Improving Vector Space Word Representations Via Kernel Canonical Correlation Analysis

XUEFENG BAI, HAILONG CAO, and TIEJUN ZHAO, Harbin Institute of Technology, China

Cross-lingual word embeddings are representations for vocabularies of two or more languages in one common continuous vector space and are widely used in various natural language processing tasks. A state-of-the-art way to generate cross-lingual word embeddings is to learn a linear mapping, with an assumption that the vector representations of similar words in different languages are related by a linear relationship. However, this assumption does not always hold true, especially for substantially different languages. We therefore propose to use kernel canonical correlation analysis to capture a non-linear relationship between word embeddings of two languages. By extensively evaluating the learned word embeddings on three tasks (word similarity, cross-lingual dictionary induction, and cross-lingual document classification) across five language pairs, we demonstrate that our proposed approach achieves essentially better performances than previous linear methods on all of the three tasks, especially for language pairs with substantial typological difference.

CCS Concepts: • **Computing methodologies** → **Machine translation**;

Additional Key Words and Phrases: Cross-lingual word representation, kernel canonical correlation analysis (KCCA), word embedding evaluation

ACM Reference format:
XueFeng Bai, HaiLong Cao, and TieJun Zhao. 2018. Improving Vector Space Word Representations Via Kernel Canonical Correlation Analysis. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.* 17, 4, Article 29 (July 2018), 16 pages.
https://doi.org/10.1145/3197566

1 INTRODUCTION

Monolingual word embeddings have made great achievements in many natural language processing (NLP) tasks, including sentiment analysis [28] and dependency parsing [5, 13]. As a natural extension of monolingual word embeddings, cross-lingual word embeddings do not merely improve the performance on monolingual NLP tasks, but facilitate some cross-lingual tasks as well, such as machine translation [23, 32, 37], cross-lingual document classification [17, 27, 31, 35], and cross-lingual dependency parsing [11, 21].

Several models for inducing cross-lingual word embeddings have been proposed recently. Most models require large parallel corpora [11, 19]. However, there have been several proposals to relax this requirement, given that the large parallel corpora are scarce in most language pairs. An

The work of this article was funded by the projects of National Natural Science Foundation of China under Grants No. 61572154 and No. 91520204.
Authors' addresses: X. Bai, H. Cao (corresponding author), and T. Zhao, No.92, West Dazhi Street, Nan Gang District, Harbin Heilongjiang, 150001, China; emails: bxf_hit@163.com, caohailong2008@gmail.com, tjzhao@hit.edu.cn.

https://doi.org/10.1145/3197566

alternative relaxation is to utilize document-aligned or label-aligned comparable corpora [12, 24, 34], but such corpora are not always sufficient for some language pairs.

The approach we adopt is to train monolingual word representations independently, and then learn a transformation to map the embeddings from one space into the other with a bilingual dictionary as constraint. In their pioneering work, Mikolov et al. [23] observe that a linear transformation can be established to connect word embeddings trained separately on monolingual corpora. Faruqui and Dyer 2014 [7] extend this idea by using canonical correlation analysis (CCA) to map both languages to a shared vector space and produce semantic vectors with state-of-the-art performance. The idea of such a model is to assume that there is often a strong linear relationship between word embeddings of two languages and then to capture these linear factors via various methods, such as Translation Matrix or CCA. However, they are still limited to previous linear assumption, which is not robust enough to represent the relationship. Hence, we propose a more powerful model to capture a non-linear relationship using kernel canonical correlation analysis (KCCA) [1].

The overall structure of the article is as follows. We firstly review some related works on cross-lingual representations in recent years (Section 2). In Section 3, we present illustrations of our proposed model, together with an explanation of the training process. Section 4 describes the evaluation tasks—our intrinsic evaluation assesses the quality of the vectors on monolingual (word similarity for English) and cross-lingual (cross-lingual dictionary induction) tasks, while our extrinsic evaluation (cross-lingual document classification) assesses the ability of cross-lingually trained vectors to facilitate model transfer across languages. In Section 5—experimental methodology—the corpora used, the setting of our model, and the results of the evaluation tasks are presented. In Section 6, a systematic analysis of the resulting word embeddings, errors, and the proposed model are given, with several case studies from the experimental data. We conclude and offer possible directions for future research in Section 7.

Our contributions are the following:

- We propose a KCCA-based method to capture a non-linear relationship between two languages.
- We extensively evaluate embeddings produced by our model on both extrinsic and intrinsic tasks across five language pairs, and we show that our method produces essentially higher-quality vectors than all previous linear model.
- We qualitatively analyze the resulting word embeddings, and we discover that there are a lot of nonlinear relationships among substantially different languages, while little among closely related languages. We believe this discovery will be helpful for further research.

2 RELATED WORKS

In this section, we briefly introduce some successful techniques for cross-lingual representation learning, which could be roughly classified into two categories: (I) Joint Learning, which trains models on parallel (and optionally monolingual) data, and jointly optimizes a combination of monolingual and cross-lingual losses. (II) Learning Bilingual Mappings, which initially trains monolingual word embeddings on large monolingual corpora, then learns a linear mapping between monolingual representations of different languages to enable them to map unknown words from the source language to the target language.

2.1 Joint Learning

Klementiev et al. [17] proposed a jointly optimized model for learning cross-lingual representations in 2012. They trained a neural language model for each language and jointly optimized the

monolingual maximum likelihood objective of each language model with a word-alignment-based machine translation (MT) regularization term as the cross-lingual objective. Luong et al. [19], in turn, extend skip-gram to the cross-lingual setting and use the skip-gram objective as both monolingual and cross-lingual objective. Instead of just predicting the surrounding words in the source language, they use the words in the source language to additionally predict their aligned words in the target language as well. Gouws et al. [11] propose a Bilingual Bag-of-Words without Word Alignments (BilBOWA) that leverages additional monolingual data. They use the skip-gram objective as a monolingual objective and a novel sampled l_2 loss as a cross-lingual regularizer. Shi et al. [27] propose a joint matrix factorisation model to learn cross-lingual representations and also use additional monolingual data.

