedy algorithm (Socher et al., 2011) is used to generate the optimal binary tree y. The parameters $\theta = (W, b)$ are optimized over all the phrases in the training data.

3.2 Semi-supervised Phrase Embedding

The above RAE is completely unsupervised and can only induce general representations of the multi-word phrases. Several researchers extend the original RAEs to a semi-supervised setting so that the induced phrase embedding can predict a target label, such as polarity in sentiment analysis (Socher et al., 2011), syntactic category in parsing (Socher et al., 2013a) and phrase reordering pattern in SMT (Li et al., 2013).

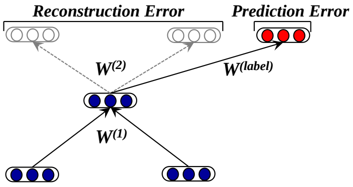

Figure 3: An illustration of a semi-supervised RAE unit. Red nodes show the label distribution.

In the semi-supervised RAE for phrase embedding, the objective function over a (phrase, label) pair (x, t) includes the reconstruction error and the prediction error, as illustrated in Fig. 3.

$$E(x, t; \theta) = \alpha E_{rec}(x, t; \theta) + (1-\alpha) E_{pred}(x, t; \theta) \tag{6}$$

Where the hyper-parameter α is used to balance the reconstruction and prediction error. For label prediction, the cross-entropy error is usually used to calculate E_{pred}. By optimizing the above objective, the phrases in the vector embedding space will be grouped according to the labels.

3.3 The BRAE Model

We know from the semi-supervised phrase embedding that the learned vector representation can be well adapted to the given label. Therefore, we can imagine that learning semantic phrase embedding is reasonable if we are given gold vector representations of the phrases.

However, no gold semantic phrase embedding exists. Fortunately, we know the fact that the two phrases should share the same semantic representation if they express the same meaning. We can make inference from this fact that if a model can learn the same embedding for any phrase pair sharing the same meaning, the learned embedding must encode the semantics of the phrases and the corresponding model is our desire.

As translation equivalents share the same semantic meaning, we employ high-quality phrase translation pairs as training corpus in this work. Accordingly, we propose the Bilingually-constrained Recursive Auto-encoders (BRAE),

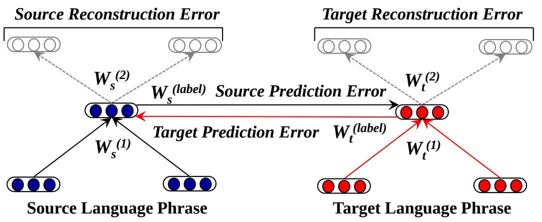

Figure 4: An illustration of the bilingual-constrained recursive auto-encoders. The two phrases are translations with each other.

whose basic goal is to minimize the semantic distance between the phrases and their translations.

3.3.1 The Objective Function

Unlike previous methods, the BRAE model jointly learns two RAEs (Fig. 4 shows the network structure): one for source language and the other for target language. For a phrase pair (s, t), two kinds of errors are involved:

1. **reconstruction error** $E_{rec}(s, t; \theta)$: how well the learned vector representations p_s and p_t represent the phrase s and t respectively?

$$E_{rec}(s, t; \theta) = E_{rec}(s; \theta) + E_{rec}(t; \theta) \qquad (7)$$

2. **semantic error** $E_{sem}(s, t; \theta)$: what is the semantic distance between the learned vector representations p_s and p_t?

