Bilingual Correspondence Recursive Autoencoders for Statistical Machine Translation

Jinsong Su[1], **Deyi Xiong**[2]*, **Biao Zhang**[1], **Yang Liu**[3], **Junfeng Yao**[1], **Min Zhang**[2]
Xiamen University, Xiamen, P.R. China[1]
Soochow University, Suzhou, P.R. China[2]
Tsinghua University, Beijing, P.R. China[3]
{jssu, biaozhang, yao0010}@xmu.edu.cn
{dyxiong, minzhang}@suda.edu.cn
liuyang2011@tsinghua.edu.cn

Abstract

Learning semantic representations and tree structures of bilingual phrases is beneficial for statistical machine translation. In this paper, we propose a new neural network model called Bilingual Correspondence Recursive Autoencoder (BCorrRAE) to model bilingual phrases in translation. We incorporate word alignments into BCorrRAE to allow it freely access bilingual constraints at different levels. BCorrRAE minimizes a joint objective on the combination of a recursive autoencoder reconstruction error, a structural alignment consistency error and a cross-lingual reconstruction error so as to not only generate alignment-consistent phrase structures, but also capture different levels of semantic relations within bilingual phrases. In order to examine the effectiveness of BCorrRAE, we incorporate both semantic and structural similarity features built on bilingual phrase representations and tree structures learned by BCorrRAE into a state-of-the-art SMT system. Experiments on NIST Chinese-English test sets show that our model achieves a substantial improvement of up to 1.55 BLEU points over the baseline.

1 Introduction

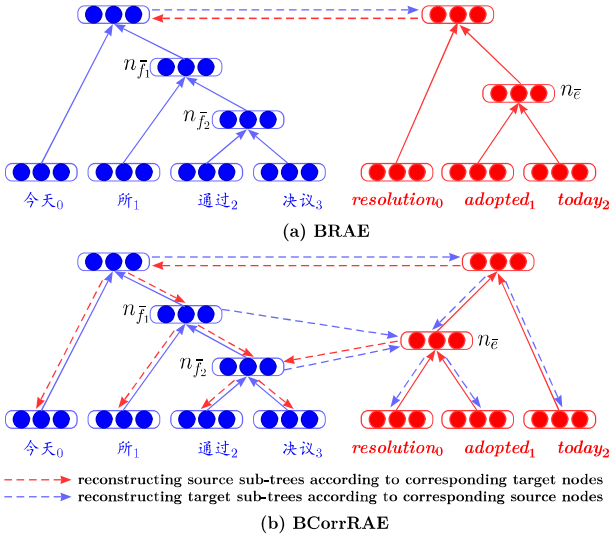
Recently a variety of "deep architecture" approaches, including autoencoders, have been successfully used in statistical machine translation (SMT) (Yang et al., 2013; Liu et al., 2013; Zou et al., 2013; Devlin et al., 2014; Tamura et al., 2014; Sundermeyer et al., 2014; Wang et al., 2014; Kočiský et al., 2014). Typically, these approaches represent words as dense, low-dimensional and real-valued vectors, i.e., word embeddings. However, translation units in machine translation have long since shifted from words to phrases (sequence of words), of which syntactic and semantic information cannot be adequately captured and represented by word embeddings. Therefore, learning compact vector representations for phrases or even longer expressions is more crucial for successful "deep" SMT.

To address this issue, many efforts have been initiated on learning representations for bilingual phrases in the context of SMT, inspired by the success of work on monolingual phrase embeddings (Socher et al., 2010; Socher et al., 2011a; Socher et al., 2013b; Chen and Manning, 2014; Kalchbrenner et al., 2014; Kim, 2014). The learning process of bilingual phrase embeddings in these efforts is normally interacted and mingled with single or multiple essential components of SMT, e.g., with reordering models (Li et al., 2013), translation models (Cui et al., 2014; Zhang et al., 2014; Gao et al., 2014), or both language and translation models (Liu et al., 2014). In spite of their success, these approaches center around capturing relations between entire source and target phrases. They do not take into account internal phrase structures and bilingual correspondences of sub-phrases within source and target phrases. The neglect of these important clues may be due to the big challenge imposed by the integration of them into the learning process of bilingual phrase representations. However, we believe such internal structures and correspondences can help us learn better phrase representations since they provide multi-level syntactic and semantic constraints.

In this paper, we propose a Bilingual Correspondence Recursive Autoencoder (BCorrRAE) to learn bilingual phrase embeddings. BCorrRAE substantially extends the Bilingually-constrained Recursive Auto-encoder (BRAE) (Zhang et al., 2014) to exploit both inner structures and corre-

*Corresponding author.

Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pages 1248–1258,
Lisbon, Portugal, 17-21 September 2015.



(a) BRAE

(b) BCorrRAE

Figure 1: BRAE vs BCorrRAE models for generating of a bilingual phrase ("今天 所 通过 决议", "*resolution adopted today*") with word alignments ("0-2 2-1 3-0"). The subscript number of each word indicates its position within phrase. Solid lines depict the generation procedure of phrase structures, while dash lines illustrate the reconstruction procedure from one language to the other. In this paper, the dimensionality of vector d in all figures is set to 3 for better illustration.

spondences within bilingual phrases. The intuitions behind BCorrRAE are twofold: 1) bilingual phrase structure generation should satisfy word alignment constraints as much as possible; and 2) corresponding sub-phrases on the source and target side of bilingual phrases should be able to reconstruct each other as they are semantic equivalents. In order to model the first intuition, BCorrRAE punishes bilingual structures that violate word alignment constraints and rewards those in consistent with word alignments. This enables BCorrRAE to produce desirable bilingual phrase structures from the perspective of word alignments. With regard to the second intuition, BCorrRAE reconstructs structures of sub-phrases of one language according to aligned nodes in the other language and minimizes the gap between original and reconstructed structures. In doing so, BCorrRAE is capable of capturing semantic relations at different levels.

