A Variational Autoencoding Approach for Inducing Cross-lingual Word Embeddings

Liangchen Wei Zhi-Hong Deng*
Key Laboratory of Machine Perception (Ministry of Education),
School of Electronics Engineering and Computer Science, Peking University,
Beijing 100871, China
liangchen.wei@pku.edu.cn zhdeng@cis.pku.edu.cn

Abstract

Cross-language learning allows one to use training data from one language to build models for another language. Many traditional approaches require word-level alignment sentences from parallel corpora, in this paper we define a general bilingual training objective function requiring sentence level parallel corpus only. We propose a variational autoencoding approach for training bilingual word embeddings. The variational model introduces *a continuous latent variable* to explicitly model the underlying semantics of the parallel sentence pairs and to guide the generation of the sentence pairs. Our model restricts the bilingual word embeddings to represent words in *exactly the same* continuous vector space. Empirical results on the task of cross lingual document classification has shown that our method is effective.

1 Introduction

Distributed representations have become an increasingly important tool in machine learning. Typically, word embeddings are trained to represent words in a continuous space in an unsupervised way, which characterizes the lexico-semantic relations among words. In many NLP tasks, they prove to be high-quality features, in contrast to hand-crafted, and thus expensive features. Successful applications of distributed representations include sentence modeling[Bengio *et al.*, 2003], sentiment analysis[Socher *et al.*, 2011] and document classification[Klementiev *et al.*, 2012].

The use of distributed representations is motivated by the idea that they capture semantics and syntax as well as encoding similarity between words. Like words have synonyms in the same language, word pairs across language also share resembling semantics. Mikolov[Mikolov *et al.*, 2013] observed a strong similarity between vector spaces across languages and suggested a linear cross-lingual mapping between the two vector spaces is technically possible.

Recently, it is becoming more and more desirable to have unsupervised techniques which can learn useful syntactic and semantic features that are invariant to tasks or languages which we are interested in. With accurate joint-space embeddings of both language, one can develop abundant textual resources from language A to build models for language B. This is especially useful for transferring limited label information from high-resources to low-resources languages, and has been demonstrated effective for document classification by Klementiev[Klementiev *et al.*, 2012], whose model outperforms a strong phrase based machine translation baseline.

Defining a joint space objective function is crucial at the core of cross-lingual learning. Several models for training cross-lingual embeddings have been proposed, usually starting from a monolingual objective following cross-lingual objective as constraints. [Zou *et al.*, 2013] learned word embeddings of different languages in separate spaces with monolingual corpus and projected the embeddings into a joint space. [Mikolov *et al.*, 2013] learned a linear projection from one embedding space to another. It was extended by [Faruqui and Dyer, 2014], who simultaneously projected source and target language embeddings into a joint space using canonical correlation analysis. Bilingual autoencoder for bags-of-words (BAE)[AP *et al.*, 2014] reconstructed the bag-of-words representation of semantic equivalent sentence pairs. BilBOWA[Gouws *et al.*,] extended CBOW and skip-gram models with minimizing differences between parallel sentence pair representations. [Šuster *et al.*, 2016] learned multi sense word embeddings with discrete autoencoders. Many of these models can be viewed as instances of a more general framework for inducing cross-lingual word embeddings, which integrates monolingual objectives(similar words in each language have similar embeddings) and cross-lingual objectives(similar words across languages have similar representations).

Inspired by the recent advances[Kingma and Welling, 2013; Rezende *et al.*, 2014] in neural variational inference, we propose a variational autoencoding model(BiVAE) to cross-lingual learning. Unlike the framework mentioned above, we explicitly model the underlying semantics of bilingual sentence pairs(see Figure1). Similar to [Suzuki *et al.*, 2016], we make two assumptions about BiVAE:

- There exists a continuous latent variable z from this underlying semantic space.
- This variable z, guides the generative process of the equivalent sentence pairs x and y independently, i.e.