Since word embeddings for two languages are learned separately and locate in different vector space, we do not enforce the phrase embeddings in two languages to be in the same semantic vector space. We suppose there is a transformation between the two semantic embedding spaces. Thus, the semantic distance is bidirectional: the distance between p_t and the transformation of p_s, and that between p_s and the transformation of p_t. As a result, the overall semantic error becomes:

$$E_{sem}(s, t; \theta) = E_{sem}(s|t, \theta) + E_{sem}(t|s, \theta) \quad (8)$$

Where $E_{sem}(s|t, \theta) = E_{sem}(p_t, f(W_s^l p_s + b_s^l))$ means the transformation of p_s is performed as follows: we first multiply a parameter matrix W_s^l by p_s, and after adding a bias term b_s^l we apply an element-wise activation function $f = tanh(\cdot)$. Finally, we calculate their Euclidean distance:

$$E_{sem}(s|t, \theta) = \frac{1}{2}||p_t - f(W_s^l p_s + b_s^l)||^2 \qquad (9)$$

$E_{sem}(t|s, \theta)$ can be calculated in exactly the same way. For the phrase pair (s, t), the joint error is:

$$E(s, t; \theta) = \alpha E_{rec}(s, t; \theta) + (1 - \alpha) E_{sem}(s, t; \theta) \qquad (10)$$

The hyper-parameter α weights the reconstruction and semantic error. The final BRAE objective over the phrase pairs training set (S, T) becomes:

$$J_{BRAE} = \frac{1}{N} \sum_{(s,t)\in(S,T)} E(s, t; \theta) + \frac{\lambda}{2}||\theta||^2 \quad (11)$$

3.3.2 Max-Semantic-Margin Error

Ideally, we want the learned BRAE model can make sure that the semantic error for the positive example (a source phrase s and its correct translation t) is much smaller than that for the negative example (the source phrase s and a bad translation t'). However, the current model cannot guarantee this since the above semantic error $E_{sem}(s|t, \theta)$ only accounts for positive ones.

We thus enhance the semantic error with both positive and negative examples, and the corresponding max-semantic-margin error becomes:

$$\begin{aligned} E^*_{sem}(s|t, \theta) = max\{0, & E_{sem}(s|t, \theta) \\ & - E_{sem}(s|t', \theta) + 1\} \end{aligned} \qquad (12)$$

It tries to minimize the semantic distance between translation equivalents and maximize the semantic distance between non-translation pairs simultaneously. Using the above error function, we need to construct a negative example for each positive example. Suppose we are given a positive example (s, t), the correct translation t can be converted into a bad translation t' by replacing the words in t with randomly chosen target language words. Then, a negative example (s, t') is available.

3.3.3 Parameter Inference

Like semi-supervised RAE (Li et al., 2013), the parameters θ in our BRAE model can also be divided into three sets:

θ_L: word embedding matrix L for two languages (Section 3.1.1);

θ_{rec}: recursive auto-encoder parameter matrices $W^{(1)}$, $W^{(2)}$, and bias terms $b^{(1)}$, $b^{(2)}$ for two languages (Section 3.1.2);

θ_{sem}: transformation matrix W^l and bias term b^l for two directions in semantic distance computation (Section 3.3.1).

To have a deep understanding of the parameters, we rewrite Eq. 10:

$$\begin{aligned} E(s,t;\theta) &= \alpha(E_{rec}(s;\theta) + E_{rec}(t;\theta)) \\ &+ (1-\alpha)(E^{*}_{sem}(s|t,\theta) + E^{*}_{sem}(t|s,\theta)) \\ &= (\alpha E_{rec}(s;\theta_s) + (1-\alpha)E^{*}_{sem}(s|t,\theta_s)) \\ &+ (\alpha E_{rec}(t;\theta_t) + (1-\alpha)E^{*}_{sem}(t|s,\theta_t)) \end{aligned} \tag{13}$$

We can see that the parameters θ can be divided into two classes: θ_s for the source language and θ_t for the target language. The above equation also indicates that the source-side parameters θ_s can be optimized independently as long as the semantic representation p_t of the target phrase t is given to compute $E_{sem}(s|t,\theta)$ with Eq. 9. It is similar for the target-side parameters θ_t.