2.2 Learning Bilingual Mappings

In their pioneering work, Mikolov et al. [23] proposed a dictionary-based approach that learns a translation matrix that minimizes the sum of squared Euclidean distances for the given dictionary entries. The same optimization objective is used by Zhang et al. [38], with constraint that the transformation matrix should be orthogonal. Motivated by a hypothetical inconsistency in Reference [23], Xing et al. [36] incorporated length normalization in the training of word embeddings and maximized the cosine similarity, preserving the length normalization after mapping. However, Reference [36] is equivalent to that used by Reference [23] with orthogonality constraint and unit vectors (Artetxe et al. [2]). Artetxe et al. [2] develop a framework to learn bilingual word embedding mappings, generalizing previous work and providing an efficient exact method to learn the optimal transformation.

Another efficient approach is proposed by Faruqui and Dyer [7], who use CCA to learn two linear transformations for both sides that maximize the Pearson correlation coefficient. By performing CCA, they get one of the best semantic vectors among all bilingual mapping methods.

In contrast to joint learning models, these models are not only computationally-efficient and easy to scale to large datasets but also reduce (eliminate) the requirement of parallel corpus.

3 OUR APPROACH

3.1 Canonical Correlation Analysis

A popular method for multi-representation learning is canonical correlation analysis [16], which is a method of correlating a linear relationship between two multidimensional random variables. It finds two projection vectors, one for each variable, which are optimal with respect to correlations. The dimension of these new projected vectors is equal to or less than the smaller dimension of the two original variables.

Given a pair of multi-variates, $X \in \mathbb{R}^{n_x}$, $Y \in \mathbb{R}^{n_y}$, CCA[1] finds a pair of projection vectors, $a \in \mathbb{R}^{n_x}, b \in \mathbb{R}^{n_y}$, let $u = a^T X, v = b^T Y$, such that the correlation coefficient $\rho = corr(u, v)$ is maximized (Figure 1).

Let $\Sigma_{xx} = cov(X, X)$, $\Sigma_{yy} = cov(Y, Y)$ and $\Sigma_{xy} = cov(X, Y)$, the objective to maximize is

$$\rho(u, v) = \frac{cov(u, v)}{\sqrt{Var(u)}\sqrt{Var(v)}} = \frac{a^T \Sigma_{xy} b}{\sqrt{a^T \Sigma_{xx} a}\sqrt{b^T \Sigma_{yy} b}}. \quad (1)$$

Observing that Equation (1) is not affected by the rescaling of a and b either together or independently, the CCA optimization problem is equivalent to maximizing the numerator subject to

[1]For more details of CCA, please refer to Relations Between Two Sets of Variates [16].

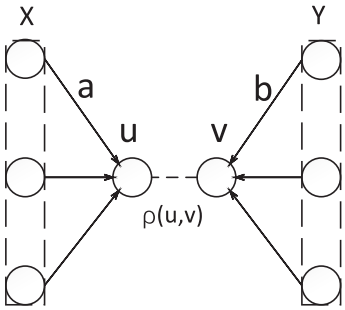

Fig. 1. CCA seeks a pair of linear transformations, $a \in \mathbb{R}^{n_x}$, $b \in \mathbb{R}^{n_y}$, to maximize the correlation coefficient $\rho = corr(u, v)$, in which $u = a^T X, v = B^T Y$.

$$Var[u] = Var[v] = 1. \tag{2}$$

The corresponding Lagrangian to this optimization problem is

$$L = a^T \Sigma_{xy} b - \frac{\lambda}{2}(a^T \Sigma_{xx} a - 1) - \frac{\theta}{2}(b^T \Sigma_{yy} b - 1). \tag{3}$$

By solving Equation (3), a and b can be found by an eigenvector corresponding to the maximal eigenvalues of a generalized eigenvalue problem:

$$\Sigma_{xx}^{-1} \Sigma_{xy} \Sigma_{yy}^{-1} \Sigma_{yx} a = \lambda^2 a. \tag{4}$$

If more dimensions are needed, then we can take eigenvectors corresponding to other maximal eigenvalues.

3.2 Kernel Canonical Correlation Analysis

In this subsection, we introduce KCCA briefly.[2] The kernel version of CCA offers an alternate solution to learn non-linear factors by first projecting the data X, Y onto Hilbert spaces H_x, H_y via kernel function Φ_x, Φ_y:

$$\Phi : X = (X_1 X_2 \cdots X_n) \rightarrow \Phi(X) = (\phi_1(X), \phi_2(X) \cdots \phi_N(X)), \text{n<N},$$

which could be a non-linear transformation, and then performing CCA in the new feature space.

First, X and Y are transformed into Hilbert space, $\Phi_x(X) \in H_x, \Phi_y(Y) \in H_y$. Similar to CCA, KCCA (Figure 2) seeks two projection vectors $a \in H_x, b \in H_y$. By taking inner products with projection vectors $a \in H_x, b \in H_y$, we find two features:

$$u = a^T \Phi_x(X), \tag{5}$$

$$v = b^T \Phi_y(Y), \tag{6}$$

which maximize the correlation coefficients $\rho = corr(u, v)$.

To solve this problem, we create a matrix P whose rows are the vectors $\Phi_x(X_i), i = 1, \ldots, m$, and similarly a matrix Q with rows $\Phi_y(Y_i), i = 1, \ldots, m$.

The covariance matrix of P and Q can be described as[3]

$$C_{pp} = P^T P, \tag{7}$$

[2]For more details, please refer to Statistical Consistency of Kernel Canonical Correlation [10].
[3]P and Q are centered matrix.