*Corresponding Author

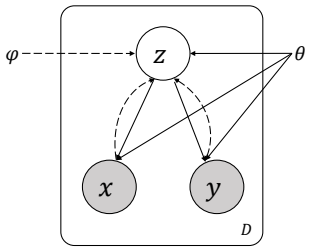

Figure 1: Illustration of BiVAE as a directed graph model. x and y denote the observable semantic equivalent sentence pairs. We use solid lines to denote generative model $p_\theta(x \mid z)$, $p_\theta(y \mid z)$ and dashed lines to denote the variational inference approximation $q_\phi(z \mid x, y)$. Both θ and ϕ are trainable parameters.

$$p(x, y \mid z) = p(x \mid z)p(y \mid z).$$

With this assumption, the following formulation characterizes our probabilistic language model:

$$p(x, y) = \int_z p(x, y \mid z)p(z)d_z \tag{1}$$

This brings the benefits that the latent variable z serves as a global variable capturing underlying semantics between parallel sentence pairs, thus forcing the words from different language to be embedded in a unified vector space without the help of extra alignment constraints. However the incorporation of it into the probabilistic language model brings one challenge: the posterior inference in this model is intractable. In order to address this issue, we propose the variational encoder-decoder model to cross-lingual learning, motivated by recent success of variational neural inference[Kingma and Welling, 2013; Rezende *et al.*, 2014]. We employ deep neural networks to model the posterior distribution of the latent hidden variable as they are capable of learning highly nonlinear functions thus making the inference tractable. In order to train model parameters efficiently, we apply a reparameterization trick[Kingma and Welling, 2013; Rezende *et al.*, 2014] on the variational lower bound which enables us using stochastic gradient optimization during training. Specifically, BiVAE has two essential components:

- A *variational inference network* infers the posterior distribution of z according to the encoded representation of parallel sentence pairs(i.e. $q_\phi(z \mid x, y)$).
- A *decoder network* reconstructs the observable sentences conditioned on the inferred distribution of z(i.e. $p_\theta(x \mid z)$, $p_\theta(y \mid z)$).

Model details will be introduced in section 3. We train the cross-lingual word embeddings with the proposed model and apply these embeddings to a standard document classification task and show that training with parallel corpus only, our model performs comparably with previous reported state-of-the-art models.

2 Background: Variational Autoencoder

In this section, we briefly review the VAE[Kingma and Welling, 2013; Rezende *et al.*, 2014]. The VAE is a generative model which is based on a regularized version of the standard autoencoder. It modifies the autoencoder architecture by replacing the deterministic function ϕ_{enc} with a learnable posterior recognition model, $q(z \mid x)$. The VAE imposes a prior distribution on the hidden variable z which enforces a regular geometry over the hidden representation and makes it possible to draw proper samples from the model using ancestral sampling. Intuitively, the VAE learns codes not as single points, but as soft ellipsoidal regions in latent space, forcing the codes to fill the space rather than memorizing the training data as isolated codes.

Given an observed variable x, the VAE introduces a latent variable z, and assumes that x is generated from z which can be characterized by the following formula:

$$p(x, z) = p_\theta(x \mid z)p(z) \tag{2}$$

where θ denotes the generative parameter of the model and $p(z)$ denotes the prior distribution of the latent variable z, e.g Gaussian distribution. $p_\theta(x \mid z)$ is the conditional distribution that models the generative process of x conditioned on the hidden variable z, typically estimated via a non-linear deep neural network.

The integration of z brings a challenge on the posterior inference and VAE adopts two techniques to tackle this problem: *variational neural inference* and *reparameterization*.

Variational neural inference employs deep neural network to approximate the posterior distribution of latent variable z, which is parameterized by a diagonal Gaussian distribution:

$$q_\phi(z \mid x) = \mathcal{N}(\mu(x), diag(\sigma^2(x))) \tag{3}$$

where mean $\mu(x)$ and variance $diag(\sigma^2(x))$ are both nonlinear functions of x parameterized with deep neural networks.

Reparameterization reparameterizes z as a function of μ and σ instead of using the standard sampling method. VAE introduces a standard Gaussian noise variable ϵ for generating samples from $q_\phi(z \mid x)$, namely the reprarameterization trick:

$$\tilde{z} = \mu + \sigma \odot \epsilon \tag{4}$$

VAE uses an objective which encourages the model to keep its posterior distribution of z close to its prior distribution. And this objective is a valid lower bound estimation on the true log likelihood of the data, making VAE a generative model. The objective takes the following form:

$$\begin{aligned}\mathcal{L}_{VAE}(\theta, \phi; x) = -KL(q_\phi(z \mid x) \mid\mid p(z)) + \\ \mathbb{E}_{q_\phi(z|x)}([logp_\theta(x \mid z)]) \leq logp(x)\end{aligned} \tag{5}$$

Maximizing the objective function is equivalent to maximizing the reconstruction likelihood of observable variable x and minimizing the Kullback-Leibler divergence between the approximated posterior and the prior distribution of latent variable z.