Assuming the target phrase representation p_t is available, the optimization of the source-side parameters is similar to that of semi-supervised RAE. We apply the Stochastic Gradient Descent (SGD) algorithm to optimize each parameter:

$$\theta_s = \theta_s - \eta \frac{\partial J_s}{\partial \theta_s} \tag{14}$$

In order to run SGD algorithm, we need to solve two problems: one for parameter initialization and the other for partial gradient calculation.

In parameter initialization, θ_{rec} and θ_{sem} for the source language is randomly set according to a normal distribution. For the word embedding L_s, there are two choices. First, L_s is initialized randomly like other parameters. Second, the word embedding matrix L_s is pre-trained with DNN (Bengio et al., 2003; Collobert and Weston, 2008; Mikolov et al., 2013) using large-scale unlabeled monolingual data. We prefer to the second one since this kind of word embedding has already encoded some semantics of the words. In this work, we employ the toolkit Word2Vec (Mikolov et al., 2013) to pre-train the word embedding for the source and target languages. The word embeddings will be fine-tuned in our BRAE model to capture much more semantics.

The partial gradient for one instance is computed as follows:

$$\frac{\partial J_s}{\partial \theta_s} = \frac{\partial E(s|t,\theta_s)}{\partial \theta_s} + \lambda \theta_s \tag{15}$$

Where the source-side error given the target phrase representation includes reconstruction error and updated semantic error:

$$E(s|t,\theta_s) = \alpha E_{rec}(s;\theta_s) + (1-\alpha)E^{*}_{sem}(s|t,\theta_s) \tag{16}$$

Given the current θ_s, we first construct the binary tree (as illustrated in Fig. 2) for any source-side phrase using the greedy algorithm (Socher et al., 2011). Then, the derivatives for the parameters in the fixed binary tree will be calculated via backpropagation through structures (Goller and Kuchler, 1996). Finally, the parameters will be updated using Eq. 14 and a new θ_s is obtained.

The target-side parameters θ_t can be optimized in the same way as long as the source-side phrase representation p_s is available. It seems a paradox that updating θ_s needs p_t while updating θ_t needs p_s. To solve this problem, we propose an co-training style algorithm which includes three steps:

1. **Pre-training:** applying unsupervised phrase embedding with standard RAE to pre-train the source- and target-side phrase representations p_s and p_t respectively (Section 2.1.2);

2. **Fine-tuning:** with the BRAE model, using target-side phrase representation p_t to update the source-side parameters θ_s and obtain the fine-tuned source-side phrase representation p'_s, and meanwhile using p_s to update θ_t and get the fine-tuned p'_t, and then calculate the joint error over the training corpus;

3. **Termination Check:** if the joint error reaches a local minima or the iterations reach the pre-defined number (25 is used in our experiments), we terminate the training procedure, otherwise we set $p_s = p'_s$, $p_t = p'_t$, and go to step 2.

4 Experiments

With the semantic phrase embeddings and the vector space transformation function, we apply the BRAE to measure the semantic similarity between a source phrase and its translation candidates in the phrase-based SMT. Two tasks are involved in the experiments: phrase table pruning that discards entries whose semantic similarity is very low and decoding with the phrasal semantic similarities as additional new features.

4.1 Hyper-Parameter Settings

The hyper-parameters in the BRAE model include the dimensionality of the word embedding n in Eq. 1, the balance weight α in Eq. 10, λs in Eq. 11, and the learning rate η in Eq. 14.

For the dimensionality n, we have tried three settings $n = 50, 100, 200$ in our experiments. We empirically set the learning rate $\eta = 0.01$. We draw α from 0.05 to 0.5 with step 0.05, and λs from $\{10^{-6}, 10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}\}$. The overall error of the BRAE model is employed to guide the search procedure. Finally, we choose $\alpha = 0.15$, $\lambda_L = 10^{-2}$, $\lambda_{rec} = 10^{-3}$ and $\lambda_{sem} = 10^{-3}$.

