A Distribution-based Model to Learn Bilingual Word Embeddings

Hailong Cao[1], **Tiejun Zhao**[1], **Shu Zhang**[2], **Yao Meng**[2]
[1]Harbin Institute of Technology, Harbin, China
[2]Fujitsu Research and Development Center, Beijing, China
{hailong,tjzhao}@mtlab.hit.edu.cn
{zhangshu,mengyao}@cn.fujitsu.com

Abstract

We introduce a distribution based model to learn bilingual word embeddings from monolingual data. It is simple, effective and does not require any parallel data or any seed lexicon. We take advantage of the fact that word embeddings are usually in form of dense real-valued low-dimensional vector and therefore the distribution of them can be accurately estimated. A novel cross-lingual learning objective is proposed which directly matches the distributions of word embeddings in one language with that in the other language. During the joint learning process, we dynamically estimate the distributions of word embeddings in two languages respectively and minimize the dissimilarity between them through standard back propagation algorithm. Our learned bilingual word embeddings allow to group each word and its translations together in the shared vector space. We demonstrate the utility of the learned embeddings on the task of finding word-to-word translations from monolingual corpora. Our model achieved encouraging performance on data in both related languages and substantially different languages.

1 Introduction

Learning word vector representations based on neural network is now a ubiquitous technique in natural language processing tasks and applications. Tremendous advances have been brought by distributed representations to the state-of-the-art methods (Bengio et al., 2003; Collobert and Weston, 2008; Mikolov et al., 2013a; Pennington et al., 2014). In these models, words are represented by *dense real-valued low-dimensional vectors* referred to as word embeddings learned from raw text. Distributed representations have the property that similar words are represented by similar vectors and thus can achieve better generalization. Such representations are usually learned from monolingual data and therefore might not be generalized well across different languages.

In order to learn useful syntactic and semantic features that are invariant to languages, several models for learning cross-lingual representations have been proposed and achieved impressive effects by incorporating cross-lingual distributional information (Klementiev et al., 2012; Zou et al., 2013; Chandar et al., 2014; Faruqui and Dyer, 2014; Hermann and Blunsom, 2014; Gouws et al., 2015; Luong et al., 2015; Shi et al., 2015; Vulić and Moens, 2015; Upadhyay et al., 2016). In the cross-lingual settings, similar representations are desired for words denoting similar concepts in different languages (e.g., the embeddings of the English word *computer* and the French word *ordinateur* should be similar). Cross-lingual representations are especially useful for many natural language processing tasks such as machine translation (Zou et al., 2013; Zhang et al., 2014), computing cross-lingual word similarity (Zhang et al., 2016) and transferring knowledge from high-resource languages to low-resource languages (Guo et al., 2015), etc.

However, all these cross-lingual models require some form of cross-lingual supervision such as seed lexicon,word-level alignments, sentence-level alignments and document-level alignments. Reliance on supervision might limit the development and application of cross-lingual representations. In this paper, we proposed a distribution based model to learn bilingual word embeddings from monolingual data. The

Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers, pages 1818–1827, Osaka, Japan, December 11-17 2016.

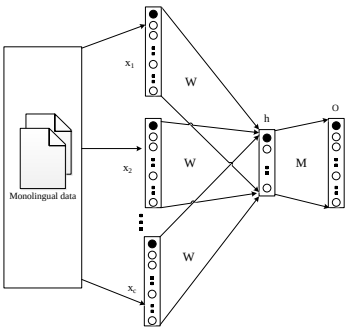

Figure 1: The CBOW architecture predicts the current word based on the context. It is a monolingual representations learning model.

proposed approach is complementary to the existing methods that rely on supervision. Our contributions are the following:

- We introduce a novel cross-lingual objective which is employed to match the distributions of monolingual embeddings as they are being trained in an online setting. Our model only requires monolingual data and therefore could be applied to any languages or domains that we are interested in.
- Our model can capture the common regularity shared by natural languages. The resulting bilingual embeddings allow to group a word and its translations together in vector space. We demonstrate the utility of our model on the task of finding word-to-word translations solely from monolingual corpora. Our model achieved encouraging performance on both related languages and substantially different languages.