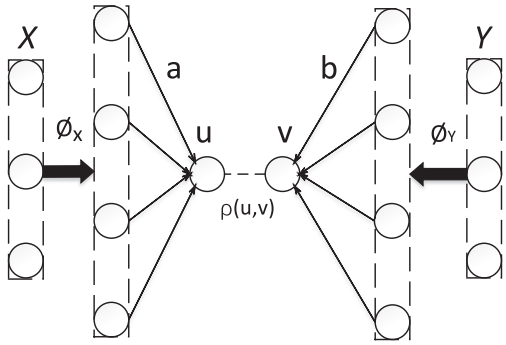

Fig. 2. KCCA seeks two projection vectors $a \in H_x$, $b \in H_y$ such that the inner products $u = a^T\Phi_x(X), v = b^T\Phi_y(Y)$ maximize the correlation coefficient $\rho = corr(u, v)$.

$$C_{pq} = P^T Q. \tag{8}$$

Further more, the projection vector a and b can be expressed as a linear combination of the training examples:

$$a = P^T\alpha, \tag{9}$$

$$b = Q^T\beta, \tag{10}$$

where α, β are m-dimensional vectors as the parameters of kernel CCA. So far, the objective to maximize can be described as

$$\rho(u, v) = \frac{cov(u, v)}{\sqrt{Var(u)}\sqrt{Var(v)}} = \frac{\alpha^T PP^T QQ^T\beta}{\sqrt{\alpha^T PP^T PP^T\alpha}\sqrt{\beta^T QQ^T QQ^T\beta}}. \tag{11}$$

Given the kernel functions Φ_x and Φ_y, let K_p and K_q be the kernel matrices corresponding to two representations of the data, where $K_p = PP^T, K_q = QQ^T$. Substituting it into Equation (11), we get

$$\rho(u, v) = \frac{\alpha^T K_p K_q\beta}{\sqrt{\alpha^T {K_p}^2\alpha}\sqrt{\beta^T {K_q}^2\beta}}. \tag{12}$$

Observing that Equation (12) is not affected by the rescaling of α and β either together or independently, the kernel CCA optimization problem is equivalent to maximizing the numerator subject to

$$\alpha^T {K_p}^2\alpha = \beta^T {K_q}^2\beta = 1. \tag{13}$$

The corresponding Lagrangian with regularization for KCCA is

$$L = \alpha^T K_p K_q\beta - \frac{\lambda}{2}(\alpha^T K_p^2\alpha - 1) - \frac{\theta}{2}(\beta^T K_q^2\beta - 1) + \frac{\eta}{2}(\| \alpha \|^2 + \| \beta \|^2). \tag{14}$$

By solving Equation (14), α and β can be found by an eigenvector corresponding to the maximal eigenvalues of a generalized eigenvalue problem[4]:

$$(K_p + \eta I)^{-1} K_q (K_q + \eta I)^{-1} K_p\alpha = \lambda^2\alpha. \tag{15}$$

Thus, we have found the first pair of canonical variables. If more dimensions are needed, then we can take eigenvectors corresponding to other maximal eigenvalues.

[4] η is a hyper-parameter; I is identity matrix.

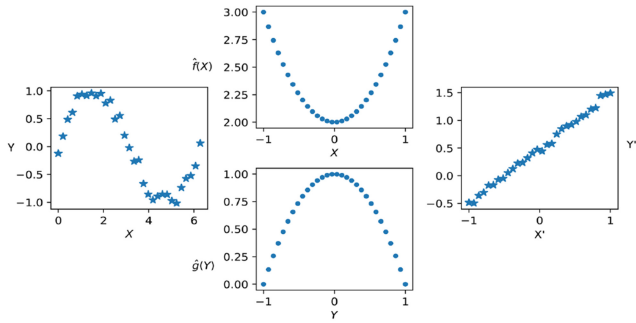

Fig. 3. An example of kernel CCA. A Gaussian RBF kernel $k(x, y) = exp(-\frac{1}{2\sigma^2}(x-y)^2)$ is used for both X and Y. Left: distribution of original data. Center: non-linear transformation functions $\hat{f}(X), \hat{g}(Y)$. Right: distribution of transformed data.

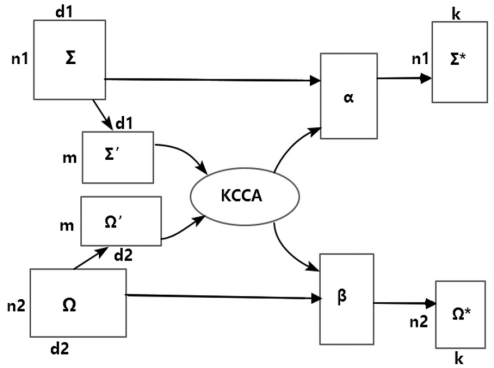

Fig. 4. Generating cross-lingual word vector using KCCA.

3.3 Generating Multi-lingual Embeddings Using KCCA

Figure 3 illustrates how KCCA captures non-linear relationship. The nonlinear mappings clarify the strong dependency between X and Y. KCCA captures such relationship by learning two non-linear transformation functions $\hat{f}(X), \hat{g}(Y)$. After transformed by $\hat{f}(X), \hat{g}(Y)$, the relationship between X', Y' becomes clear and easy to capture. Note that the dependency of the original data can not be captured by CCA, because they have no linear correlation.

Now, we describe how to apply KCCA to our task. We first use KCCA to learn the relationship between two monolingual vocabularies from a bilingual lexicon and output two transformation matrices. By using these two transformation matrices, we then project the original word embeddings onto a new shared vector space. The schema of performing KCCA on the monolingual word representations of two languages (BiKCCA) is shown in Figure 4.

Let $\Sigma \in \mathbb{R}^{n_1 \times d_1}$ and $\Omega \in \mathbb{R}^{n_2 \times d_2}$ be word embeddings of two languages' vocabularies, where n_1, n_2 represent the size of vocabularies, and d_1, d_2 denote the dimensionality of word vectors. The training matrices Σ', Ω' are constructed as $\Sigma' \subset \Sigma$, $\Omega' \subset \Omega$, where Ω'_i is translated from Σ'_i.