To better illustrate our model, let us consider the example in Figure 1. Similar to the conventional recursive antoencoder (RAE), BRAE neglects bilingual correspondences of sub-phrases. Thus, it may combine "*adopted*" and "*today*" together to generate an undesirable target tree structure which violates word alignments. In contrast, BCorrRAE aligns source-side nodes (e.g. ("通过", "决议")) to their corresponding target-side nodes (accordingly ("*resolution*", "*adopted*")) according to word alignments. Furthermore, in BCorrRAE, each subtree on the target side can be reconstructed from the corresponding source node that aligns to the target-side node dominating the subtree and vice versa. These advantages allow us to obtain improved bilingual phrase embeddings with better inner correspondences of sub-phrases and word alignment consistency.

We conduct experiments with a state-of-the-art SMT system on large-scale data to evaluate the effectiveness of BCorrRAE model. Results on the NIST 2006 and 2008 datasets show that our system achieves significant improvements over baseline methods. The main contributions of our work lie in the following three aspects:

- We learn both embeddings and tree structures for bilingual phrases using cross-lingual RAE reconstruction that minimizes semantic distances between original and reconstructed subtrees. To the best of our knowledge, this has not been investigated before.
- We incorporate word alignment information to guide phrase structure generation and establish internal semantic associations of sub-phrases within bilingual phrases.
- We integrate two similarity features based on BCorrRAE to enhance translation candidate selection, and achieve an improvement of 1.55 BLEU points on Chinese-English translation.

2 RAE and BRAE

In this section, we briefly introduce the RAE and its bilingual variation BRAE. This will provide background knowledge on our proposed BCorrRAE.

2.1 RAE

The component in the dash box of Figure 2 illustrates an instance of an RAE applied to a three-word phrase. The input to the RAE is $x = (x_1, x_2, x_3)$, which are the d-dimensional vector representations of the ordered words in a phrase. For two children $c_1 = x_1$ and $c_2 = x_2$, the parent vector y_1 can be computed in the following way:

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

where $[c_1; c_2] \in \mathbb{R}^{2d\times 1}$ is the concatenation of c_1 and c_2, $W^{(1)} \in \mathbb{R}^{d\times 2d}$ is a parameter matrix, $b^{(1)} \in \mathbb{R}^{d\times 1}$ is a bias term, and f is an element-

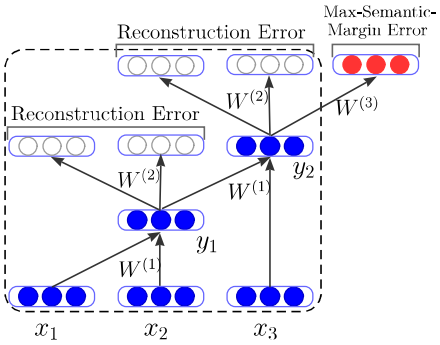


Figure 2: An illustration of the BRAE architecture.

wise activation function such as $tanh(\cdot)$, which is used for all activation functions in BRAE and our model. The learned parent vector p is also a d-dimensional vector. In order to measure how well p represents its children, we reconstruct the original children nodes in a reconstruction layer:

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

where c_1' and c_2' are reconstructed children vectors, $W^{(2)} \in \mathbb{R}^{2d \times d}$ and $b^{(2)} \in \mathbb{R}^{2d \times 1}$.

We can set $y_1 = p$ and then further use Eq. (1) again to compute y_2 by setting $[c_1; c_2] = [y_1; x_3]$. This combination and reconstruction process of auto-encoder repeats at each node until the vector of the entire phrase is generated. To obtain the optimal binary tree and phrase representation for x, we employ a greedy algorithm (Socher et al., 2011c) to minimize the sum of *reconstruction error* at each node in the binary tree $T(x)$:

$$E_{rec}(x;\theta) = \sum_{n \in T(x)} \frac{1}{2} \parallel [c_1; c_2]_n - [c_1'; c_2']_n \parallel^2 \quad (3)$$

where θ denotes model parameters and n represents a node in $T(x)$.

2.2 BRAE

BRAE jointly learns two RAEs for source and target phrase embeddings as shown in Figure 1(a). The core idea behind BRAE is that a source phrase and its target correct translation should share the same semantic representations, while non-equivalent pairs should have different semantic representations. Zhang et al. (2014) use this intuition to constrain semantic pharse embedding learning.

As shown in Figure 2, in addition to the above-mentioned reconstruction error, BRAE introduces a *max-semantic-margin error* to minimize the semantic distance between translation equivalents and maximize the semantic distance between non-equivalent pairs simultaneously. Formally, the max-semantic-margin error of a bilingual phrase (f, e) is defined as

$$E_{sem}(f, e; \theta) = E^*_{sem}(f|e, \theta) + E^*_{sem}(e|f, \theta) \quad (4)$$

where $E^*_{sem}(f|e, \theta)$ is used to ensure that the semantic error for an equivalent pair is much smaller than that for a non-equivalent pair (the source phrase f and a bad translation e'):

$$E^*_{sem}(f|e, \theta) = max\{0, \; E_{sem}(f|e, \theta) - E_{sem}(f|e', \theta) + 1\} \quad (5)$$

where $E_{sem}(f|e, \theta)$ is defined as the semantic distance between the learned vector representations of f and e, denoted by p_f and p_e, respectively. Since phrase embeddings for the source and target language are learned separately in different vector spaces, a transformation matrix $W^{(3)}_f \in \mathbb{R}^{d \times d}$ is introduced to capture this semantic transformation in the source-to-target direction. Thus, $E_{sem}(f|e, \theta)$ is calculated as

$$E_{sem}(f|e, \theta) = \frac{1}{2} \parallel p_e - f(W^{(3)}_f p_f + b^{(3)}_f) \parallel^2 \quad (6)$$

where $b^{(3)}_f \in \mathbb{R}^{d \times 1}$ is a bias term. $E^*_{sem}(e|f, \theta)$ and $E_{sem}(e|f, \theta)$ can be computed in a similar way. The joint error of (f, e) is therefore defined as follows:

$$E(f, e; \theta) = \alpha(E_{rec}(f, \theta) + E_{rec}(e, \theta)) + (1 - \alpha)(E^*_{sem}(f|e, \theta) + E^*_{sem}(e|f, \theta)) \quad (7)$$

The final BRAE objective function over the training instance set $\mathcal{D}$ becomes:

$$J_{BRAE} = \sum_{(f,e) \in D} E(f, e; \theta) + \frac{\lambda}{2} \|\theta\|^2 \quad (8)$$

Model parameters can be optimized over the total errors on training bilingual phrases in a co-training style algorithm (Zhang et al., 2014).