2 Monolingual Word Embeddings Learning

Our framework is general enough to be built based on any monolingual embedding learning model. We adopt the popular continuous bag-of-Words (CBOW) model (Mikolov et al., 2013a) to demonstrate our approach. The CBOW model uses continuous distributed representation of the context where each word is mapped to a learned vector. The architecture is shown in the Figure 1. The training data D is a set of pairs in the form of (x, y), in which y is a word and $x = (x_1, x_2, ..., x_C)$ is a set containing C context words in which y appears. We have $y, x_i \in (1, 2, ..., V)$, where V is the vocabulary size. The training criterion is to seeking parameters minimizing the loss function which is the negative log probability of y given x:

$$\hat{W}, \hat{M} = \underset{W,M}{\operatorname{argmin}} \sum_{(x,y)\in D} L(W, M, x, y) = \underset{W,M}{\operatorname{argmin}} \sum_{(x,y)\in D} -log(P(y|x, W, M)) \tag{1}$$

We use the one-hot V-dimension column vector $\vec{x}_i$ to refer the context word x_i in which only the x_i^{th} unit is 1, and all other units are 0. W is a $K \times V$ matrix representing the weights between the input layer and the K-dimensional hidden layer. Each column of W is the K-dimensional vector representation of the associated word of the input layer. W transforms each context word x_i into a K-dimension real value vector. Each context word x_i is mapped into a real value K-dimensional vector by W. The CBOW model takes the average of these vectors as the value of hidden layer.

$$\vec{h} = \frac{1}{C} \sum_{i=1}^{C} W\vec{x}_i \tag{2}$$

where the matrix W is shared by all context words.

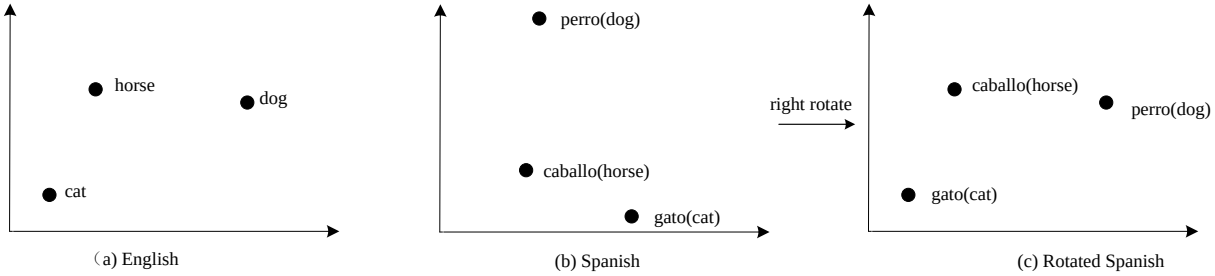

Figure 2: Monolingual embeddings have been shown to have similar geometric shape (a & b). It is desired to have more explicit similarity (a & c). (Mikolov et al., 2013b) achieved this by leaning a linear projection with a bilingual dictionary.

From the hidden layer to the output layer, there is a $V \times K$ weight matrix M by which the hidden vector is mapped into a V-dimensional output vector:

$$\vec{O} = M\vec{h} \tag{3}$$

which will be normalized by the soft-max function as a distribution over V candidate words. Finally, the probability of the word y given its context x is:

$$P(y|x, W, M) = \frac{exp(O_y)}{\sum_{i=1}^{V} exp(O_i)} \tag{4}$$

where O_i is the i^{th} unit of vector $\vec{O}$.

The parameters of W and M are tuned by the standard stochastic gradient descent (SGD) algorithm. There are very interesting properties in the learned word vectors. For example, similar words are nearby vectors in a vector space. And more importantly, if vectors learned for languages are manually rotated, Mikolov et al. (2013b) observed that languages share similar geometric arrangements in vector spaces (shown in Figure 2). The reason is that all common languages share universal structure of human lexical semantics(Youn et al., 2016). To capture the similarities, they use a bilingual dictionary to learn a linear projection between vectors learned independently from each language.

Our work is also motivated by the observation in (Mikolov et al., 2013b). Rather than relying on geometric transformation, we focus on word embeddings learning itself and explore an joint bilingual learning framework.