We use KCCA to maximize ρ for the given set Σ' and Ω' and output two parameters α, β:

$$\alpha, \beta = KCCA(\Sigma', \Omega') = \arg\max_{\alpha,\beta} \rho(a^T \Phi_x(\Sigma'), b^T \Phi_y(\Omega')). \tag{16}$$

a, b in Equation (16) can be denoted by α, β as in Equations (9) and (10).

Using two such vectors, α, β, we can project the entire vocabulary of the two languages Σ and Ω onto a shared vector space $\mathbb{R}^{D_k}$ via Equations (5) and (6). Substituting into Equations (9) and (10), this problem could be summarized as

$$\alpha_k, \beta_k = KCCA(\Sigma', \Omega'), \tag{17}$$

$$\Sigma_k^* = P_\Sigma P_{\Sigma'}^T \alpha_k, \Omega_k^* = Q_\Omega Q_{\Omega'}^T \beta_k, \tag{18}$$

where P_Σ is a matrix whose rows are vectors $\Phi_x(X_i)$[5], $X_i \in \Sigma$, Q_Ω is a matrix whose rows are vectors $\Phi_y(Y_i)$, $Y_i \in \Omega$, and $P_{\Sigma'}$, $Q_{\Omega'}$ are similar to P_Σ, Q_Ω. We perform experiments by taking projections of the top k correlated dimensions, α_k, β_k are the corresponding parameter matrix. $\Sigma_k^* \in \mathbb{R}^{n_1 \times k}$, $\Omega_k^* \in \mathbb{R}^{n_2 \times k}$ are new word vector matrices for two languages.

4 WORD REPRESENTATION EVALUATION

Upadhyay et al. [31] perform a systematic comparison of four popular approaches of inducing cross-lingual embeddings. We follow the setup of Upadhyay to evaluate our embeddings. The following three tasks are performed to assess the quality of the induced cross-lingual word embeddings:

- Monolingual word similarity for English
- Cross-lingual dictionary induction
- Cross-lingual document classification

The first two tasks intrinsically show how much benefit would be gained from cross-lingual training. The last task measures the ability of cross-lingually trained vectors to extrinsically facilitate model transfer across languages.[6]

4.1 Monolingual Evaluation

We first evaluate whether the proposed model can improve the quality of English embeddings. The task of monolingual evaluation assesses how well the trained word embeddings capture human intuitions about semantic relatedness via word similarity datasets.

We use the SimLex dataset for English [15], which contains 666 noun pairs, 222 verb pairs, and 111 adjective pairs. SimLex is claimed to capture word similarity merely instead of WordSim-353 [8], which captures both word similarity and relatedness. To evaluate embeddings, we compute cosine similarity between two vectors in each pair, order the pairs by similarity, and compute Spearman correlation (ρ) [25] between the model's ranking and human ranking.

4.2 Cross-lingual Dictionary Induction (CLDI)

In this task, we assess whether our non-linear model improves the quality of the vector on cross-lingual dictionary induction task. The task of cross-lingual dictionary induction [11, 23, 31, 33, 38] judges the ability of cross-lingual embeddings to detect word pairs that are semantically similar across languages.

We follow the setup of Reference [31] and derive our gold dictionaries via the Open Multilingual WordNet data released by Bond and Foster [3]. Using the approach described in Reference [31], we generated gold dictionaries for each language pairs. Top-1 accuracy is reported on this task. For each pair (e, f) in the gold dictionary, we check if f belongs to the list of top-1 neighbors of e, according to the induced cross-lingual word vectors.

[5] Φ_x, Φ_y are kernel functions defined in Section 3.1.

[6] To ensure a fair comparison, all models are trained with embeddings of size 200, and product embeddings of size 100.

4.3 Cross-lingual Document Classification (CLDC)

The cross-lingual document classification task assesses whether the learned cross-lingual representations are semantically coherent across multiple languages.

We follow the cross-lingual document classification (CLDC) setup of Reference [31] but choose another popular corpus, WIT TED [4], which has more topics available than RCV2 [18] for our evaluation. Fifteen topics are chosen for the classification task in our experiment. Document representations are computed by taking the tf-idf-weighted average of vectors of the words present in it.[7] A multi-class classifier is then trained on the labeled training data in the source language via an averaged perceptron [9] for 10 iterations, using the document vectors of language l_1 as features. For each language pair (l_1, l_2), we train a document classifier using the document embeddings derived from word embeddings in language l_1, and we test this model on document embeddings from l_2.

We refer to the result of MT system trained by Hermann and Blunsom [14] on TED corpus as a baseline. It uses the CDEC decoder [6] with default settings to translate documents.

5 EXPERIMENTS

In this section, we perform the tasks described in (Section 4) to evaluate the utility of the induced bilingual word embeddings and present the results.

5.1 Baseline

We introduce the pioneer bilingual mapping model translation matrix (TM) [23], which uses seed lexicon as our baseline. In their work, a linear transformation between source and target language is learned to project source language embeddings to target language. We choose a publicly available implementation here.[8] We also report the CCA-based method BiCCA [7].[9] Another state-of-the-art work [2], which claims to yield the best results, is reported in addition. They generalize previous work and provide an efficient exact method to learn the optimal linear transformation. We use their original implementation on github.[10]

5.2 Data

We train cross-lingual embeddings for five language pairs: English-German (en-de), English-French (en-fr), English-Arabic (en-ar), English-Russian (en-ru), and English-Chinese (en-zh). For English, French, and German, monolingual corpora are obtained from Europarl,[11] and for Arabic, Russian, and Chinese, text from Leipzig[12] is used.

To generate a seed lexicon, parallel corpora from Europarl are used. First, word pair (a, b), $a \in l_1$, $b \in l_2$ is selected such that a is aligned to b the most number of times in parallel corpus and vice versa. Then, our seed lexicon is constructed from the most common pairs. For Arabic, Russian, and Chinese, where parallel corpus is unavailable in Europarl, dictionaries are induced by translating the 20k most common words in the English monolingual corpus with Google Translation. To balance the speed and performance, we use a bilingual lexicon about 7k words for each language pair to train the model in practice. Details of training and evaluation data are presented in Table 1.