3 Bilingual Variational Autoencoder

In this section, we introduce our proposed model in detail. Formally, given the definition in Eq.1, the variational lower bound of BiVAE can be formulated as follows:

$$\begin{aligned}\mathcal{L}_{BiVAE}(\theta, \phi; x, y) = -KL(q_\phi(z \mid x, y) \mid\mid p(z)) \\ +\mathbb{E}_{q_\phi(z|x,y)}[logp_\theta(x \mid z) + logp_\theta(y \mid z)] \leq logp(x, y)\end{aligned} \tag{6}$$

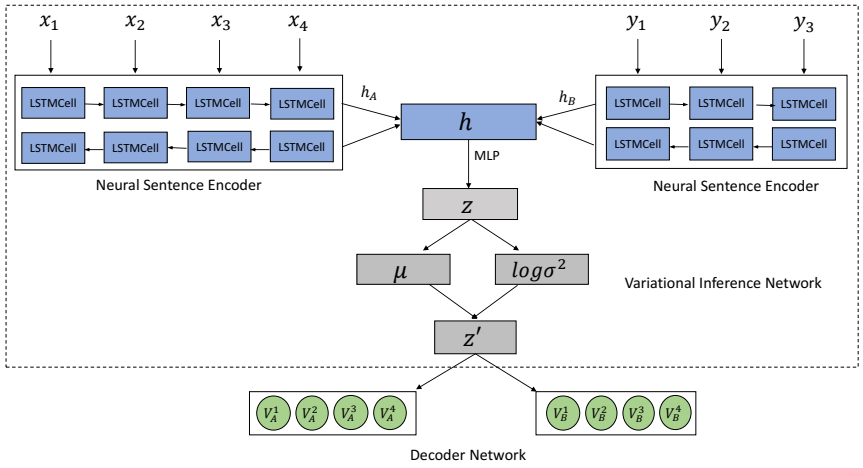

Figure 2: Illustration of core structure of BiVAE. Variational inference network approximates the posterior distribution of z based on representation h encoded by bidirectional LSTM encoder. Decoder network projects the common hidden codes to BoW representation of observable variables x and y.

where x and y denote observable semantic equivalent sentences from each language respectively, $p(z)$ is the prior distribution of latent variable z, which is Gaussian distribution here. $q_\phi(z \mid x, y)$ is our posterior approximator, $p_\theta(x \mid z)$ and $p_\theta(y \mid z)$ represent the conditional distribution of x and y conditioned on z. θ and ϕ are parameters of generative and inference neural networks respectively. Our goal is to train high quality bilingual word features with the proposed model.

3.1 Variational Inference Network

Neural sentence encoder

The neural encoders(see Figure2) aim at encoding the parallel sentence pairs into distributed representations. Following [Bowman *et al.*, 2015]'s approach, we adopt RNN to encode the parallel sentence pairs. But different from Bowman, who encodes the monolingual sentences using vanilla RNN, we adopt bidirectional LSTM as sentences are better summarized with context both forwards and backwards. Given an instance of parallel sequence pairs $[w_1^A, w_2^A, ..., w_{T_A}^A]$ and $[w_1^B, w_2^B, ..., w_{T_B}^B]$, the forward LSTM reads the sequence from left to right and the backward LSTM in the opposite direction:

$$\begin{aligned}
\overrightarrow{h}^i_A &= LSTMCell(\overrightarrow{h}^{i-1}_A, W^i_A) \\
\overleftarrow{h}^i_A &= LSTMCell(\overleftarrow{h}^{i+1}_A, W^i_A) \\
\overrightarrow{h}^i_B &= LSTMCell(\overrightarrow{h}^{i-1}_B, W^i_B) \\
\overleftarrow{h}^i_B &= LSTMCell(\overleftarrow{h}^{i+1}_B, W^i_B)
\end{aligned} \tag{7}$$