3 Learning Bilingual Word Embeddings by Distribution Matching

In the cross-lingual setup, we desire word embeddings to be generalized well across different languages. For example, given embedding of English context words {*the, cats, on, the, mat*}, cross-lingual models should not only be able to predict that the current word can be *sits* in English, but also be able to predict it can be *assis* in French. Similarly, given embeddings of French context words {*le, chat, est, sur, ma*}, cross-lingual models should be able to predict both *sits* and *assis*. Such desire bears a strong resemblance to the problem of domain adaptation which is well-studied in the field of natural language processing (Blitzer et al., 2006; Daume III, 2007). We apply the theory on domain adaptation which can learn cross-domain features to the task of learning cross-lingual features(word embeddings). Before we detail the proposed framework for unsupervised cross-lingual representation learning, we briefly introduce methods in domain adaptation.

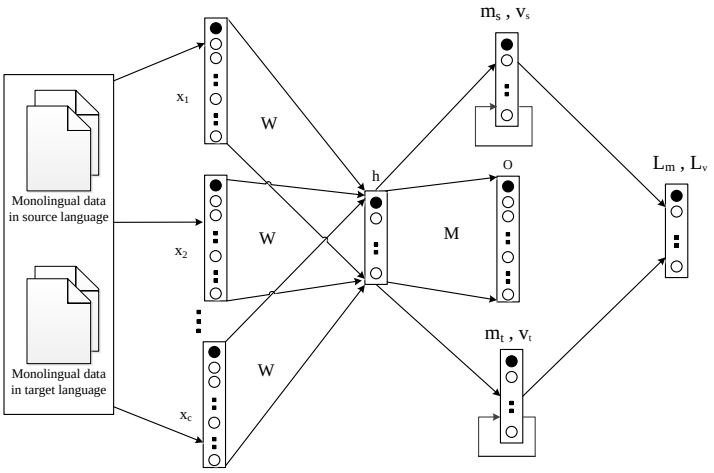

Figure 3: Bi-lingual representation learning is achieved by matching the distribution of the source and target languages. The dynamic statistic of hidden states of the source and target languages are calculated on line, and the dissimilarities between them are minimized through standard back propagation algorithm.

3.1 Domain Adaptation

This work is inspired by theory on domain adaptation (Ben-David et al., 2006; Ben-David et al., 2010) which suggest that, for effective domain transfer, predictions must be made based on data representations that cannot discriminate the source and target domains. Based on this theory, very simple and efficient domain adaptation approaches have been developed (Ajakan et al., 2014; Ganin and Lempitsky, 2015) for representation learning in neural networks. The main idea is matching feature space distributions of source and target domain. To this end, Ganin and Lempitsky (2015) added a domain classifier connected to the feature extractor. Feature distributions over the two domains are made similar (as indistinguishable as possible for the domain classifier), thus resulting in the domain-invariant features.

A straightforward way to learn cross-lingual representations is extending the ideas of Ganin and Lempitsky (2015) by replaceing their domain classifer with a language classifier. Alternatively, in this paper we explore a much more direct approach.

3.2 Model Architecture

We observed that, unlike features in image processing which are *high-dimensional*, language representations are usually in form of *dense real-valued low-dimensional* vector and therefore the distribution of them can be accurately estimated, given the freely available large scale raw text data. Based on this observation, we propose a novel learning objective which directly matches the distributions of word embeddings in one language with that in the other language. An overview of the architecture of bilingual word embeddings learning is given in Figure 3.

Same to that in the CBOW model, W and M are still $K \times V$ and $V \times K$ matrixes. But, now we have $V = V_s + V_t$ where V_s and V_t is the vocabulary size of source and target data respectively. All what we add to the CBOW model are the statistics of hidden states. We assume that the distribution of hidden states of each language is subject to a multi-dimensional normal distribution. So we have two statistics for each language, namely the mean $\vec{m}_s$ and the variance $\vec{v}_s$ on the source side, the mean $\vec{m}_t$ and the variance $\vec{v}_t$ on the target side. Like the hidden states, each statistic is also a K-dimensional vector. We use D_s and D_t to denote the nonparallel monolingual training data set from source language and target language respectively. Mathematically, on the source data, the mean is defined as:

$$\vec{m}_s = \frac{1}{|D_s|} \sum_{(x,y) \in D_s} \vec{h}(x) \tag{5}$$

where the hidden state h is defined in equation 2. And the variance is:

$$\vec{v}_s = \frac{1}{|D_s|} \sum_{(x,y) \in D_s} (\vec{h}(x) - \vec{m}_s)^2 \tag{6}$$

where the square operation is performed on each element of the K-dimensional vector respectively. On the target data, $\vec{m}_t$ and $\vec{v}_t$ are defined in the same way.

