Bilingual Lexicon Extraction with Temporal Distributed Word Representation from Comparable Corpora

Chunyue Zhang and Tiejun Zhao(✉)

School of Computer Science and Technology,
Harbin Institute of Technology, Harbin, China
cyzhang@mtlab.hit.edu.cn, tjzhao@hit.edu.cn

Abstract. Distributed word representation has been found to be highly effective to extract a bilingual lexicon from comparable corpora by a simple linear transformation. However, *polysemous words* often vary their meanings at different time points in the corresponding corpora. A single word representation which is learned from the whole corpora can't express the temporal change of the word meaning very well. This paper proposes a simple solution which exploits the temporal distributed word representation for *polysemous words*. The experimental results confirm that the proposed solution can offer better performance on the English-to-Chinese bilingual lexicon extraction task.

1 Introduction

Over the years, the automatic extraction of bilingual lexicons (BLE) from comparable corpora, where documents are not direct translations but share a topic or domain, has attracted many researchers. In this field, how to represent a word is an essential problem. In recent years, **Distributed Word Representation** [1,15], which is often called **word embedding**, has been extensively studied. Word embedding projects discrete words to a dense low-dimensional and continuous vector space where co-occurred words are located close to each other. Often the word embedding is learned from a big text corpora. In [11], inspired by the linear relation in the bilingual scenario, a linear transform is learned to project semantically identical words from a language to another with word embedding.

However, many words are *polysemous*. When occurring in the time sequential corpora, they often vary their meaning at different time points. For example, the word *apple* in Chinese was nearly the name for some fruit in the corpora twenty years ago, but recently it's more possible to refer a technology company. A single word embedding which is learned from the whole corpora can't express the change of the word meaning with the time very well.

In fact, comparable corpora often are automatically collected from some specific multilingual information source such as Wikipedia[1] and Xinhua News

[1] https://www.wikipedia.org

J. Li et al. (Eds.): NLPCC 2015, LNAI 9362, pp. 380–387, 2015.
DOI: 10.1007/978-3-319-25207-0_33

Agency[2], whose corpora are contents updated constantly [5] at different time stamps. And the different word embedding can be trained from the corpora at the different time stamp. Furthermore, different transform matrices will be learned from the word embedding which is obtained from comparable corpora with the different time stamp. So here are two natural questions to motivate this paper:

- How does the word embedding with the different time stamp affect the quantity of learned bilingual lexicon?
- How can BLE get better performance by exploiting the word embedding with the different time stamp?

In this paper, we propose a solution by exploiting the **Temporal Distributed Word Representation** to learn a more accurate translation matrix. Specifically, in this work:

- firstly we divide the sub-corpora set into the corresponding windows according to the different time stamp,
- after fixing a common vocabulary in every corpora window, then we learn the different word embedding from the corresponding corpora,
- then we concatenate these different word embedding into a single new word embedding,
- finally, we learn a new linear transform matrix from the new word embedding.

2 Background: Linear Translation Transformation

The bilingual lexicon extraction provided in [11] learns a linear transform from the source language to the target language by the linear regression. During the training period, suppose we are given a set of bilingual word pairs and their associated word embeddings $\{x_i, z_i\}_{i=1}^{n}$, and $x_i \in R^{d_1}$ is the word embedding of word i in the source language, $z_i \in R^{d_2}$ is the word embedding of its translation.

The objective function is as follows:

$$\hat{\mathbf{W}} = \underset{W \in R^{d_2 \times d_1}}{\operatorname{argmin}} \sum_{i=1}^{n} \|Wx_i - z_i\|^2 \tag{1}$$

During the prediction period, given a new source word embedding x, the standard way to retrieve its translation word in the target language is to return the nearest neighbour (in terms of cosine similarity measure) of mapped $z = \hat{W}x$ from the set of word embedding of the target language.

[2] http://www.news.cn/english/

3 Temporal Distributed Word Representation

In [10], the authors proposed a skip-gram model to learn word embedding in which aims at predicting the context words with the word in the central position. Formally, the training process maximizes the following likelihood function with a word sequence $w_1, w_2, \ldots, w_N$:

$$\frac{1}{N}\sum_{i=1}^{N}\sum_{-C\leq j\leq C} logP(w_{i+j}|w_i) \tag{2}$$

Obviously, the word embedding learned depends on the training word sequence, i.e. the corpora. In the bilingual scenario, this corpora is the source (target) side of the comparable corpora. And often comparable corpora collected from the Internet is often labeled with the time stamp. So one can train the different word embedding for the same word with the corresponding corpora at the different time stamp. And every word embedding trained from the different corpora can represent the word meaning in a specific time slot. So exploiting these different word embeddings can represent the multiple meanings for a polysemous word better. In this paper we therefore propose the **Temporal Distributed Word Representation** which concatenates these different word embedding into a single one.

Mathematically, suppose we are given a list of sub-corpora ordered by their time stamp $C = \{C_1, C_2, \ldots, C_T\}$. For a word we can learn T different word embedding theoretically. However, we found the quality of word embedding is very poor when the size of training corpora is small. So we propose three kinds of temporal distributed word representations which can be trained from large scale corpora:

- **Sliding-Window Temporal Distributed Word Representation (STWR)**
  After empirically setting the predefined size of the corpora window M and the sliding step k, we first divide the T sub-corpora into $N = \lceil \frac{T-M}{k} \rceil$ windows. Note that $T - M$ does not have to be divisible by k and the last window can have a smaller size than M. Then we can train the skip-gram model on the window corpora set $\{SC_1, \ldots, SC_i, \ldots, SC_N\}$, where $SC_i = \{C_{1+k(i-1)}, \ldots, C_{M+k(i-1)}\}$. For a word w, we can get a word embedding list $\{sw_1, \ldots, sw_i, \ldots, sw_N\}$. Finally, we concatenate the word embedding in the list in order, i.e.

  $$\mathbf{STWR}(w) = \oplus_{i=1}^{N} sw_i \tag{3}$$

  where $\oplus$ means vector concatenation operator.
- **Accumulated Temporal Distributed Word Representation (ATWR)**
  In order to express the meaning of a word in a global time slot, we can accumulate the small sub-corpora into a larger one according to the time stamp.