3 The BCorrRAE Model

As depicted above, the learned embeddings using BRAE may be unreasonable due to the neglect of bilingual constraints at different levels. To address this drawback, we propose the BCorrRAE for bilingual phrase embeddings, which incorporates bilingual correspondence information into the learning process of structures and embeddings via word alignments. In our model, we explore word alignments in two ways: (1) ensuring that a learned bilingual phrase structure is consistent with word alignments as much as possi-

ble; (2) identifying corresponding sub-phrases in the source language for reconstructing sub-phrases in the target language, and vice versa. More specifically, the former is to encourage alignment-consistent generation of sub-structures, while the latter is to minimize semantic distances between bilingual sub-phrases.

In this section, we first formally introduce a concept of structural alignment consistency encoded in bilingual phrase structure learning, which is the basis of our model. Then, we describe the objective function which is composed of three types of errors. Finally, we provide details on the training of our model.

3.1 Structural Alignment Consistency

We adapt word alignment to structural alignment and introduce some related concepts. Given a bilingual phrase (f, e) with its binary tree structures (T_f, T_e), if the source node $n_{\bar{f}} \in T_f$ covers a source-side sub-phrase $\bar{f}$, and there exists a target-side sub-phrase $\bar{e}$ such that $(\bar{f}, \bar{e})$ are consistent with word alignments (Och and Ney, 2003), we say $n_{\bar{f}}$ satisfies the *structural alignment consistency*, and it is referred to as a *structural-alignment-consistent* (SAC) *node*. Further, if $\bar{e}$ is covered by a target node $n_{\bar{e}} \in T_e$, we say $n_{\bar{e}}$ is the *aligned node* of $n_{\bar{f}}$. In this way, several different target nodes may be all aligned to the same source node because of null alignments. For this, we choose the target node with the smallest span as the aligned one for the considered source node. This is because a smaller span reflects a stronger semantic relevance in most situations.

Likewise, we have similar definitions for target nodes. Note that alignment relations between source- and target-side nodes may not be symmetric. For example, in Figure 1(b), node $n_{\bar{e}}$ is the aligned node of node $n_{\bar{f}_1}$, while node $n_{\bar{f}_2}$ rather than $n_{\bar{f}_1}$ is the aligned node of $n_{\bar{e}}$.

3.2 The Objective Function

We elaborate the three types of errors defined for a bilingual phrase (f, e) with its binary tree structures (T_f, T_e) on both sides below.

3.2.1 Reconstruction Error

Similar to RAE, the first error function is used to estimate how well learned phrase embeddings represent corresponding phrases. The *reconstruction error* $E_{rec}(f, e; \theta)$ of (f, e) is defined as follows:

$$E_{rec}(f, e; \theta) = E_{rec}(f; \theta) + E_{rec}(e; \theta) \quad (9)$$

where both $E_{rec}(f; \theta)$ and $E_{rec}(e; \theta)$ can be calculated according to Eq. (3).

3.2.2 Consistency Error

This metric corresponds to the first way in which we exploit word alignments mentioned before, which enables our model to generate as many SAC nodes as possible to respect word alignments.

Formally, the *consistency error* $E_{con}(f, e; \theta)$ of (f, e) is defined in the following way:

$$E_{con}(f, e; \theta) = E_{con}(T_f; \theta) + E_{con}(T_e; \theta) \quad (10)$$

where $E_{con}(T_f; \theta)$ and $E_{con}(T_e; \theta)$ denote the consistency error score for T_f and T_e, given word alignments. Here we only describe the calculation of the former while the latter can be calculated in exactly the same way.

To calculate $E_{con}(T_f; \theta)$, we first judge whether a source node $n_{\bar{f}}$ is an SAC node according to word alignments. Let $p_{n_{\bar{f}}}$ be the vector representation of $n_{\bar{f}}$. Following Socher et al. (2010), who use a simple inner product to measure how well the two words are combined into a phrase, we use inner product to calculate the consistency/inconsistency score for $n_{\bar{f}}$:

$$s(n_{\bar{f}}) = W^{score} p_{n_{\bar{f}}} \quad (11)$$

where $W^{score} \in \mathbb{R}^{1 \times d}$ is the score parameter. We calculate W^{score} by distinguishing SAC from non-SAC nodes defined as follows:

$$W^{score} = \begin{cases} W^{score}_{cns} & \text{if } n_{\bar{f}} \text{ is an SAC node} \\ W^{score}_{inc} & \text{otherwise} \end{cases}$$

where the subscript *cns* and *inc* represent consistency and inconsistency, respectively. For example, in Figure 3, as $n_{\bar{f}_3}$ is a non-SAC node, we calculate the inconsistency score using W^{score}_{inc} for it.