4.2 SMT Setup

We have implemented a phrase-based translation system with a maximum entropy based reordering model using the bracketing transduction grammar (Wu, 1997; Xiong et al., 2006).

The SMT evaluation is conducted on Chinese-to-English translation. Accordingly, our BRAE model is trained on Chinese and English. The bilingual training data from LDC [2] contains 0.96M sentence pairs and 1.1M entity pairs with 27.7M Chinese words and 31.9M English words. A 5-gram language model is trained on the Xinhua portion of the English Gigaword corpus and the English part of bilingual training data. The NIST MT03 is used as the development data. NIST MT04-06 and MT08 (news data) are used as the test data. Case-insensitive BLEU is employed as the evaluation metric. The statistical significance test is performed by the re-sampling approach (Koehn, 2004).

In addition, we pre-train the word embedding with toolkit Word2Vec on large-scale monolingual data including the aforementioned data for SMT. The monolingual data contains 1.06B words for Chinese and 1.12B words for English. To obtain high-quality bilingual phrase pairs to train our BRAE model, we perform forced decoding for the bilingual training sentences and collect the phrase pairs used. After removing the duplicates, the remaining 1.12M bilingual phrase pairs (length ranging from 1 to 7) are obtained.

4.3 Phrase Table Pruning

Pruning most of the phrase table without much impact on translation quality is very important for translation especially in environments where memory and time constraints are imposed. Many algorithms have been proposed to deal with this problem, such as significance pruning (Johnson et al., 2007; Tomeh et al., 2009), relevance pruning (Eck et al., 2007) and entropy-based pruning (Ling et al., 2012; Zens et al., 2012). These algorithms are based on corpus statistics including co-occurrence statistics, phrase pair usage and composition information. For example, the significance pruning, which is proven to be a very effective algorithm, computes the probability named p-value, that tests whether a source phrase s and a target phrase t co-occur more frequently in a bilingual corpus than they happen just by chance. The higher the p-value, the more likely of the phrase pair to be spurious.

Our work has the same objective, but instead of using corpus statistics, we attempt to measure the quality of the phrase pair from the view of semantic meaning. Given a phrase pair (s, t), the BRAE model first obtains their semantic phrase representations (p_s, p_t), and then transforms p_s into target semantic space $p_s{}^*$, p_t into source semantic space $p_t{}^*$. We finally get two similarities $Sim(p_s{}^*, p_t)$ and $Sim(p_t{}^*, p_s)$. Phrase pairs that have a low similarity are more likely to be noise and more prone to be pruned. In experiments, we discard the phrase pair whose similarity in two directions are smaller than a threshold [3].

Table 1 shows the comparison results between our BRAE-based pruning method and the significance pruning algorithm. We can see a common phenomenon in both of the algorithms: for the first few thresholds, the phrase table becomes smaller and smaller while the translation quality is not much decreased, but the performance jumps a lot at a certain threshold (16 for Significance pruning, 0.8 for BRAE-based one).

Specifically, the Significance algorithm can safely discard 64% of the phrase table at its threshold 12 with only 0.1 BLEU loss in the overall test. In contrast, our BRAE-based algorithm can remove 72% of the phrase table at its threshold 0.7 with only 0.06 BLEU loss in the overall evaluation. When the two algorithms using a similar portion of the phrase table [4] (35% in BRAE and 36% in Significance), the BRAE-based algorithm outperforms the Significance algorithm on all the test sets except for MT04. It indicates that our BRAE model is a good alternative for phrase table pruning. Furthermore, our model is much more in-

[2] LDC category numbers: LDC2000T50, LDC2002L27, LDC2003E07, LDC2003E14, LDC2004T07, LDC2005T06, LDC2005T10 and LDC2005T34.

[3] To avoid the situation that all the translation candidates for a source phrase are pruned, we always keep the first 10 best according to the semantic similarity.