where $W_A \in \mathbb{R}^{|V_A| \times d_A}$ and $W_B \in \mathbb{R}^{|V_B| \times d_B}$ are lookup matrix for words in language A and B respectively. $\mid V_A \mid$ and $\mid V_B \mid$ denote vocabulary size, d_A and d_B denote dimension of the word embeddings, T_A and T_B denote length of sequence A and B respectively. $\overrightarrow{h}^i_A$, $\overleftarrow{h}^i_A$, $\overrightarrow{h}^i_B$ and $\overleftarrow{h}^i_B$ are hidden representations at position i generated in two directions. We call W_A and W_B lookup matrix as they learn word features in separate vector spaces, we use them only for sentence summarization. Finally, we concatenate the hidden states at the last time step of each LSTM encoder to represent the sentences:

$$\begin{aligned}
h_A &= [\overrightarrow{h}^{T_A}_A; \overleftarrow{h}^1_A] \\
h_B &= [\overrightarrow{h}^{T_B}_B; \overleftarrow{h}^1_B]
\end{aligned} \tag{8}$$

Variational neural inferer

Exactly modeling the posterior $p(z \mid x, y)$ is intractable. Conventional models typically employ the *mean field* approaches. However, due to its oversimplified assumption, it fails to capture the true posterior distribution of z. Following [Kingma and Welling, 2013], we employ neural networks to approximate the posterior distribution of z to get a tighter lower bound and assume the approximator has the following form:

$$q_\phi(z \mid x, y) = \mathcal{N}(z; \mu(f(h_A, h_B)), \sigma^2(f(h_A, h_B))I) \tag{9}$$

where mean μ and standard variance σ are outputs of neural networks based on sentence representations h_A and h_B. Function f projects the separate representations of parallel sentence pairs onto our concerned latent semantic space:

$$h = f(h_A, h_B) = g(W_z[h_A; h_B] + b_z) \tag{10}$$

where $W_z \in \mathbb{R}^{d_z \times (d_{h_A} + d_{h_B})}$ and $b_z \in \mathbb{R}^{d_z}$ are connection matrix and bias respectively, d_z is the dimensionality of the latent space, d_{h_A} and d_{h_B} are dimensionality of outputs of variational encoders. $g(\cdot)$ is elementwise activation function which we set Relu throughout our experiment.

We obtain the diagonal Gaussian distribution parameter μ and σ^2 through linear regression:

$$\mu = W_\mu h + b_\mu, log\sigma^2 = W_\sigma h + b_\sigma \tag{11}$$

where μ, $log\sigma^2$ are both d_z dimension vectors.

To obtain representations of z, we employ the same technique as VAE and reparameterize it as:

$$z' = \mu + \sigma \odot \epsilon, \epsilon \sim \mathcal{N}(0, I) \tag{12}$$

During encoding, we reconstruct the parallel sentence pairs based on sampled z' from $q_\phi(z \mid x, y)$

3.2 Decoder Network

Given the latent variable z, decoder network defines the probability over the observable variables x and y. By sampling z' from the posterior approximator $q_\phi(z \mid x, y)$, we are able to reconstruct x and y and estimate the expectation likelihood term. We represent the reconstructed parallel sentence pairs using one-hot BoW representations for the following reasons:

- By projecting the common distributed representations of parallel sentence pairs, namely z, to each language's vocabulary, where we introduce two connection matrix E_A and E_B which can be viewed as word embeddings, we obtain cross lingual representations of words embedded in exactly the same vector space.
- Acceleration. Large scale learning requires efficient approaches. As our goal is training cross lingual word embeddings instead of machine translation, we employ a softmax decoder by independently generating the words($z \rightarrow [\{x_i\}, \{y_i\}]$).
- Vanilla LSTM decoder is sensitive to subtle variation in the hidden states, thus making it hard to train the neural sentence encoder.

Inspired by [Miao *et al.*, 2016], the conditional probability over observable variables x and y is modelled by multinomial logistic regression with parameter shared across sentence pairs:

$$\begin{aligned} p_\theta(x, y \mid z) =& p_\theta(x \mid z) p_\theta(y \mid z) \\ =& \prod_{i=1}^{|T_A|} \frac{exp\{zE_A x_i + b_A^i\}}{\sum_{j=1}^{|V_A|} exp\{zE_A x_j + b_A^i\}} \cdot \\ & \prod_{i=1}^{|T_B|} \frac{exp\{zE_B y_i + b_B^i\}}{\sum_{j=1}^{|V_B|} exp\{zE_B y_j + b_B^i\}} \end{aligned} \tag{13}$$

where $E_A \in \mathbb{R}^{d_z \times |V_A|}$ and $E_B \in \mathbb{R}^{d_z \times |V_B|}$ learns joint-space semantic word embeddings and b_A, b_B represent the bias terms.