We expect T_f to satisfy structural alignment consistency as much as possible. Therefore we encourage the consistency score for T_f to be larger than its inconsistency score using a max-margin consistency error function:

$$\begin{aligned} E_{con}(T_f; \theta) = & max\{0, 1 - s(T_f)_{cns} \\ & + s(T_f)_{ins}\} \end{aligned} \quad (12)$$

where $s(T_f)_{cns}$ denotes the sum of consistency scores over all SAC nodes and $s(T_f)_{ins}$ the sum of inconsistency scores over all non-SAC nodes in T_f. Minimizing this error function will maximize the sum of consistency scores of SAC nodes and minimize (up to a margin) the sum of inconsis-

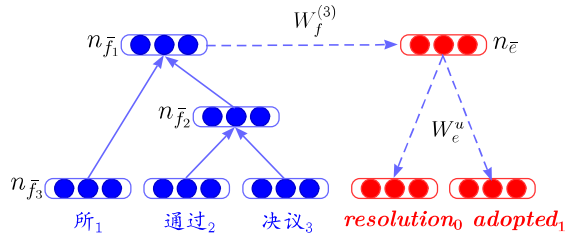


Figure 3: The structure generation procedure of the source sub-phrase "所 通过 决议" and the structure reconstruction procedure of the target sub-phrase "*resolution adopted*". According to word alignments ("2-1 3-0"), the node $n_{\bar{f}_1}$ and $n_{\bar{f}_2}$ are SAC ones while the node $n_{\bar{f}_3}$ is a non-SAC node.

tency scores of non-SAC nodes.

3.2.3 Cross-Lingual Reconstruction Error

This metric corresponds to the second way in which we exploit word alignments. The assumption behind this is that a source/target node should be able to reconstruct the entire subtree rooted at its target/source aligned node as they are semantically equivalent. Based on this, for the considered node, we calculate the cross-lingual reconstruction error along the entire subtree rooted at its aligned node in the other language and use the error to measure how well the learned vector represents this node.

Similarly, the *cross-lingual reconstruction error* $E_{clrec}(f, e; \theta)$ of (f, e) can be decomposed into two parts as follows:

$$E_{clrec}(f, e; \theta) = E_{f2e\cdot rec}(T_f, T_e; \theta) + E_{e2f\cdot rec}(T_f, T_e; \theta) \quad (13)$$

where $E_{f2e\cdot rec}(T_f, T_e; \theta)$ denotes the error score using T_f to reconstruct T_e. Note that in this process, the structure and the original node vector representations of T_e have been already generated. $E_{e2f\cdot rec}(T_f, T_e; \theta)$ denotes the reconstruction error score using T_e to reconstruct T_f. Here we still only describe the method of computing the former, which also applies to the latter.

To calculate $E_{f2e\cdot rec}(T_f, T_e; \theta)$, we first collect all source nodes $(n_{\bar{f}})$ in T_f and their aligned nodes $(n_{\bar{e}})$ in T_e to form a set of aligned node pairs $S = \{\langle n_{\bar{f}}, n_{\bar{e}}\rangle\}$ according to word alignments. We then calculate $E_{f2e\cdot rec}(T_f, T_e; \theta)$ as the sum of error scores over all node pairs in S. Given a source node $n_{\bar{f}}$ with its aligned node $n_{\bar{e}}$ on the target side, we use $n_{\bar{f}}$ to reconstruct the sub-tree structure $T_{\bar{e}}$ rooted at $n_{\bar{e}}$ and compute the error score based on the semantic distance between the original and reconstructed vector representations of nodes in $T_{\bar{e}}$. As source and target phrase embeddings are separately learned, we first introduce a transformation matrix $W_f^{(3)}$ and a bias term $b_f^{(3)}$ to transform source phrase embeddings into the target-side semantic space, following Zhang et al. (2014) and Hermann and Blunsom (2014):

$$p'_{n_{\bar{e}}} = f(W_f^{(3)} p_{n_{\bar{f}}} + b_f^{(3)}) \quad (14)$$

here $p'_{n_{\bar{e}}}$ denotes the reconstructed vector representation of $n_{\bar{e}}$, which is transformed from the vector representation $p_{n_{\bar{f}}}$ of $n_{\bar{f}}$. Then, we repeat the reconstruction procedure in a top-down manner along the corresponding target tree structure until leaf nodes are reached, following Socher et al. (2011a). Specifically, given the vector representation $p'_{n_{\bar{e}}}$, we reconstruct vector representations of its two children nodes:

$$[c_{e1}^u; c_{e2}^u] = f(W_e^u p'_{n_{\bar{e}}} + b_e^u) \quad (15)$$

where c_{e1}^u and c_{e2}^u are the reconstructed vector representations of the children nodes, $W_e^u \in \mathbb{R}^{2d\times d}$, and $b_e^u \in \mathbb{R}^{2d\times 1}$. Eventually, given the original and reconstructed target phrase representations, we calculate $E_{f2e\cdot rec}(T_f, T_e; \theta)$ as follows:

$$E_{f2e\cdot rec}(T_f, T_e; \theta) = \frac{1}{2} \sum_{\langle n_{\bar{f}}, n_{\bar{e}}\rangle \in \mathcal{S}} \sum_{n \in T_{\bar{e}}} \| p_n - p'_n \|^2 \quad (16)$$

where p_n and p'_n are the original and reconstructed vector representations of node n in the sub-tree structure $T_{\bar{e}}$ rooted at $n_{\bar{e}}$. This error function will be minimized so that semantic differences between original and reconstructed structures are minimal.

Figure 3 demonstrates the structure reconstruction from a generated source sub-tree to its target counterpart. In this way, BCorrRAE propagates semantic information along dash lines sequentially until leaf nodes in the generated structure of the target phrase.