[7] tf-idf [26] was computed by using all documents for each language in TED.
[8] http://clic.cimec.unitn.it/~georgiana.dinu/down.
[9] We reimplement Faruqui and Dyer's work from 2014 [7] and pre-process the training embeddings with length normalization and mean centering (proposed by References [2, 36], which greatly improves the performance) to get a stronger baseline.
[10] https://github.com/artetxem/vecmap.
[11] http://www.statmt.org/europarl/.
[12] http://wortschatz.uni-leipzig.de/en.

Table 1. Experiment Data

L_1	L_2	Seeds	CLDI (#golden words)	CLDC (#articles)
en	fr	6.9k	1.5k	1.5k
	de	7.1k	1.4k	1.0k
	ar	6.8k	1.5k	1.1k
	ru	6.9k	1.4k	1.3k
	zh	7.0k	1.6k	1.3k

The data statistics of different language pairs used for training and evaluating cross-lingual word vectors.

Table 2. Intrinsic Evaluation of English Word Vectors

L_1	L_2	Mono	TM	Artetxe et al.	BiCCA	BiKCCA (ours)
en	fr	0.336	0.335	0.339	0.353	**0.356**
	de	0.336	0.328	0.339	0.356	**0.362**
	ar	0.336	0.335	0.339	0.379	**0.417**
	ru	0.336	0.338	0.339	0.371	**0.387**
	zh	0.336	0.337	0.339	0.382	**0.395**
avg.		0.336	0.335	0.339	0.368	**0.383**

Word similarity score measured in Spearman's correlation ratio for English on SimLex-999, with higher being better. Scores that are significantly better are underlined. Bold indicates best result in each language pair. The similar trend was also observed when computing QVEC [30].

5.3 Settings

First, original monolingual vectors are trained via the skip-gram model[13] with negative sampling [22] with window size 5 (tuned over 5, 10, 20), and the dimension of embeddings is set to 200.[14] Bilingual dictionaries are generated as described in (Section 5.2). We use k = 0.5 as the scaling factor of canonical components (tuned over 0.2, 0.3, 0.5, 1.0) as also done by Faruqui and Dyer [7]. After performing this, we get embeddings of size 100 for evaluation. For KCCA, we use a radial basis function (RBF) kernel[15] for both views: $k_1(x_i, y_i) = exp(-\gamma(\| x_i - y_i \|^2)$ and similarly for k_2. The parameters γ_1, γ_2 are tuned over the range $[10^{-1}, 10^{-5}]$. Regularization parameter η in Equation (13) is tuned over the range $[10, 10^{-6}]$. For each evaluation task, we perform fivefold cross-validation and choose the hyperparameters with the best performance.

5.4 Results

Monolingual Word Similarity. Table 2 shows our main results of the monolingual word similarity task (Section 4.1). We compare the performance of monolingual word embeddings (Mono), translation matrix (TM) word embeddings, CCA-based bilingual word embeddings (BiCCA), Artetxe et al. [2], and KCCA-based bilingual word embeddings (BiKCCA). Monolingual model is used as a baseline. We declare significant improvement if $p < 0.15$ according to Steiger's method [29] for calculating the statistical significant differences between two dependent correlation coefficients.

[13] http://code.google.com/p/word2vec.
[14] All training embeddings are preprocessed with normalization.
[15] We also tried poly kernel, did not observe any superiority in performance.

Table 3. Cross-lingual Dictionary Induction

L_1	L_2	TM	Artetxe et al.	BiCCA	BiKCCA (ours)
en	fr	48.9	50.6	52.4	**53.3**
	de	48.4	51.1	53.2	**54.6**
	ar	28.4	31.2	32.7	**49.8**
	ru	26.8	35.2	37.3	**49.9**
	zh	34.3	35.5	36.3	**46.1**
avg.		37.3	40.7	42.4	**50.8**

Cross-lingual dictionary induction results (top-k accuracy, k = 1). Bold indicates best result. The similar trend was also observed across models when computing MRR (mean reciprocal rank).

It can be seen from Table 2 that two CCA-based models show better performance than TM and Artetxe et al.'s work. The reason is that CCA can incorporates multilingual evidence into vectors generated monolingually (Faruqui and Dyer [7]), which cannot be done by translation matrix. Furthermore, BiKCCA results show consistent improvements across most language pairs BiCCA (all the BiKCCA results in bold are better than BiCCA). This indicates that the non-linear relationship is useful to improve the quality of word embeddings on word similarity task. Moreover, we note that across-language pairs, which are substantially different, such as English-Russian, English-Arabic, and English-Chinese, BiKCCA achieves greater improvement than other language pairs.

Cross-lingual Dictionary Induction. In Table 3, we report the Top-1 accuracy of cross-lingual dictionary induction task (Section 4.2).

From Table 3, we note that by capturing non-linear relationship between two source languages, BiKCCA performs essentially better than other linear systems listed across all language pairs. This matches our assumption that the non-linear model is more suitable for capturing factors among languages. Clearly, the effect of BiKCCA on substantially different language pairs like en-ar, en-ru, and en-zh, is much more obvious.

Cross-lingual Document Classification. Table 4 shows performance of different models on cross-lingual document classification task (Section 4.3) across different language pairs. We report average F1-score of 15 topics, which can be interpreted as a weighted average of the precision and recall. The fourth and tenth row of Table 4 show the result of MT system reported by Hermann and Blunsom [14]. We list it here for reference but note that it is not comparable to our results, since the classifier of their system has access to significantly more information (all words in the document) as opposed to our model (one embedding per document), and we do not expect to defeat this system.

It can be easily seen that BiCCA generally have better performance than TM and are comparable to Artetxe et al. [2]. When comparing the results of the linear model (TM, Artetxe et al. [2], BiCCA) and non-linear model (BiKCCA), the benefit of this additional non-linear relationship becomes clear (all the BiKCCA results underlined are better than other linear system). This suggests that for transferring semantic knowledge across languages via embeddings, non-linear model proves superior to linear model.