In order to encourage the source and the target data to have similar distributions in the shared space, one can directly minimize the dissimilarity between statistics of the source and the target data. More formally, the bilingual training criterion is to seeking parameters minimizing the standard monolingual objective of all data and the distribution dissimilarity:

$$\hat{W}, \hat{M} = \underset{W,M}{\operatorname{argmin}} (\sum_{(x,y) \in D_s \cup D_t} L(W, M, x, y) + \lambda_m L_m(\vec{m}_s, \vec{m}_t) + \lambda_v L_v(\vec{v}_s, \vec{v}_t)) \tag{7}$$

where L is defined in equation 1. L_m and L_v are cross-lingual objectives which are defined as the dissimilarities between statistics:

$$L_m(\vec{m}_s, \vec{m}_t) = \frac{1}{2} \sum_{i=1}^{K} ((\vec{m}_s)_i - (\vec{m}_t)_i)^2 \tag{8}$$

$$L_v(\vec{v}_s, \vec{v}_t) = \frac{1}{2} \sum_{i=1}^{K} ((\vec{v}_s)_i - (\vec{v}_t)_i)^2 \tag{9}$$

where i is index of each element in the vectors.

3.3 Dynamic Estimation

However, it is nontrivial to optimize the monolingual and cross-lingual objectives simultaneously in equation 7. The statistics defined in equation 5 and 6 can only be calculated when all word embeddings are given and fixed, while word embeddings have to be learned online by stochastic gradient descent algorithm. So it is impractical to use these statistics to guide the online learning of word embeddings. To deal with this chicken-egg problem, we propose to dynamically estimate the statistics. Initially, we have:

$$\vec{m}_s = 0 \tag{10}$$

$$\vec{v}_s = 0 \tag{11}$$

which will be iteratively updated by each incoming training instance (x_i,y_i):

$$\vec{m}_s = \frac{1}{scount + 1} (\vec{m}_s * scount + \vec{h}(x_i)) \tag{12}$$

$$\vec{v}_s \approx \frac{1}{scount + 1} (\vec{v}_s * scount + (\vec{h}(x_i) - \vec{m}_s)^2) \tag{13}$$

where $scount$ is the number of source training instances that have been used by the learning algorithm so far. The value of $scount$ is increased by one when a new training instance comes. To capture the latest trends and decay the effect the outdate data during training, we do not increase $scount$ anymore when it reaches 100 thousands.

On the target data, $\vec{m}_t$ and $\vec{v}_t$ are approximated in the same way.

3.4 Online Bilingual Training

Now we are ready to introduce the joint training procedure. In practice, we use multiple threads to train our model with source data and target data in parallel. On the source data, when a training instance (x_i, y_i) is accessed by the learning algorithm, we dynamically update the $\vec{m}_s$ and $\vec{v}_s$ with equation 12 and 13 as the output of the network. Then the golden references are the real time values of $\vec{m}_t$ and $\vec{v}_t$ which are being estimated in parallel on the target data. Based on loss function defined in equation 8 and 9, the gradient of distribution dissimilarities with respect to the hidden state is:

$$\frac{\partial(L_m + L_v)}{\partial \vec{h}(x_i)} = \frac{\lambda_m}{scount + 1}(\vec{m}_t - \vec{m}_s) + \frac{\lambda_v}{scount + 1}(\vec{v}_t - \vec{v}_s) \tag{14}$$

which will be added to the standard gradient of monolingual objective of the CBOW model:

$$\frac{\partial(L_m + L_v)}{\partial \vec{h}(x_i)} + \frac{\partial L(W, M, x, y)}{\partial \vec{h}(x_i)} \tag{15}$$

This gradient sum is utilized by the standard back propagation algorithm to make source data distribution similar to that of target data on one hand, and optimize the monolingual objective on the other hand.

In parallel, on the target data, the similar training procedure is being performed. Thus, the jointly learned source language word embeddings and target word embeddings will share similar distributions.

3.5 Bilingual Negative Sampling

The exact computation for probability shown in equation 4 for all words for every training instance is very expensive. Following the CBOW model, we adopted the negative sampling algorithm for high computational efficiency. As usual, for each source word y_s and its context x_s, we randomly sample a few words other than y_s from the *source* vocabulary. For example, given source word *sits* and its context {*the, cats, on, the, mat*}, we will sample a few words other than *sits*. Each selected word y_s^n is treated as a negative sample and the probability of predicting y_s^n given x_s is minimized.