We use Monte Carlo method to estimate the expectation term over the posterior in Eq.6 which typically has less variance than the generic estimator. The training objective for an instance of parallel sentence pair (x, y) is defined as follows:

$$\begin{aligned} \tilde{L}_{BiVAE}(\phi, \theta, x, y) =& -D_{KL}(q_\phi(z \mid x, y) \mid\mid p(z)) \\ &+ \frac{1}{L} \sum_{l=1}^{L} (logp_\theta(x \mid z'_l) + logp_\theta(y \mid z'_l)) \end{aligned} \tag{14}$$

where $z'_l = \mu + \sigma \odot \epsilon_l$, $\epsilon_l \sim \mathcal{N}(0, I)$. L is the number of samples.

The first term is the Kullback-Leibler divergence between the posterior and prior distribution of z, which can be integrated analytically[Kingma and Welling, 2013]. The KL-divergence term can be interpreted as regularizing ϕ, encouraging the approximate posterior to be close to the prior $p(z)$. And the second term can be interpreted as the negative expected reconstruction error of x and y. Since the objective in Eq.14 is differentiable, we can jointly optimize the model parameters θ and ϕ using stochastic gradient optimization.

The model is implemented using Tensorflow[Abadi *et al.*, 2016], the model parameters of both generative process and posterior estimator are trained jointly using ADAM[Kingma and Ba, 2014]. We also apply dropout and batch normalization[Srivastava *et al.*, 2014; Ioffe and Szegedy, 2015] to the neural networks introduced in our model to reduce overfitting. We use 200 units for LSTM memory cell and 40 units for latent variable z, consequently 40 units for the word embeddings.

4 Experiment

In this section we present experiments which evaluate the quality of induced cross-lingual representations. We evaluate the embeddings in a standard cross-lingual document classification task which tests semantic transfer of information across languages.

4.1 Cross Lingual Document Classification

We use an exact replication of the cross-lingual document classification(CLDC) setup introduced by [Klementiev *et al.*, 2012] to evaluate the embeddings. The CLDC task setup is as follows: A labeled data set of documents in some language A is available to train a classifier, however we are interested in classifying documents in another language B at test time. As Klementiev noted, the aim of this task is not to provide a state-of-the-art document classifier but rather to examine the validity of joint semantic space model. Specifically, we train an averaged perceptron classifier[Freund and Schapire, 1999] on the labelled training data in the source language and then attempt to apply the classifier to the target data. Documents are represented as the tf-idf weighted sum of the embedding vectors of the words that appear in the documents.

4.2 Dataset and Setup

As our joint space model utilizes parallel corpus only, we train the bilingual embeddings for the English-German language pair using Europarl v7 parallel corpus[Koehn, 2005], and use the induced representations to classify a subset of the English and German sections of the Reuters RCV1/RCV2 multilingual corpora[Lewis *et al.*, 2004] that are assigned to only one of four categories: CCAT (Corporate/Industrial), ECAT (Economics), GCAT (Government/Social), and MCAT (Markets).

We preprocess the corpus by lowercasing all the words, removing all the punctuations and replacing all the digits with 0. We select the top words by term frequency appearing in the corpus and keep the vocabulary size as $\mid V^{EN} \mid$=40000 and $\mid V^{DE} \mid$= 50000.

For the classification experiment, 15000 documents(for each language) were selected randomly by Klementiev[Klementiev *et al.*, 2012] from RCV1/RCV2 corpus. One third

Model	Data	$en \rightarrow de$	$de \rightarrow en$
Majority Baseline	-	46.8	46.8
MT Baseline	Europarl	68.1	67.4
Klementiev et al.	Europarl+RCV	77.6	71.1
BiCVM	Europarl	83.7	71.4
BAE	Europarl+RCV	91.8	74.2
BilBOWA	Europarl+RCV	86.5	75.0
BiSkip	Europarl	90.7	80.3
CLSim	Europarl+RCV	**92.7**	80.1
BiVAE	Europarl	91.0	**80.4**

Table 1: Accuracy of cross lingual document classification. We compare BiVAE embeddings to the best models from past work. Data denotes the corpus utilized for inducing bilingual word embeddings. Numbers in boldface highlight the best scores per metric

of the selected documents(5000) were used as test sets and a varying size between 100 and 10000 of the remainder were used as training set. Another 1000 documents were kept as development set for hyper-parameter tuning. A multi class document classifier was trained for 10 epoch with English documents and is used to classify German documents and vice versa.