[4] In the future, we will compare the performance by enforcing the two algorithms to use the same portion of phrase table

Method	Threshold	PhraseTable	MT03	MT04	MT05	MT06	MT08	ALL
Baseline		100%	35.81	36.91	34.69	33.83	27.17	34.82
BRAE	0.4	52%	35.94	36.96	35.00	34.71	27.77	**35.16**
	0.5	44%	35.67	36.59	34.86	33.91	27.25	**34.89**
	0.6	35%	35.86	36.71	34.93	34.63	27.34	**35.05**
	0.7	28%	35.55	36.62	34.57	33.97	27.10	**34.76**
	0.8	20%	35.06	36.01	34.13	33.04	26.66	34.04
Significance	8	48%	35.86	36.99	34.74	34.53	27.59	**35.13**
	12	36%	35.59	36.73	34.65	34.17	27.16	**34.72**
	16	25%	35.19	36.24	34.26	33.32	26.55	34.09
	20	18%	35.05	36.09	34.02	32.98	26.37	33.97

Table 1: Comparison between BRAE-based pruning and Significance pruning of phrase table. Threshold means similarity in BRAE and negative-log-p-value in Significance. ”ALL” combines the development and test sets. **Bold numbers** denote that the result is better than or comparable to that of baseline. $n = 50$ is used for embedding dimensionality.

tuitive because it is directly based on the semantic similarity.

4.4 Decoding with Phrasal Semantic Similarities

Besides using the semantic similarities to prune the phrase table, we also employ them as two informative features like the phrase translation probability to guide translation hypotheses selection during decoding. Typically, four translation probabilities are adopted in the phrase-based SMT, including phrase translation probability and lexical weights in both directions. The phrase translation probability is based on co-occurrence statistics and the lexical weights consider the phrase as bag-of-words. In contrast, our BRAE model focuses on compositional semantics from words to phrases. Therefore, the semantic similarities computed using our BRAE model are complementary to the existing four translation probabilities.

The semantic similarities in two directions $Sim(p_s{}^*, p_t)$ and $Sim(p_t{}^*, p_s)$ are integrated into our baseline phrase-based model. In order to investigate the influence of the dimensionality of the embedding space, we have tried three different settings $n = 50, 100, 200$.

As shown in Table 2, no matter what n is, the BRAE model can significantly improve the translation quality in the overall test data. The largest improvement can be up to 1.7 BLEU score (MT06 for $n = 50$). It is interesting that with dimensionality growing, the translation performance is not consistently improved. We speculate that using $n = 50$ or $n = 100$ can already distinguish good translation candidates from bad ones.

4.5 Analysis on Semantic Phrase Embedding

To have a better intuition about the power of the BRAE model at learning semantic phrase embeddings, we show some examples in Table 3. Given the BRAE model and the phrase training set, we search from the set the most semantically similar English phrases for any new input English phrase.

The input phrases contain different number of words. The table shows that the unsupervised RAE can at most capture the syntactic property when the phrases are short. For example, the unsupervised RAE finds *do not want* for the input phrase *do not agree*. When the phrase becomes longer, the unsupervised RAE cannot even capture the syntactic property. In contrast, our BRAE model learns the semantic meaning for each phrase no matter whether it is short or relatively long. This indicates that the proposed BRAE model is effective at learning semantic phrase embeddings.

5 Discussions

5.1 Applications of The BRAE model

As the semantic phrase embedding can fully represent the phrase, we can go a step further in the phrase-based SMT and feed the semantic phrase embeddings to DNN in order to model the whole translation process (e.g. derivation structure prediction). We will explore this direction in our future work. Besides SMT, the semantic phrase embeddings can be used in other cross-lingual tasks, such as cross-lingual question answering, since the semantic similarity between phrases in different languages can be calculated accurately.