Different from the CBOW model, we also randomly sample a few words from the *target* vocabulary and minimize the probability of predicting each selected target word y_t^n given x_s. Such bilingual negative sampling(BNS) procedure may introduce noises if the sampled target word y_t^n just happens to be the translation of the source word y_s. But, statistically such chance is quite small given the big vocabulary size of large scale text.

To further reduce such chance, we apply word frequency based diagonal beam (Nuhn et al., 2012) to constrain the BNS. Intuitively, the translation of a high frequency word should also be a frequent word, and vice-versa. Nuhn et al. (2012) use diagonal beam to select translation candidates, in this paper we apply it to filter out possible translations. We sort both source and target words by their frequency. Let $r(y_s)$ and $r(y_t)$ be the frequency rank of a source/target word. To avoid selecting y_t^n which is the translation of y_s, we require that the frequency rank of the sampled target word y_t^n should satisfy:

$$\left| r(y_t^n) - r(y_s)\frac{V_t}{V_s} \right| > BS \tag{16}$$

where BS is the beam size.

Similarly, we also apply the above BNS procedure when learning word embeddings for the target data in parallel.

4 Experiments

In this section we present experiments which evaluate the utility of the induced bilingual word embeddings. We implemented our model in C by building on the word2vec. The implementation launches a monolingual CBOW model by separate threads for each language. All threads access the shared embedding parameters and distribution means/variances. We evaluated the induced bilingual embeddings on the task of finding word-to-word translations from nonparallel corpora. This task is referred as decipherment which has drawn significant amounts of interest in the past few years (Nuhn et al., 2012;

	French	English
Training	29,608,749	27,355,418
Evaluation	60,474,279	54,478,614

Table 1: Size of data in tokens used in French to English decipherment experiment.

	5k	10k
MonoGiza without word embeedings	13.74	7.8
MonoGiza with word embeedings	17.98	10.56
Optimizing L	7.62	4.74
Optimizing L and L_m	22.24	17.05
Optimizing L, L_m and L_v	**23.54**	**17.82**

Table 2: French to English decipherment top-5 accuracy (%) of 5k and 10k most frequent word types.

Ravi, 2013; Dou et al., 2015). Decipherment views a foreign language as a cipher for English and finds a translation table that converts foreign texts into sensible English. It is a very challenging task since there is not any supervision.

4.1 Settings

We use MonoGiza[1] which implemented the state-of-the-art decipherment algorithms described in Dou and Knight (2012) and Dou et al. (2015) as baseline. In the preprocessing step, MonoGiza converts all words in data into integers and does not make any use of morphology similarity. For a fair comparison and to be general, we neither utilize that at all. All experiments are performed on plain raw text, we leave the use of syntactic relations for future work. The word embeddings used by MonoGiza are trained with word2vec. For all word embeddings in both word2vec and our model, the dimensionality is 50.

4.2 French to English Decipherment

Data The datasets in our English to French experiments are publicly-available Europarl data[2]. From the English-French Europarl parallel data, we select the first half of English sentences and the second half French sentences respectively as non-parallel corpora for decipherment experiments. To evaluate the decipherment, we use Giza++ (Och and Ney, 2003) to align the Europarl parallel data to build a dictionary. All texts are tokenized by scripts from www.statmt.org. Table 1 lists the sizes of monolingual and parallel data used in this experiment.

Decipherment In order to make all results comparable, results for all methods reported here were obtained using the same nonparallel corpora. λ_m and λ_v were set to 0.2 and 0.1 respectively. The number of negative samples from both source and target side for each word are 5. The beam size of BNS was set to 1000. We use default values for all other hyper parameters in word2vec and MonoGiza. Table 2 shows the experimental results. Bilingual word embeddings are induced by optimizing the objectives in equation 7. For each French word, we select its top-5 nearest neighbor words in English as translations based on the cosine similarity defined in the shared 50-dimensional space. We use the evaluation script included in the package of MonoGiza. Though the absolute accuracy is not very high, we believe it is encouraging given that there is not any supervised information. Only optimizing the monolingual training objective L can capture some similarities between languages, but the accuracy is pretty low. Simultaneously minimizing L and distribution mean dissimilarity L_m is effective. We achieved the best performance when three objectives L, L_m and L_v are optimized together. In such case, the distributions of source and target data share more similarities.