4.3 Results

Table1 summarizes results on the task of CLDC. We compare the performance of our model with some baselines and previous work. The *Majority Class* is a system where we simply classify the test documents as the class with the most training examples. The *MT* is a phrase-based machine translation system where test documents were translated into the same language as training documents. We also summarize some of the previous work. [Klementiev *et al.*, 2012] proposed to train two neural network languages models simultaneously along with a regularization term that encourages pairs of frequently aligned words to have similar word embeddings. BiCVM[Hermann and Blunsom, 2013] learned word embeddings via minimizing the compositional representations between parallel sentence pairs. BAE[AP *et al.*, 2014] reconstructed the bag-of-words representation of semantic equivalent sentence pairs to learn word embeddings. Bi-Skip[Luong *et al.*, 2015] extended skip-gram model to bilingual circumstances where separate context of aligned word pairs were jointly predicted. [Shi *et al.*, 2015] proposed a matrix cofactorization framework, CLSim, for learning cross lingual embeddings.

From Table1 we conclude that our proposed model outperforms most previous work and performs comparably with the previous state-of-the-art model, CLSim, with BiVAE tops the accuracy of submission $de \rightarrow en$. But note that our model is trained with Europarl corpus only while CLSim induce the embeddings using both Europarl and RCV corpus.

Different from those approaches which start from monolingual objective learning embeddings in separate vector space following cross lingual objective as constraints, BiVAE learns word representations embedded in *exactly* the same vector space without the help of explicit alignments. While on the other hand, one drawback of BiVAE is that the learned features can not be enhanced with monolingual corpus due to the structure of its probabilistic language model.

Word	Language	Nearest neighbours
january	En	january, february, march
	De	januar, jänner, april
oil	En	oil, crude, slick
	De	Öl, erdöl, rohöl
man	En	man, woman, person
	De	mann, mensch, mannes
economy	En	economy, economics
	De	wirtschaft, weltwirtschaft
microsoft	En	microsoft, google, intel
	De	microsoft, google, intel
president	En	president, mr
	De	präsident, herr
market	En	market, internal
	De	markt, binnenmarkt
great	En	great, deal, huge
	De	großes, großen, enorme
communication	En	communication, feedback
	De	kommunikation, mitteilung
law	En	law, rule
	De	gesetz, rechtsstaat

Table 2: Example English words with nearest words in English (En) and German (De) measured by Euclidean distance.

4.4 Qualitative Examples and Visualization

We find, for each English word, a list of top several English and German words closest to it based on Euclidean distance in a learned joint bilingual vector space. Our list of words includes { *january, oil, man, economy, microsoft, president, market, great, communication, law*}. Table 2 illustrates the properties captured within and across languages. Bilingually, our embeddings succeed in selecting the 1-best translations for all words in the list. Monolingually, our embeddings possess a clearly good clustering structure, which reveals topic nature of the semantic vector sapce.

Figure 3 gives a visualization of the joint vector space using t-SNE[Maaten and Hinton, 2008]. The English and German words which are translations of each other are represented by almost the same point in the vector space, revealing the semantic validity of the joint vector space.

5 Related Work

5.1 Cross-lingual Learning

Overall, approaches for training bilingual word embeddings can be categorized into several classes: *offline mapping, monolingual adaption and parallel training*.

In *offline mapping*, word representations are first trained on each language independently and a mapping is then learned to transform representations from one language to another. The advantage of this approach is its speed as no further training of word representations is required given monolingual word embeddings. Word embeddings trained in this approach include [Mikolov *et al.*, 2013] which utilizes translation pairs

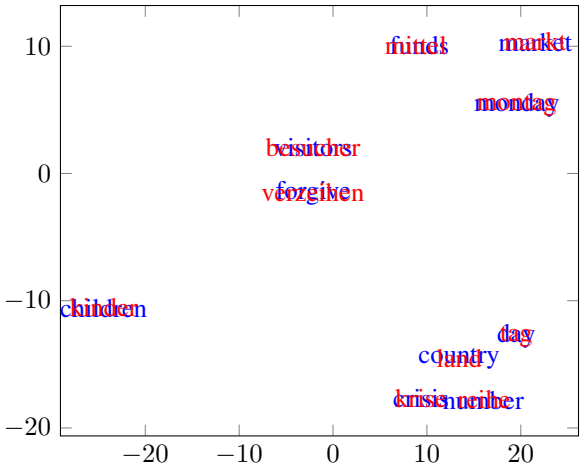

Figure 3: Visualization of joint semantic space using t-SNE. We represent 10 high frequency En-De word translation pairs.

to learn linear mapping.