6 ANALYSIS

Result Analysis. Table 5 compares the total correlation on development sets across different language pairs obtained for the 50 most correlated dimensions with linear CCA and KCCA. As it

Table 4. Cross-lingual Document Classification Results

Setting	Languages				
	French	German	Arabic	Russian	Chinese
En-L_2					
MT Baseline	**0.526**	**0.465**	**0.429**	**0.432**	—
TM	0.377	0.315	0.242	0.147	0.195
Artetxe et al.	0.439	0.400	0.333	0.280	0.206
BiCCA	0.446	0.399	0.344	0.276	0.204
BiKCCA (ours)	0.450	0.410	0.345	0.285	0.223
L_2-En					
MT Baseline	0.358	**0.469**	**0.448**	**0.404**	—
TM	0.445	0.405	0.364	0.201	0.339
Artetxe et al.	0.450	0.442	0.344	0.313	0.370
BiCCA	0.452	0.387	0.373	0.329	0.374
BiKCCA (ours)	**0.470**	0.435	0.387	0.346	0.375

F1-scores for the TED document classification task for individual languages. Results are for two directions (training on English, evaluating on L2 and vice versa). Bold indicates best result, underline indicates best result between the bilingual vectors. We refer to the MT Baseline reported by Hermann and Blunsom [14].

Table 5. Correlation Captured in the 50 Most Correlated Dimensions

Method	En-Fr	En-De	En-Ar	En-Ru	En-Zh
BiCCA	41.1	39.55	29.4	31.5	30.9
BiKCCA	**41.9**	**40.33**	**40.3**	**38.8**	**39.8**

can be seen in Table 5, BiKCCA shows an overall improvement over BiCCA, especially for languages pairs like En-Ar, En-Ru, and En-Zh. When the average Pearson correlation coefficient is calculated, one may be surprised to find that the linear correlation coefficient for En-Ar, En-Ru, and En-Zh is very small (0.588, 0.630, 0.619, respectively), which goes against the previous strong linear assumption. After a non-linear transformation by KCCA, the coefficient rises to a normal level (0.805, 0.775, 0.795 respectively). This suggests that non-linear assumption is more suitable to describe the relationship between two languages.

To further understand how BiKCCA gets better performance in our evaluation tasks, we analyze the distribution of the cross-lingual word embeddings in shared vector space. Figure 5 shows the t-SNE [20] visualization of some high-frequency word pairs in the English-Chinese corpus. The original monolingual word vectors as well as bilingual word vectors generated by BiCCA (linear transformation) and BiKCCA (non-linear transformation) are presented, respectively. For each word pair, our goal is to learn transformations for each view to make two words across languages have similar representations, which can also be described as that two words are close in vector space. By comparing regions (b) and (d), we observe that: (i) for word pairs as (heaven, 天堂), (merchant, 商人), BiKCCA catches the corresponding words in Chinese while BiCCA does not (The distance between these words is too long in BiCCA); (ii) for word pairs like (notice, 注 意), (consecutive, 连续), which are caught both by BiCCA and BiKCCA, the distance between two translationally equivalent words in BiKCCA's vector space is shorter than BiCCA, which is a good property in many downstream applications. These two observations explain how BiKCCA gets

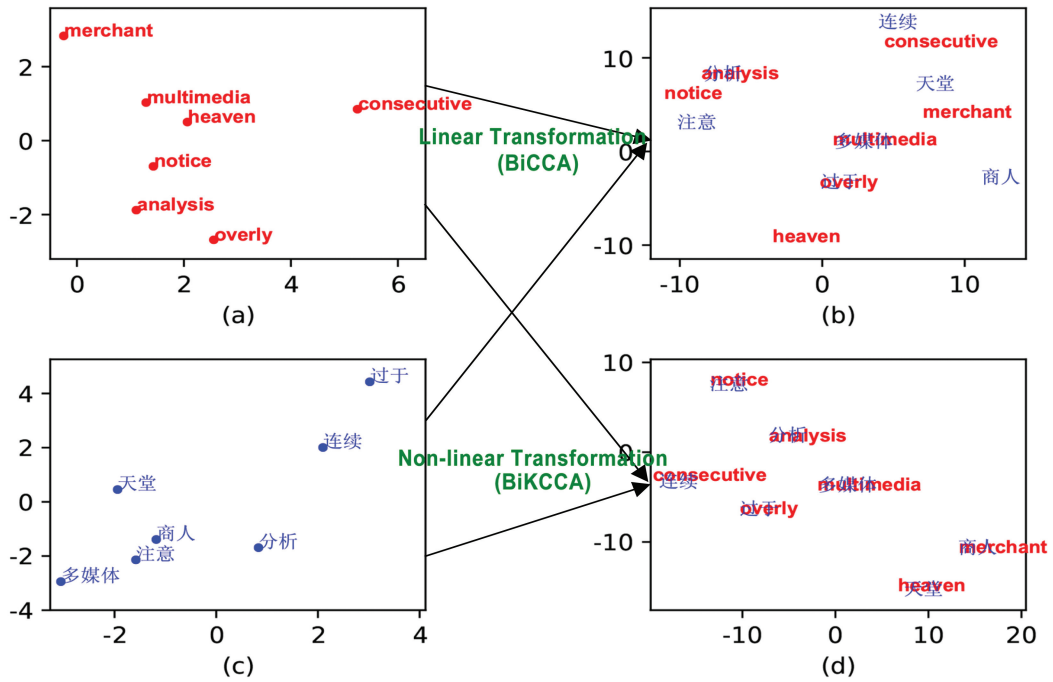

Fig. 5. t-SNE visualization of some frequent words in English-Chinese corpus. English and Chinese words are shown in red and blue, respectively. Regions (a) and (b) show words in origin English/Chinese vector space, (c) words in BiCCA shared vector space, (d) words in BiKCCA shared vector space.