Table 3 shows a number of example translations from French to English. Though they are far from perfect, some translations are meaningful and are semantically related to the correct translation.

[1] http://www.isi.edu/natural-language/software/monogiza_release_v1.0.tar.gz
[2] http://www.statmt.org/europarl/

french word	English Translations	cosine similarity	Dictionary Entry
Du	the	-0.128120	the
	is	-0.131779	
	in	-0.136466	
	that	-0.137784	
	which	-0.139639	
sincèrement	Liberalisation	-0.007873	Frankly
	Essentially	-0.009341	
	Throughout	-0.016099	
	vodka	-0.023932	
	Frankly	-0.031336	
principaux	important	-0.018043	important
	good	-0.018260	
	able	-0.018508	
	much	-0.018643	
	come	-0.018987	

Table 3: Examples of translations of words from French to English. The five most likely translations are shown.

	Chinese	English
Training	315,800,768	403,215,310
Evaluation	41,888,921	49,822,055

Table 4: Size of data in tokens used in Chinese to English decipherment experiment.

4.3 Chinese to English Decipherment

We have demonstrated the effect of our model with experiments on French and English which are related languages. To further evaluate the ability our model, we experiment on data in Chinese and English which are substantially different.

Data We use large scale Chinese and English data released by LDC. The monolingual Chinese data is the Xinhua part of Chinese Gigaword. The monolingual English data is the LDC English Gigaword. Bilingual word embeddings are induced based the above non-parallel data. A golden dictionary is built by Giza++ based on parallel corpus LDC2002E18, LDC2003E07, LDC2003E14, Hansards portion of LDC2004T07, LDC2004T08 and LDC2005T06. All Chinese sentences are segmented by the Stanford Word Segmenter[3]. Table 4 lists the sizes of monolingual and parallel data used in this experiment. The settings are the same as the French-English case.

Decipherment For the calculation of accuracy, we discarded Chinese words if they are not covered by the gold dictionary. Table 5 shows the experimental results of Chinese to English decipherment. Though the two languages are substantially different, the results indicate that our model is still able to learn translation equivalences from the monolingual data by using only the learned bilingual word embeddings.

	5k	10k
MonoGiza without word embeedings	15.56	9.04
MonoGiza with word embeedings	17.5	10.57
Optimizing L, L_m and L_v	**24.76**	**18.45**

Table 5: Chinese to English decipherment top-5 accuracy (%) of 5k and 10k most frequent word types.

5 Conclusions and Future Work

We have proposed a novel model to learn bilingual word embeddings directly from monolingual raw text, without requiring any parallel data or dictionaries. A novel cross-lingual learning objective is proposed which directly matches the distributions of word embeddings in one language with that in the other language. We have demonstrated the utility of the learned word embeddings in the task of decipherment. Our model achieved encouraging performance on data from both related languages and substantially

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

different languages. In the future, we would apply our method to more real applications such as cross-lingual dependency parsing, cross-lingual document classification and machine translation. Our model is complementary to the existing methods that rely on supervision, so we are also interested in combining it with supervised models to achieve much better cross-lingual word representations.

Acknowledgments

We thank anonymous reviewers for their insightful comments. The work of HIT is funded by the projects of National Natural Science Foundation of China(No.91520204, No.71531013, No. 61572154) and the project of National High Technology Research and Development Program of China(No. 2015AA015405)

References

Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, and Mario Marchand. 2014. Domain-adversarial neural networks. *arXiv preprint arXiv:1412.4446*.

Shai Ben-David, John Blitzer, Koby Crammer, Fernando Pereira, et al. 2006. Analysis of representations for domain adaptation. In *Proceedings of the Conference on Advances in Neural Information Processing Systems(NIPS)*.

Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. 2010. A theory of learning from different domains. *Machine Learning*, 79(1-2):151–175.

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

John Blitzer, Ryan McDonald, and Fernando Pereira. 2006. Domain adaptation with structural correspondence learning. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, pages 120–128, Sydney, Australia, July. Association for Computational Linguistics.

A. P. Sarath Chandar, Stanislas Lauly, Hugo Larochelle, Mitesh M. Khapra, Balaraman Ravindran, Vikas C. Raykar, and Amrita Saha. 2014. An autoencoder approach to learning bilingual word representations. In *Proceedings of the Conference on Advances in Neural Information Processing Systems(NIPS)*, pages 1853–1861.