Monolingual adaption jointly optimize the monolingual objectives of each language, with cross-lingual objective enforced as cross-lingual regularizer. One advantage of this approach is that it utilizes monolingual corpus in addition to parallel corpus to enhance the learned features.

Unlike previous schemes fix representations on either one or both sides, *Parallel training* leverage sentence aligned parallel corpus only and train a model to learn similar representations for aligned sentences. [Hermann and Blunsom, 2013; AP *et al.*, 2014] and the model proposed in this paper follow this approach. One advantage of this approach is that models can be designed to train word representations embedded in exactly the same continuous vector space, avoiding explicit alignment.

As introduced in section 1, cross-lingual word embeddings can be applied to various NLP tasks, including semantic applications such as cross-lingual dictionary induction[Vulic and Moens, 2013a; Mikolov *et al.*, 2013] and CLDC[Klementiev *et al.*, 2012] as well as syntactic applications such as cross-lingual syntactic dependency parsing[Täckström *et al.*, 2012] and lexicon extraction[Vulic and Moens, 2013b]. We prove our embeddings effective in CLDC task but its application remains to be explored in other cross-lingual NLP tasks.

5.2 Variational Neural Inference

In order to perform efficient inference and learning in generative probabilistic models on large-scale dataset, variational autoencoder was recently proposed by [Kingma and Welling, 2013; Rezende *et al.*, 2014]. Different from conventional *mean field* approximation, VAE employs neural network to approximate the posterior distribution of latent variable and optimize the model parameters with a reparameterized variational lower bound using the stochastic gradient optimization technique. Following Kingma, semi-supervised VAE has been proposed to model labeled dataset. Variational RNN has been proposed to deal with sequential data, which have been proved successful in speech modeling.

Variational neural inference has also shown strong performance in text processing. [Miao *et al.*, 2016] proposes a generic variational inference framework for generative and conditional models of text. [Bowman *et al.*, 2015] imposes a prior distribution on the hidden states of a standard RNN language model, helping generating sentences from the latent semantic space. [Zhang *et al.*, 2016] introduces a latent variable z to a standard neural machine translation framework to guide the generation of target translations. To the best of our knowledge, we are the first to introduce this technique to learn cross-lingual word embeddings.

6 Conclusion

In this paper, we propose a variational encoder-decoder framework for cross-lingual learning. By introducing the hidden variable z, we learn cross-lingual word embeddings in exactly the same continuous vector space instead of projecting them from separate spaces. We also conduct a standard CLDC task to evaluate BiVAE. Experiment results show that BiVAE performs comparably with the previous reported state-of-the-art model.

For future work, we are interested in modifying the variational neural decoder, e.g. LSTM decoder, to generate plausible parallel sentence pairs from the latent semantic space. With limited training corpus, we can train BiVAE to help generate more semantic equivalent sentence pairs to enrich the corpus in an unsupervised way.

Acknowledgements

This work is partially supported by the National High Technology Research and Development Program of China (Grant No. 2015AA015403). We would also like to thank the anonymous reviewers for their helpful comments.

References

[Abadi *et al.*, 2016] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, et al. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467*, 2016.

[AP *et al.*, 2014] Sarath Chandar AP, Stanislas Lauly, Hugo Larochelle, Mitesh Khapra, Balaraman Ravindran, Vikas C Raykar, and Amrita Saha. An autoencoder approach to learning bilingual word representations. In *Advances in Neural Information Processing Systems*, pages 1853–1861, 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(Feb):1137–1155, 2003.

[Bowman *et al.*, 2015] Samuel R Bowman, Luke Vilnis, Oriol Vinyals, Andrew M Dai, Rafal Jozefowicz, and Samy Bengio. Generating sentences from a continuous space. *arXiv preprint arXiv:1511.06349*, 2015.