In addition to the cross-lingual applications, we believe the BRAE model can be applied in many

Method	n	**MT03**	**MT04**	**MT05**	**MT06**	**MT08**	**ALL**
Baseline		35.81	36.91	34.69	33.83	27.17	34.82
BRAE	50	36.43	37.64	35.35	35.53	28.59	$\mathbf{35.84}^{+}$
	100	36.45	37.44	35.58	35.42	28.57	$\mathbf{36.03}^{+}$
	200	36.34	37.35	35.78	34.87	27.84	$\mathbf{35.62}^{+}$

Table 2: Experimental results of decoding with phrasal semantic similarities. n is the embedding dimensionality. "+" means that the model significantly outperforms the baseline with $p < 0.01$.

New Phrase	**Unsupervised RAE**	**BRAE**
military force	core force main force labor force	military power military strength armed forces
at a meeting	to a meeting at a rate a meeting ,	at the meeting during the meeting at the conference
do not agree	one can accept i can understand do not want	do not favor will not compromise not to approve
each people in this nation	each country regards each country has its each other , and	every citizen in this country all the people in the country people all over the country

Table 3: Semantically similar phrases in the training set for the new phrases.

monolingual NLP tasks which depend on good phrase representations or semantic similarity between phrases, such as named entity recognition, parsing, textual entailment, question answering and paraphrase detection.

5.2 Model Extensions

In fact, the phrases having the same meaning are translation equivalents in different languages, but are paraphrases in one language. Therefore, our model can be easily adapted to learn semantic phrase embeddings using paraphrases.

Our BRAE model still has some limitations. For example, as each node in the recursive auto-encoder shares the same weight matrix, the BRAE model would become weak at learning the semantic representations for long sentences with tens of words. Improving the model to semantically embed sentences is left for our future work.

6 Conclusions and Future Work

This paper has explored the bilingually-constrained recursive auto-encoders in learning phrase embeddings, which can distinguish phrases with different semantic meanings. With the objective to minimize the semantic distance between translation equivalents and maximize the semantic distance between non-translation pairs simultaneously, the learned model can semantically embed any phrase in two languages and can transform the semantic space in one language to the other. Two end-to-end SMT tasks are involved to test the power of the proposed model at learning the semantic phrase embeddings. The experimental results show that the BRAE model is remarkably effective in phrase table pruning and decoding with phrasal semantic similarities.

We have also discussed many other potential applications and extensions of our BRAE model. In the future work, we will explore four directions. 1) we will try to model the decoding process with DNN based on our semantic embeddings of the basic translation units. 2) we are going to learn semantic phrase embeddings with the paraphrase corpus. 3) we will apply the BRAE model in other monolingual and cross-lingual tasks. 4) we plan to learn semantic sentence embeddings by automatically learning different weight matrices for different nodes in the BRAE model.

Acknowledgments

We thank Nan Yang for sharing the baseline code and anonymous reviewers for their valuable comments. The research work has been partially funded by the Natural Science Foundation of China under Grant No. 61333018 and 61303181, and Hi-Tech Research and Development Program (863 Program) of China under Grant No. 2012AA011102.

References

Michael Auli, Michel Galley, Chris Quirk, and Geoffrey Zweig. 2013. Joint language and translation modeling with recurrent neural networks. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1044–1054.

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

Yoshua Bengio, Holger Schwenk, Jean-Sébastien Senécal, Fréderic Morin, and Jean-Luc Gauvain. 2006. Neural probabilistic language models. In *Innovations in Machine Learning*, pages 137–186.

Ronan Collobert and Jason Weston. 2008. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th international conference on Machine learning*, pages 160–167.

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

George E Dahl, Dong Yu, Li Deng, and Alex Acero. 2012. Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition. *IEEE Transactions on Audio, Speech, and Language Processing*, 20(1):30–42.

Kevin Duh, Graham Neubig, Katsuhito Sudoh, and Hajime Tsukada. 2013. Adaptation data selection using neural language models: Experiments in machine translation. In *51st Annual Meeting of the Association for Computational Linguistics*, pages 678–683.