Table 6. Cosine Similarity Between the Two Vectors in Each Word Pair

Word pair	Linear Trans (BiCCA)	Non-linear Trans (BiKCCA)
(analysis, 分析)	0.706	0.883
(multimedia, 多媒体)	0.863	0.935
(consecutive, 连续)	0.685	0.821
(heaven, 天堂)	0.656	0.830
(merchant, 商人)	0.561	0.832

higher accuracy on cross-lingual dictionary induction and cross-lingual document classification tasks.

We also present the cosine similarity between vectors[16] of each word pair to verify our observations in Table 6, higher is better. By comparing the result of two models, we found that BiKCCA leads to a relative improvement over BiCCA (cosine similarity between vectors in BiKCCA are higher than BiCCA).

The reason for these two observations described above is that the relationship between words of different languages can not be captured well by a linear model, and this results in longer distance between two translationally equivalent words, while our non-linear model is able to group translationally equivalent words together in the vector space.

In Figure 6, we present the PCA projection of three groups of synonym in English (embeddings are trained on En-Zh corpus). It can be easily seen that synonyms in red and blue group of (b)

[16]Vectors have been normalized before computation.

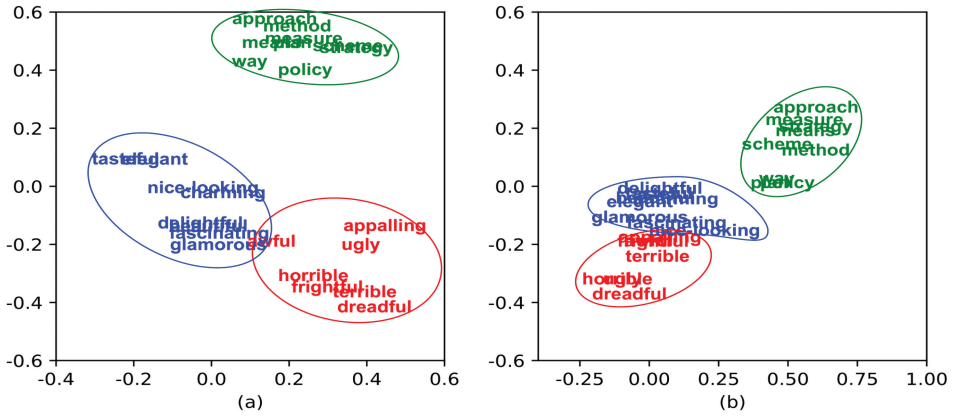

Fig. 6. PCA projection of word embeddings of three groups of near-synonym in English, different groups with different colors. BiCCA (a) and BiKCCA (b).

are closer than (a), and both models perform well in green group. This indicates that semantically similar words in the same language of BiKCCA are closer than BiCCA in most groups, and this also explains the better performance of BiKCCA on word similarity task. The reason for this phenomenon is that for English-side transformation, non-linear transformation reflects the fact better than linear transformation. In addition, we have also found that both models are good at separating antonyms (the red group and blue group are well separated in BiCCA and BiKCCA). This can be attributed to the fact that both models use bilingual dictionary, which helps in pulling apart the synonyms and antonyms. Last but not least, based on results of all experiment above, we discover that in closely related languages, BiKCCA gains little improvement in performance over BiCCA whether on intrinsic or extrinsic tasks, while in substantially different languages, BiKCCA gains great improvement. Supposing that we have captured almost all non-linear relationships, from a quantitative point of view, we could draw a conclusion that there are a lot of non-linear relationships among substantially different languages, while little among closely related languages.

Error Analysis. We analyze errors occurring most with BiCCA and BiKCCA on cross-lingual dictionary induction task. A typical error is that a word is often mistranslated into another word with close but different meaning, for example, the word "way/方法" is mistranslated into "政策/policy." This error appears in both models, with a lot in BiCCA and less in BiKCCA. According to Figure 6, we observe that semantically similar words in the same language always gather together, words like "way" and "policy" are gathered into one group, the same as "方法" and "政策." However, we have also shown that translationally equivalent words are also close in shared vector space, this causes that all semantically similar words crowd together so that word is often mistranslated. BiKCCA avoids part of such error, because translationally equivalent words are closer than noise words in vector space, while BiCCA can not reduce this error due to the limitation of linear transformation.

Model Analysis. We have proposed a model that is computationally efficient, easy to scale to large datasets, and outperforms previous linear models in chosen evaluation tasks. But our model also has two limitations—one is that it is not able to learn multiple embeddings per word. Actually, homonymy and polysemy are common in natural languages, and they are often the sources of error in word embedding algorithms. The other shortcoming of our method is that it only learns embeddings at word level, hence currently we cannot capture compositional semantics.

7 CONCLUSIONS AND FUTURE WORK

We have presented a KCCA-based non-linear approach for learning multilingual word embeddings, which addresses drawbacks of previous linear models. Based on results from extrinsic and intrinsic evaluation tasks, we show that KCCA embeddings consistently outperform previous embeddings on each task across various language pairs, especially across substantially different languages. By qualitative analysis, we note that previous linear assumption does not always hold true, and non-linear assumption is more suitable for representing the relationship among languages. In the future, we would extend our experiment to more languages to further inspect the performance of our model, and apply our method to more downstream applications such as cross-lingual dependency parsing and machine translation.

ACKNOWLEDGMENT

We thank anonymous reviewers for their insightful comments.

REFERENCES

[1] Shotaro Akaho. 2006. A kernel method for canonical correlation analysis. *CoRR* abs/cs/0609071.

[2] Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2016. Learning principled bilingual mappings of word embeddings while preserving monolingual invariance. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. 2289–2294.

[3] Francis Bond and Ryan Foster. 2013. Linking and extending an open multilingual wordnet. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, 1352–1362.

[4] Mauro Cettolo, Christian Girardi, and Marcello Federico. 2012. WIT^3: Web inventory of transcribed and translated talks. In *Proceedings of the 16th Conference of the European Association for Machine Translation (EAMT'12)*. 261–268.