[Faruqui and Dyer, 2014] Manaal Faruqui and Chris Dyer. Improving vector space word representations using multilingual correlation. Association for Computational Linguistics, 2014.

[Freund and Schapire, 1999] Yoav Freund and Robert E Schapire. Large margin classification using the perceptron algorithm. *Machine learning*, 37(3):277–296, 1999.

[Gouws *et al.*,] S Gouws, Y Bengio, and G Corrado. Bilbowa: fast bilingual distributed representations without word alignments (2014). *arXiv preprint arXiv:1410.2455*.

[Hermann and Blunsom, 2013] Karl Moritz Hermann and Phil Blunsom. Multilingual distributed representations without word alignment. *arXiv preprint arXiv:1312.6173*, 2013.

[Ioffe and Szegedy, 2015] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

[Kingma and Ba, 2014] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[Kingma and Welling, 2013] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[Klementiev *et al.*, 2012] Alexandre Klementiev, Ivan Titov, and Binod Bhattarai. Inducing crosslingual distributed representations of words. 2012.

[Koehn, 2005] Philipp Koehn. Europarl: A parallel corpus for statistical machine translation. In *MT summit*, volume 5, pages 79–86, 2005.

[Lewis *et al.*, 2004] David D Lewis, Yiming Yang, Tony G Rose, and Fan Li. Rcv1: A new benchmark collection for text categorization research. *Journal of machine learning research*, 5(Apr):361–397, 2004.

[Luong *et al.*, 2015] Thang Luong, Hieu Pham, and Christopher D Manning. Bilingual word representations with monolingual quality in mind. In *Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing*, pages 151–159, 2015.

[Maaten and Hinton, 2008] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(Nov):2579–2605, 2008.

[Miao *et al.*, 2016] Yishu Miao, Lei Yu, and Phil Blunsom. Neural variational inference for text processing. In *Proc. ICML*, 2016.

[Mikolov *et al.*, 2013] Tomas Mikolov, Quoc V Le, and Ilya Sutskever. Exploiting similarities among languages for machine translation. *arXiv preprint arXiv:1309.4168*, 2013.

[Rezende *et al.*, 2014] Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. *arXiv preprint arXiv:1401.4082*, 2014.

[Shi *et al.*, 2015] Tianze Shi, Zhiyuan Liu, Yang Liu, and Maosong Sun. Learning cross-lingual word embeddings via matrix co-factorization. In *ACL (2)*, pages 567–572, 2015.

[Socher *et al.*, 2011] Richard Socher, Jeffrey Pennington, Eric H Huang, Andrew Y Ng, and Christopher D Manning. Semi-supervised recursive autoencoders for predicting sentiment distributions. In *Proceedings of the conference on empirical methods in natural language processing*, pages 151–161. Association for Computational Linguistics, 2011.

[Srivastava *et al.*, 2014] Nitish Srivastava, Geoffrey E Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(1):1929–1958, 2014.

[Šuster *et al.*, 2016] Simon Šuster, Ivan Titov, and Gertjan van Noord. Bilingual learning of multi-sense embeddings with discrete autoencoders. *arXiv preprint arXiv:1603.09128*, 2016.

[Suzuki *et al.*, 2016] Masahiro Suzuki, Kotaro Nakayama, and Yutaka Matsuo. Joint multimodal learning with deep generative models. *arXiv preprint arXiv:1611.01891*, 2016.

[Täckström *et al.*, 2012] Oscar Täckström, Ryan McDonald, and Jakob Uszkoreit. Cross-lingual word clusters for direct transfer of linguistic structure. In *Proceedings of the 2012 conference of the North American chapter of the association for computational linguistics: Human language technologies*, pages 477–487. Association for Computational Linguistics, 2012.

[Vulic and Moens, 2013a] Ivan Vulic and Marie-Francine Moens. Cross-lingual semantic similarity of words as the similarity of their semantic word responses. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT 2013)*, pages 106–116. ACL, 2013.

[Vulic and Moens, 2013b] Ivan Vulic and Marie-Francine Moens. A study on bootstrapping bilingual vector spaces from non-parallel data (and nothing else). In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP 2013)*, pages 1613–1624. ACL, 2013.

[Zhang *et al.*, 2016] Biao Zhang, Deyi Xiong, Jinsong Su, Hong Duan, and Min Zhang. Variational neural machine translation. *arXiv preprint arXiv:1605.07869*, 2016.

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