Ronan Collobert and Jason Weston. 2008. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of International Conference on Machine Learning*.

Hal Daume III. 2007. Frustratingly easy domain adaptation. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 256–263, Prague, Czech Republic, June. Association for Computational Linguistics.

Qing Dou and Kevin Knight. 2012. Large scale decipherment for out-of-domain machine translation. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 266–275, Jeju Island, Korea, July. Association for Computational Linguistics.

Qing Dou, Ashish Vaswani, Kevin Knight, and Chris Dyer. 2015. Unifying bayesian inference and vector space models for improved decipherment. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 836–845, Beijing, China, July. Association for Computational Linguistics.

Manaal Faruqui and Chris Dyer. 2014. Improving vector space word representations using multilingual correlation. In *Advances in neural information processing systems the European Chapter of the Association for Computational Linguistics*, pages 462–471, Gothenburg, Sweden, April. Association for Computational Linguistics.

Yaroslav Ganin and Victor Lempitsky. 2015. Unsupervised domain adaptation by backpropagation. In *Proceedings of The 32nd International Conference on Machine Learning*, Lille, France.

Stephan Gouws, Yoshua Bengio, and Greg Corado. 2015. Fast bilingual distributed representations without word alignments. In *Proceedings of The 32nd International Conference on Machine Learning*, Lille, France.

Jiang Guo, Wanxiang Che, David Yarowsky, Haifeng Wang, and Ting Liu. 2015. Cross-lingual dependency parsing based on distributed representations. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1234–1244, Beijing, China, July. Association for Computational Linguistics.

Karl Moritz Hermann and Phil Blunsom. 2014. Multilingual models for compositional distributed semantics. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 58–68, Baltimore, Maryland, June. Association for Computational Linguistics.

Alexandre Klementiev, Ivan Titov, and Binod Bhattarai. 2012. Inducing crosslingual distributed representations of words. In *Proceedings of COLING 2012*, pages 1459–1474, Mumbai, India, December. The COLING 2012 Organizing Committee.

Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Bilingual word representations with monolingual quality in mind. In *Proceedings of the Workshop on Vector Space Modeling for NLP*.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. In *Proceedings of 2013 Workshop at ICLR*.

Tomas Mikolov, Quoc V. Le, and Ilya Sutskever. 2013b. Exploiting similarities among languages for machine translation. *arXiv:1309.4168*, abs/1309.4168.

Malte Nuhn, Arne Mauser, and Hermann Ney. 2012. Deciphering foreign language by combining language models and context vectors. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 156–164, Jeju Island, Korea, July. Association for Computational Linguistics.

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

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar, October. Association for Computational Linguistics.

Sujith Ravi. 2013. Scalable decipherment for machine translation via hash sampling. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 362–371, Sofia, Bulgaria, August. Association for Computational Linguistics.

Tianze Shi, Zhiyuan Liu, Yang Liu, and Maosong Sun. 2015. Learning cross-lingual word embeddings via matrix co-factorization. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 567–572, Beijing, China, July. Association for Computational Linguistics.

Shyam Upadhyay, Manaal Faruqui, Chris Dyer, and Dan Roth. 2016. Cross-lingual models of word embeddings: An empirical comparison. In *Proceedings of ACL*.

Ivan Vulić and Marie-Francine Moens. 2015. Bilingual word embeddings from non-parallel document-aligned data applied to bilingual lexicon induction. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 719–725, Beijing, China, July. Association for Computational Linguistics.

Hyejin Youn, Logan Sutton, Eric Smith, Cristopher Moore, Jon F. Wilkins, Ian Maddieson, William Croft, and Tanmoy Bhattacharya. 2016. On the universal structure of human lexical semantics. *In Proceedings of the National Academy of Sciences*.

Jiajun Zhang, Shujie Liu, Mu Li, Ming Zhou, and Chengqing Zong. 2014. Bilingually-constrained phrase embeddings for machine translation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 111–121, Baltimore, Maryland, June. Association for Computational Linguistics.

Meng Zhang, Yang Liu, Huan-Bo Luan, Maosong Sun, Tatsuya Izuha, and Jie Hao. 2016. Building earth mover's distance on bilingual word embeddings for machine translation. In Dale Schuurmans and Michael P. Wellman, editors, *AAAI*, pages 2870–2876. AAAI Press.

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, Seattle, Washington, USA, October. Association for Computational Linguistics.