3.2.4 The Final Objective

Similar to Eq. (8), we define the final objective function of our model based on the three types of errors described above

$$\begin{aligned} J_{BCorrRAE} = \sum_{(f,e)\in\mathcal{D}} \{ & \alpha \left(E_{rec}(f;\theta) + E_{rec}(e;\theta)\right) \\ & + \beta \left(E_{con}(T_f;\theta) + E_{con}(T_e;\theta)\right) \\ & + \gamma \left(E_{f2e\cdot rec}(T_f, T_e;\theta) + E_{e2f\cdot rec}(T_f, T_e;\theta)\right)\} \\ & + R(\theta) \end{aligned} \quad (17)$$

where weights α, β, γ (s.t. $\alpha+\beta+\gamma=1$) are used to balance the preference among the three errors, and $R(\theta)$ is the regularization term. Parameters θ are divided into four sets[1]:

1. θ_L: the word embedding matrix;
2. θ_{rec}: the RAE parameter matrices $W^{(1)}$, $W^{(2)}$ and bias terms $b^{(1)}$, $b^{(2)}$ (Section 2.1);
3. θ_{con}: the consistency/inconsistency score parameter matrices W_{cns}^{score}, W_{inc}^{score} (Section 3.2.2);
4. θ_{clrec}: the cross-lingual RAE semantic transformation parameter matrices $W^{(3)}$, W^{u} and bias terms $b^{(3)}$, b^{u} (Section 3.2.3).

For regularization, we assign each parameter set a unique weight:

$$\begin{aligned} R(\theta) = &\frac{\lambda_L}{2}\|\theta_L\|^2 + \frac{\lambda_{rec}}{2}\|\theta_{rec}\|^2 \\ &+ \frac{\lambda_{con}}{2}\|\theta_{con}\|^2 + \frac{\lambda_{lcrec}}{2}\|\theta_{lcrec}\|^2 \end{aligned} \tag{18}$$

Additionally, in order to prevent the hidden layer from being very small, we normalize all output vectors of the hidden layer to have length 1, $p = \frac{p}{\|p\|}$, following Socher et al. (2011c).

3.3 Model Training

Similar to Zhang et al. (2014), we adopt a co-training style algorithm to train model parameters in the following two steps:

First, we use a normal distribution ($\mu = 0, \sigma = 0.01$) to randomly initialize all model parameters, and adopt the standard RAE to pre-train source- and target-side phrase embeddings and tree structures (Section 2.1).

Second, for each bilingual phrase, we update its source-side parameters to obtain the fine-tuned vector representation and binary tree of the source phrase, given the target-side phrase structure and node representations, and vice versa. In this process, we apply L-BFGS to tune parameters based on gradients over the joint error, as implemented in (Socher et al., 2011c).

We repeat the procedure of the second step until either the joint error (shown in Eq. (17)) reaches a local minima or the number of iterations is larger than a pre-defined number (25 is used in experiments).

[1]Note that the source and target languages have different four sets of parameters.

4 Decoding with BCorrRAE

Once the model training is completed, we incorporate two different phrasal similarity features built on the trained BCorrRAE into the standard log-linear framework of SMT. Given a bilingual phrase (f, e), we first obtain their semantic phrase representations (p_f, p_e). Then we transform p_f into p'_e in the target semantic space and p_e into p'_f in the source semantic space via transformation matrixes. Finally, we reconstruct sub-trees of p'_f along the source structure T_f learned by BCorrRAE, sub-trees of p'_e along the target structure T_e.

We exploit two kinds of phrasal similarity features based on the learned phrase representations and their tree structures as follows:

- *Semantic Similarity* measures the similarity between original and transformed phrase representations of (f, e):

$$\begin{aligned} Sim_{SM}(p_f, p'_f) &= \frac{1}{2}\|p_f - p'_f\|^2 \\ Sim_{SM}(p_e, p'_e) &= \frac{1}{2}\|p_e - p'_e\|^2 \end{aligned} \tag{19}$$

- *Structural Similarity* calculates the similarity between original and reconstructed tree structures learned by BCorrRAE for (f, e):

$$\begin{aligned} Sim_{ST}(p_f, p'_f) &= \frac{1}{2C_f}\sum_{n\in T_f}\|p_n - p'_n\|^2 \\ Sim_{ST}(p_e, p'_e) &= \frac{1}{2C_e}\sum_{n\in T_e}\|p_n - p'_n\|^2 \end{aligned} \tag{20}$$

where p_n and p'_n represent vector representations of original and reconstructed node n, and C_f and C_e count the number of nodes in the source and target tree structure respectively. Note that if we only compute the similarity for root nodes in the bilingual tree of (f, e), the structural similarity equals to the semantic similarity in Eq. (19).

5 Experiments

We conducted experiments on NIST Chinese-English translation task to validate the effectiveness of BCorrRAE.

5.1 System Overview

Our baseline decoder is a state-of-the-art phrase-based translation system equipped with a maximum entropy based reordering model (MEBTG). It adopts three bracketing transduction grammar rules (Wu, 1997; Xiong et al., 2006): merging

rules $A \rightarrow [A_1, A_2]|\langle A_1, A_2\rangle$ which are used to merge two neighboring blocks[2] A_1 and A_2 in a straight|inverted order, and lexical rule $A \rightarrow f/e$ used to translate a source phrase f into a target phrase e.

The MEBTG system features a maximal entropy classifier based reordering model that predicts orientations of neighboring blocks. During training, we extract bilingual phrases containing up to 7 words on the source side from the training corpus. With the collected reordering examples, we adopt the maximal entropy toolkit[3] developed by Zhang to train the reordering model with the following parameters: iteration number $iter$=200 and gaussian prior g=1.0. Following Xiong et al. (2006), we use only boundary words of blocks to trigger the reordering model.

The whole translation model is organized in a log-linear framework (Och and Ney, 2002). The adopted sub-models mainly include: (1) rule translation probabilities in two directions, (2) lexical weights in two directions, (3) targets-side word number, (4) phrase number, (5) language model score, and (6) the score of maximal entropy based reordering model. We perform *minimum error rate training* (Och, 2003) to tune various feature weights. During decoding, we set $ttable\text{-}limit$=20 for translation candidates kept for each source phrase, $stack\text{-}size$=100 for hypotheses in each span, and $swap\text{-}span$=15 for the length of the maximal reordering span.