[5] Chris Dyer, Miguel Ballesteros, Wang Ling, Austin Matthews, and Noah A. Smith. 2015. Transition-based dependency parsing with stack long short-term memory. 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)*. Association for Computational Linguistics, 334–343.

[6] Chris Dyer, Adam Lopez, Juri Ganitkevitch, Jonathan Weese, Ferhan Ture, Phil Blunsom, Hendra Setiawan, Vladimir Eidelman, and Philip Resnik. 2010. cdec: A decoder, alignment, and learning framework for finite-state and context-free translation models. In *Proceedings of the ACL 2010 System Demonstrations*. Association for Computational Linguistics, 7–12.

[7] Manaal Faruqui and Chris Dyer. 2014. Improving vector space word representations using multilingual correlation. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics, 462–471.

[8] Lev Finkelstein, Evgeniy Gabrilovich, Yossi Matias, Ehud Rivlin, Zach Solan, Gadi Wolfman, and Eytan Ruppin. 2001. Placing search in context: The concept revisited. In *Proceedings of the 10th International Conference on World Wide Web (WWW'01)*. ACM, New York, NY, 406–414.

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

[10] Kenji Fukumizu, Francis R. Bach, and Arthur Gretton. 2007. Statistical consistency of kernel canonical correlation analysis. *J. Mach. Learn. Res.* 8 (May 2007), 361–383.

[11] Stephan Gouws, Yoshua Bengio, and Greg Corrado. 2015. BilBOWA: Fast bilingual distributed representations without word alignments. In *Proceedings of the 32nd International Conference on Machine Learning (ICML'15)*, David Blei and Francis Bach (Eds.). JMLR Workshop and Conference Proceedings, 748–756.

[12] Stephan Gouws and Anders Søgaard. 2015. Simple task-specific bilingual word embeddings. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, 1386–1390.

[13] 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)*. Association for Computational Linguistics, 1234–1244.

[14] 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)*. Association for Computational Linguistics, 58–68.

[15] Felix Hill, Roi Reichart, and Anna Korhonen. 2014. SimLex-999: Evaluating semantic models with (genuine) similarity estimation. *CoRR* abs/1408.3456 (2014).

[16] Harold Hotelling. 1936. Relations between two sets of variates. *Biometrika* 28, 3/4 (1936), 321–377.

[17] Alexandre Klementiev, Ivan Titov, and Binod Bhattarai. 2012. Inducing crosslingual distributed representations of words. In *Proceedings of the International Conference on Computational Linguistics (COLING'12)*.

[18] David D. Lewis, Yiming Yang, Tony G. Rose, and Fan Li. 2004. RCV1: A new benchmark collection for text categorization research. *J. Mach. Learn. Res.* 5 (Dec. 2004), 361–397.

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

[20] Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *J. Mach. Learn. Res.* 9 (Nov. 2008), 2579–2605.

[21] Ryan McDonald, Joakim Nivre, Yvonne Quirmbach-Brundage, Yoav Goldberg, Dipanjan Das, Kuzman Ganchev, Keith Hall, Slav Petrov, Hao Zhang, Oscar Täckström, Claudia Bedini, Núria Bertomeu Castelló, and Jungmee Lee. 2013. Universal dependency annotation for multilingual parsing. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Association for Computational Linguistics, 92–97.

[22] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *CoRR* abs/1301.3781.

[23] Tomas Mikolov, Quoc V. Le, and Ilya Sutskever. 2013. Exploiting similarities among languages for machine translation. *CoRR* abs/1309.4168 (2013).

[24] Aditya Mogadala and Achim Rettinger. 2016. Bilingual word embeddings from parallel and non-parallel corpora for cross-language text classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

[25] Jerome L. Myers and Arnold Well. 1995. *Research Design and Statistical Analysis* (1st ed.). Lawrence Erlbaum Associates, Mahwah, NJ.

[26] Gerard Salton and Christopher Buckley. 1988. Term-weighting approaches in automatic text retrieval. *Inf. Process. Manage.* 24, 5 (Aug. 1988), 513–523.

[27] 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)*. Association for Computational Linguistics, 567–572.

[28] Richard Socher, John Bauer, Christopher D. Manning, and Andrew Y. Ng. 2013. Parsing with compositional vector grammars. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, 455–465.

[29] James H. Steiger. 1980. Tests for comparing elements of a correlation matrix. *Psychol. Bull.* 87, 2 (1980), 245.

[30] Yulia Tsvetkov, Manaal Faruqui, Wang Ling, Guillaume Lample, and Chris Dyer. 2015. Evaluation of word vector representations by subspace alignment. In *Proceedings of the Conference on Empirical Methods on Natural Language Processing (EMNLP'15)*.

[31] Shyam Upadhyay, Manaal Faruqui, Chris Dyer, and Dan Roth. 2016. Cross-lingual models of word embeddings: An empirical comparison. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, 1661–1670.

[32] Jakob Uszkoreit, Jay M. Ponte, Ashok C. Popat, and Moshe Dubiner. 2010. Large scale parallel document mining for machine translation. In *Proceedings of the 23rd International Conference on Computational Linguistics (COLING'10)*. Association for Computational Linguistics, 1101–1109.

[33] Ivan Vulić and Marie-Francine Moens. 2013. 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*. Association for Computational Linguistics, 106–116.

[34] 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)*. Association for Computational Linguistics, 719–725.

[35] Xiaojun Wan. 2009. Co-training for cross-lingual sentiment classification. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP (ACL'09)*. Association for Computational Linguistics, 235–243.

[36] Chao Xing, Dong Wang, Chao Liu, and Yiye Lin. 2015. Normalized word embedding and orthogonal transform for bilingual word translation. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (HLT-NAACL'15)*.

[37] 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)*. Association for Computational Linguistics, 111–121.

[38] Meng Zhang, Yang Liu, Huanbo Luan, Maosong Sun, Tatsuya Izuha, and Jie Hao. 2016. Building earth mover's distance on bilingual word embeddings for machine translation. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence (AAAI'16)*. AAAI Press, 2870–2876.