Matthias Eck, Stephen Vogal, and Alex Waibel. 2007. Estimating phrase pair relevance for translation model pruning. In *MTSummit XI*.

Jianfeng Gao, Xiaodong He, Wen-tau Yih, and Li Deng. 2013. Learning semantic representations for the phrase translation model. *arXiv preprint arXiv:1312.0482*.

Christoph Goller and Andreas Kuchler. 1996. Learning task-dependent distributed representations by backpropagation through structure. In *IEEE International Conference on Neural Networks*, volume 1, pages 347–352.

John Howard Johnson, Joel Martin, George Foster, and Roland Kuhn. 2007. Improving translation quality by discarding most of the phrasetable. In *Proceedings of EMNLP*.

Nal Kalchbrenner and Phil Blunsom. 2013. Recurrent continuous translation models. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1700–1709.

Koray Kavukcuoglu, Pierre Sermanet, Y-Lan Boureau, Karol Gregor, Michaël Mathieu, and Yann L Cun. 2010. Learning convolutional feature hierarchies for visual recognition. In *Advances in neural information processing systems*, pages 1090–1098.

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

Alex Krizhevsky, Ilya Sutskever, and Geoff Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems 25*, pages 1106–1114.

Peng Li, Yang Liu, and Maosong Sun. 2013. Recursive autoencoders for itg-based translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*.

Wang Ling, Joao Graça, Isabel Trancoso, and Alan Black. 2012. Entropy-based pruning for phrase-based machine translation. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 962–971.

Lemao Liu, Taro Watanabe, Eiichiro Sumita, and Tiejun Zhao. 2013. Additive neural networks for statistical machine translation. In *51st Annual Meeting of the Association for Computational Linguistics*, pages 791–801.

Tomas Mikolov, Martin Karafiát, Lukas Burget, Jan Cernockỳ, and Sanjeev Khudanpur. 2010. Recurrent neural network based language model. In *INTERSPEECH*, pages 1045–1048.

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

Richard Socher, Christopher D Manning, and Andrew Y Ng. 2010. Learning continuous phrase representations and syntactic parsing with recursive neural networks. In *Proceedings of the NIPS-2010 Deep Learning and Unsupervised Feature Learning Workshop*.

Richard Socher, Jeffrey Pennington, Eric H Huang, Andrew Y Ng, and Christopher D Manning. 2011. Semi-supervised recursive autoencoders for predicting sentiment distributions. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 151–161.

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

Richard Socher, Alex Perelygin, Jean Y Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts. 2013b. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*.

Nadi Tomeh, Nicola Cancedda, and Marc Dymetman. 2009. Complexity-based phrase-table filtering for statistical machine translation. In *Proceedings of Summit XII*, pages 144–151.

Ashish Vaswani, Yinggong Zhao, Victoria Fossum, and David Chiang. 2013. Decoding with large-scale neural language models improves translation. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1387–1392.

Dekai Wu. 1997. Stochastic inversion transduction grammars and bilingual parsing of parallel corpora. *Computational linguistics*, 23(3):377–403.

Deyi Xiong, Qun Liu, and Shouxun Lin. 2006. Maximum entropy based phrase reordering model for statistical machine translation. In *Proceedings of ACL-COLING*, pages 505–512.

Nan Yang, Shujie Liu, Mu Li, Ming Zhou, and Nenghai Yu. 2013. Word alignment modeling with context dependent deep neural network. In *51st Annual Meeting of the Association for Computational Linguistics*.

Richard Zens, Daisy Stanton, and Peng Xu. 2012. A systematic comparison of phrase table pruning techniques. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 972–983.

Will Y Zou, Richard Socher, Daniel Cer, and Christopher D Manning. 2013. Bilingual word embeddings for phrase-based machine translation. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1393–1398.