5.2 Setup

Our bilingual data is the combination of the FBIS corpus and Hansards part of LDC2004T07 corpus, which contains 1.0M parallel sentences (25.2M Chinese words and 29M English words). Following Zhang et al. (2014), we collected 1.44M bilingual phrases using forced decoding (Wuebker et al., 2010) to train BCorrRAE from the training data. We used a 5-gram language model trained on the Xinhua portion of Gigaword corpus using *SRILM* Toolkits[4]. Translation quality is evaluated by case-insensitive BLEU-4 metric (Papineni et al., 2002). We performed paired bootstrap sampling (Koehn, 2004) to test the significance in BLEU score differences. In our experiments, we used NIST MT05 and MT06/MT08 data set as the development and test set, respectively.

Parameter	BRAE	BCorrRAE
α	0.119	0.121
β	-	0.6331
γ	-	0.2459
λ_L	4.95×10^{-5}	3.13×10^{-5}
λ_{rec}	2.64×10^{-7}	2.05×10^{-5}
λ_{con}	-	7.32×10^{-6}
λ_{lcrec}	9.31×10^{-5}	5.25×10^{-6}

Table 1: Hyper-parameters for BCorrRAE and BRAE model.

Method	d	MT06	MT08	AVG
$\mathbf{BCorrRAE}_{SM}$	25	30.81	$22.68^{\Downarrow}$	26.75
	50	$30.58^{\downarrow}$	$22.72^{\Downarrow}$	26.65
	75	30.50	$22.53^{\Downarrow}$	26.52
	100	$30.34^{\Downarrow}$	$22.61^{\Downarrow}$	26.48
$\mathbf{BCorrRAE}_{ST}$	25	30.56	23.28	26.92
	50	30.94	23.33	27.14
	75	30.73	23.40	27.07
	100	30.90	23.50	27.20

Table 2: Experiment results for different dimensions (d). BCorrRAE_{SM} and BCorrRAE_{ST} are our systems that are enhanced with the semantic and structural similarity features learned by BCorrRAE, respectively. $\downarrow/\Downarrow$: significantly worse than the BCorrRAE_{ST} with the same dimensionality ($p < 0.05/p < 0.01$).

In addition to the baseline described below, we also compare our method against the BRAE model, which focuses on modeling relations of source and target phrases as a whole unit. Word embeddings in BRAE are pre-trained with toolkit *Word2Vec*[5] (Mikolov et al., 2013) on large-scale monolingual data that contains 0.83B words for Chinese and 0.11B words for English.

Hyper-parameters in all neural models are optimized by *random search* (Bergstra and Bengio, 2012) based on related joint errors. We randomly extracted $250,000$ bilingual phrases from the above-mentioned training data as training set, $5,000$ as development set and another $5,000$ as test set. We drew α, β, γ uniformly from 0.10 to 0.50, and $\lambda_L, \lambda_{rec}, \lambda_{con}$ and λ_{lcrec} exponentially from 10^{-8} to 10^{-2}. Final parameters are shown in Table 1 for both BRAE and BCorrRAE.

5.3 Dimensionality of Embeddings

To investigate the impact of embedding dimensionality on our BCorrRAE, we tried four different dimensions from 25 to 100 with an increment of 25 each time. The results are displayed in Table 2. We can observe that the performance of our model is not consistently improved with the increment of dimensionality. This may be because a

[2] A block is a bilingual phrase without maximum length limitation.

[3] http://homepages.inf.ed.ac.uk/lzhang10/maxent_toolkit.html

[4] http://www.speech.sri.com/projects/srilm/download.html

[5] https://code.google.com/p/word2vec/

larger dimension brings in much more parameters, and therefore makes parameter tuning more difficult. In practice, setting the dimension d to 50, we can get satisfactory results without much computation effort, which has also been found by Zhang et al. (2014).

5.4 Structural Similarity vs. Semantic Similarity

Table 2 also shows that the performance of BCorrRAE$_{ST}$, the system with the structural similarity feature in Eq. (20), is always superior to that of BCorrRAE$_{SM}$ with the semantic similarity feature in Eq. (19). BCorrRAE$_{ST}$ is better than BCorrRAE$_{SM}$ by 0.483 BLEU points on average. In most cases, differences between BCorrRAE$_{ST}$ and BCorrRAE$_{SM}$ with the same dimensionality are statistically significant. This suggests that digging into structures of bilingual phrases (BCorrRAE$_{ST}$) can obtain further improvements over only modeling bilingual phrases as whole units (BCorrRAE$_{SM}$).

5.5 Overall Performance

Table 3 summarizes the comparison results of different models on the test sets. The BCorrRAE$_{SM}$ outperforms the baseline and BRAE by 1.06 and 0.25 BLEU points on average respectively, while BCorrRAE$_{ST}$ gains 1.55 and 0.74 BLEU points on average over the baseline and BRAE. The improvements of BCorrRAE$_{ST}$ over the baseline, BRAE and BCorrRAE$_{SM}$ are statistically significant at different levels. This demonstrates the advantage of our BCorrRAE over BRAE in that BCorrRAE is able to explore sub-structures of bilingual phrases.

5.6 Analysis

We compute a ratio of aligned nodes (Section 3.1) over all nodes to estimate how well tree structures of bilingual phrases generated by BRAE and BCorrRAE are consistent with word alignments. We consider two factors when computing the ratio: the length of the source side of a bilingual phrase l_s and the length of a span covered by an aligned node l_a. The result is illustrated in Table 4.[6] We find that BCorrRAE significantly outperforms BRAE model by 7.22% on average in terms of the aligned node ratio. This strongly demonstrates that the proposed BCorrRAE is able to generate tree structures that are more consistent with word alignments than those generated by BRAE.

[6]We only give ratios for bilingual phrases with source-side length from 3 to 4 words because 1) ratios of BRAE and BCorrRAE in the case of $l_a < 3$ are very close and 2) phrases with length > 4 are rarely used during decoding (accounting for $< 0.5\%$).

Method	MT06	MT08	AVG
Baseline	$29.66^{\Downarrow}$	$21.52^{\Downarrow}$	25.59
BRAE	$30.27^{\Downarrow}$	$22.53^{\Downarrow}$	26.40
BCorrRAE$_{SM}$	$30.58^{\downarrow}$	$22.72^{\Downarrow}$	26.65
BCorrRAE$_{ST}$	30.94	23.33	27.14

Table 3: Experiment results on the test sets. **AVG** = average BLEU scores for test sets. For both BRAE and BCorrRAE, we set d=50. $\downarrow/\Downarrow$: significantly worse than the BCorrRAE$_{ST}$ with d=50 ($p < 0.05$/$p < 0.01$, respectively).

$[l_s, l_a]$	[3,2]	[4,2]	[4,3]
BRAE	52.70%	39.88%	46.58%
BCorrRAE	60.08%	46.32%	54.43%

Table 4: Aligned node ratio for source phrases of different lengths.

We further show example source phrases in Table 5 with their most semantically similar translations learned by BRAE and BCorrRAE in the training corpus. Both models can select correct translations for content words. However, they are different in dealing with function words. Compared to our model, the BRAE model prefers longer target phrases surrounded with function words. Take the source phrase "严峻 挑战" as an example, the BRAE model learns both "*a serious challenge to*" and "*a serious challenge from*" as its semantically similar target phrases. Although the content words "严峻" and "挑战" are translated correctly into "*serious*" and "*challenge*", the function words "*to*" and "*from*" express exactly the opposite meanings. In contrast, our model, especially the BCorrRAE$_{ST}$ model, tends to choose shorter translations that are consistent with word alignments.

6 Related Work

A variety of efforts have been devoted to learning vector representations for words/phrases with deep neural networks. According to the difference of learning contexts, previous work mainly include the following two strands.

(1) *Monolingual Word/Phrase Embeddings.* The straightforward approach to represent word/phrases is to learn their hidden representations with traditional feature vectors, which requires manual and task-dependent feature engineering (Cui et al., 2014; Wu et al., 2014;

Source Phrase	BRAE	BCorrRAE$_{SM}$	BCorrRAE$_{ST}$
鼓吹 (advocate)	to advocate the in preaching the the promotion of	out to advocate been encouraging an advocate	encouraging claimed advocate
严峻 挑战 (serious challenge)	as well as severe challenges a serious challenge to a serious challenge from	of rigorous challenges as well as severe challenges of severe challenges	rigorous challenge enormous challenge severe challenge
公布 的 数据 (data released)	by the figures published by the the statistics released by data published by the	to the estimates announced at the figures published the statistics released by	published data released figures the estimates announced

Table 5: Semantically similar target phrases in the training set for example source phrases.

Chen and Manning, 2014). To avoid exploiting manually input features, Bengio et al. (2003) convert words to dense, real-valued vectors by learning probability distributions of n-grams. Mikolov et al. (2013) generate word vectors by predicting their limited context words. Instead of exploiting outside context information, recursive auto-encoder is usually adopted to learn the composition of internal words (Socher et al., 2010; Socher et al., 2011b; Socher et al., 2013b; Socher et al., 2013a). Recently, convolution architecture has drawn more and more attention due to its ability to explicitly capture short and long-range relations (Collobert et al., 2011; Kalchbrenner and Blunsom, 2013; Kalchbrenner et al., 2014; Kim, 2014).

(2) *Bilingual Word/Phrase Embeddings*. In the field of machine translation and cross-lingual information processing, bilingual embedding learning has become an increasingly important study. The bilingual embedding research origins in the word embedding learning, upon which Zou et al. (2013) utilize word alignments to constrain translational equivalence. Kočiský et al. (2014) propose a probability model to capture more semantic information by marginalizing over word alignments. More specifically to SMT, its main components have been exploited to learn better bilingual phrase embeddings in different aspects: language models (Wang et al., 2014; Garmash and Monz, 2014), reordering models (Li et al., 2013) and translation models (Tran et al., 2014; Zhang et al., 2014). Instead of exploiting a single model, Liu et al. (2014) combine the recursive and recurrent neural network to incorporate the language and translation model.

Different from the methods mentioned above, our model considers both the cross-language consistency of phrase structures and internal correspondence relations inside bilingual phrases. The most related works include Zhang et al. (2014) and Socher et al. (2011a). Compared with these works, our model exploits different levels of correspondence relations inside bilingual phrases instead of only the top level of entire phrases, and reconstructs tree structures of sub-phrases in one language according to aligned nodes in the other language, which, to the best of our knowledge, has never been investigated before.

7 Conclusions and Future Work

In this paper, we have presented the BCorrRAE to learn phrase embeddings and tree structures of bilingual phrases for SMT. Punishing structural-alignment-inconsistent sub-structures and minimizing the gap between original and reconstructed structures, our approach is able to not only generate alignment-consistent phrase structures, but also capture different levels of semantic relations within bilingual phrases. Experiment results demonstrate the effectiveness of our model.

In the future, we would like to derive more features from BCorrRAE, e.g., consistency/inconsistency scores of bilingual phrases, to further enhance SMT. Additionally, we also want to apply our model to other bilingual tasks, e.g., learning bilingual terminology or paraphrases.

Acknowledgments

The authors were supported by National Natural Science Foundation of China (Grant Nos 61303082 and 61403269), Natural Science Foundation of Jiangsu Province (Grant No. BK20140355), National 863 program of China (No. 2015AA015407), the Special and Major Subject Project of the Industrial Science and Technology in Fujian Province 2013 (Grant No. 2013HZ0004-1), and 2014 Key Project of Anhui Science and Technology Bureau (Grant No. 1301021018). We also thank the anonymous reviewers for their insightful comments.

References

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Janvin. 2003. A neural probabilistic language model. *Journal of Machine Learning Research*, 3:1139–1155.

James Bergstra and Yoshua Bengio. 2012. Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, 13:22–29.

Danqi Chen and Christopher Manning. 2014. A fast and accurate dependency parser using neural networks. In *Proc. of EMNLP 2014*, pages 740–750.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *Journal of Machine Learning Research.*, 16:2493–2537.

Lei Cui, Dongdong Zhang, Shujie Liu, Qiming Chen, Mu Li, Ming Zhou, and Muyun Yang. 2014. Learning topic representation for smt with neural networks. In *Proc. of ACL 2014*, pages 133–143.

Jacob Devlin, Rabih Zbib, Zhongqiang Huang, Thomas Lamar, Richard Schwartz, and John Makhoul. 2014. Fast and robust neural network joint models for statistical machine translation. In *Proc. of ACL 2014*, pages 1370–1380.

Jianfeng Gao, Xiaodong He, Wen-tau Yih, and Li Deng. 2014. Learning continuous phrase representations for translation modeling. In *Proc. of ACL 2014*, pages 699–709.

Ekaterina Garmash and Christof Monz. 2014. Dependency-based bilingual language models for reordering in statistical machine translation. In *Proc. of EMNLP 2014*, pages 1689–1700.

Karl Moritz Hermann and Phil Blunsom. 2014. Multilingual models for compositional distributed semantics. In *Proc. of ACL 2014*, pages 58–68.

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

Nal Kalchbrenner, Edward Grefenstette, and Phil Blunsom. 2014. A convolutional neural network for modelling sentences. In *Proc. of ACL 2014*, pages 655–665.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proc. of EMNLP 2014*, pages 1746–1751.

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

Tomáš Kočiský, Karl Moritz Hermann, and Phil Blunsom. 2014. Learning bilingual word representations by marginalizing alignments. In *Proc. of ACL 2014*, pages 224–229.

Peng Li, Yang Liu, and Maosong Sun. 2013. Recursive autoencoders for ITG-based translation. In *Proc. of EMNLP 2013*, pages 567–577.

Lemao Liu, Taro Watanabe, Eiichiro Sumita, and Tiejun Zhao. 2013. Additive neural networks for statistical machine translation. In *Proc. of ACL 2013*, pages 791–801.

Shujie Liu, Nan Yang, Mu Li, and Ming Zhou. 2014. A recursive recurrent neural network for statistical machine translation. In *Proc. of ACL 2014*, pages 1491–1500.

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

Franz Josef Och and Hermann Ney. 2002. Discriminative training and maximum entropy models for statistical machine translation. In *Proc. of ACL 2002*, pages 295–302.

Franz Josef Och and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51.

Franz Josef Och. 2003. Minimum error rate training in statistical machine translation. In *Proc. of ACL 2003*, pages 160–167. Association for Computational Linguistics.

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

Richard Socher, Christopher D. Manning, and Andrew Y. Ng. 2010. Learning continuous phrase representations and syntactic parsing with recursive neural networks. In *Proc. of NIPS 2010.*

Richard Socher, Eric H. Huang, Jeffrey Pennington, Andrew Y. Ng, and Christopher D. Manning. 2011a. Dynamic pooling and unfolding recursive autoencoders for paraphrase detection. In *Proc. of NIPS 2011.*

Richard Socher, Cliff Chiung-Yu Lin, Andrew Y. Ng, and Christopher D. Manning. 2011b. Parsing natural scenes and natural language with recursive neural networks. In *Proc. of ICML 2011.*

Richard Socher, Jeffrey Pennington, Eric H. Huang, Andrew Y. Ng, and Christopher D. Manning. 2011c. Semi-supervised recursive autoencoders for predicting sentiment distributions. In *Proc. of EMNLP 2011*, pages 151–161.

Richard Socher, John Bauer, Christopher D. Manning, and Ng Andrew Y. 2013a. Parsing with compositional vector grammars. In *Proc. of ACL 2013*, pages 455–465.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. 2013b. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proc. of EMNLP 2013*, pages 1631–1642.

Martin Sundermeyer, Tamer Alkhouli, Joern Wuebker, and Hermann Ney. 2014. Translation modeling with bidirectional recurrent neural networks. In *Proc. of EMNLP 2014*, pages 14–25.

Akihiro Tamura, Taro Watanabe, and Eiichiro Sumita. 2014. Recurrent neural networks for word alignment model. In *Proc. of ACL 2014*, pages 1470–1480.

Ke M. Tran, Arianna Bisazza, and Christof Monz. 2014. Word translation prediction for morphologically rich languages with bilingual neural networks. In *Proc. of EMNLP 2014*, pages 1676–1688.

Rui Wang, Hai Zhao, Bao-Liang Lu, Masao Utiyama, and Eiichiro Sumita. 2014. Neural network based bilingual language model growing for statistical machine translation. In *Proc. of EMNLP 2014*, pages 189–195.

Haiyang Wu, Daxiang Dong, Xiaoguang Hu, Dianhai Yu, Wei He, Hua Wu, Haifeng Wang, and Ting Liu. 2014. Improve statistical machine translation with context-sensitive bilingual semantic embedding model. In *Proc. of EMNLP 2014*, pages 142–146.

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

Joern Wuebker, Arne Mauser, and Hermann Ney. 2010. Training phrase translation models with leaving-one-out. In *Proc. of ACL 2010*, pages 475–484.

Deyi Xiong, Qun Liu, and Shouxun Lin. 2006. Maximum entropy based phrase reordering model for statistical machine translation. In *Proc. of ACL 2006*, pages 521–528.

Nan Yang, Shujie Liu, Mu Li, Ming Zhou, and Nenghai Yu. 2013. Word alignment modeling with context dependent deep neural network. In *Proc. of ACL 2013*, pages 166–175.

Jiajun Zhang, Shujie Liu, Mu Li, Ming Zhou, and Chengqing Zong. 2014. Bilingually-constrained phrase embeddings for machine translation. In *Proc. of ACL 2014*, pages 111–121.

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