As predefining M and k in **SWTR**, we can get the accumulated corpora set $\{AC_1, \ldots, AC_i, \ldots, AC_N\}$, where $AC_i = \{C_1, \ldots, C_{M+(i-1)k}\}$ and $N = \lceil \frac{T-M}{k} \rceil$. For a word w, we can get a word embedding list $\{aw_1, \ldots, aw_i, \ldots, aw_N\}$. Finally, we concatenate the word embedding in the list in order, i.e.

$$\mathbf{ATWR}(w) = \oplus_{i=1}^{N} aw_i \tag{4}$$

– **Ensemble Temporal Distributed Word Representation(ETWR)** From the definition above, **STWR** can represent the meaning of a word in a local time slot, and **ATWR** can represent a global meaning conversely. So it's natural to ensemble this two representations. We define the **Ensemble Temporal Word Representation** as follows:

$$\mathbf{ETWR}(w) = \oplus_{i=1}^{N} sw_i \oplus aw_N \tag{5}$$

where sw_i holds the word embedding trained from the slide-window corpora, and aw_N holds the word embedding learned from the whole corpora AC_N.

4 Experiment and Results

4.1 Experimental Settings

In this paper, we carry on the bilingual lexicon extraction task in the English-to-Chinese direction. For the comparable corpora, we use the English Gigaword Corpus (LDC2009T13) and the Chinese Gigaword Corpus (LDC2009T27). In order to align the two comparable corpora better, we select the part of the two corpus published by Xinhua News Agency which contains news articles from January 1995 to December 2008. So we have 14 corpus at the different year in every language.

For **STWR** and **ATWR**, we set window size $M = 10$ and sliding step $k = 2$, then get three corpora $\{SC_1, SC_2, SC_3\}$ for every language and also get three corpora $\{AC_1, AC_2, AC_3\}$ for every language where AC_1 is identical to SC_1. Details of every corpora are reported in Table 1. We intersect the vocabulary sets of $\{SC_1, SC_2, SC_3\}$ as the common vocabulary. After that, we get an English common vocabulary consisting of 92335 English words and a Chinese common vocabulary consisting of 143621 Chinese words.

From every language, we use the same setup to train the skip-model. We use the word2vec toolkit[3] to learn a 200-dimensional word embedding. We just consider the words occurred at least 10 times, and set a context windows of 3 words to either side of the target. Other hyper-parameters follows the default software setup.

To obtain a bilingual training lexicon between English-to-Chinese, we use an in-house dictionary which consists of 55668 English words and 137420 Chinese words. Firstly, we filter the dictionary with the intersection set of vocabulary of

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

Table 1. The sizes of the monolingual training dataset for **STWR** and **ATWR**

		Training Tokens	Vocabulary Size
EN	SC_1/AC_1	222M/222M	106K/106K
	SC_2/AC_2	237M/274M	112K/122K
	SC_3/AC_3	245M/326M	116K/136K
CH	SC_1/AC_1	223M/223M	163K/163K
	SC_2/AC_2	238M/275M	168K/181K
	SC_3/AC_3	266M/346M	177K/205K

$\{SC_1, SC_2, SC_3\}$. From the filtered dictionary, we randomly select 2500 different English words and their translation as test set, and the left dictionary as the training set. Details of the train set and test set are listed in Table 2.

Table 2. The statistics for the train set and test set

	Train Set	Test Set
Entries	64692	10576
Words	14413	2500
Avg	4.48	4.23

4.2 Results

We choose the approach in [11] as our baseline where a single embedding is learned from $AC3$. The performance is measured by accuracy of translation retrieval list of the test set at Top-k, where k is set $\{1, 5, 10\}$. Here accuracy means if there is one candidate in the Top-k list occurs in the reference list, the translation will be right. Results of **STWR**, **ATWR** and **ETWR** are given in Table 3. In all tables in this paper, the $AC3$ row represents the baseline approach.

From Table 3, a continuous performance improvement from $\{AC_1, AC_2, AC_3\}$ can be seen. It shows that using a larger corpus can learn better word embedding. And **ATWR** achieves significant improvements over the baseline $AC3$. **ATWR** increases the accuracy at Top-5 from 0.225 to 0.255. From Table 3, the performance achieved by **STWR** is also improved significantly over the baseline $AC3$ and is comparable with **ATWR**. As can be seen, exploiting the local word embedding at the different time stamp is effective for extracting bilingual lexicon. Finally, from Table 3, the **ETWR** performs best. This representation achieves an improvement of near 15% over the baseline $AC3$ at Top-5.

Furthermore, we also use the metric **unnormalized precision** which can measure the times of the correct translation occurring in the Top-k translation list for a word like used in information retrieval. Obviously, for a word with multiple translations in the test set, higher precision of an approach means better performance for *polysemous words*. In order to compare all the representations in this paper fairly, we choose the intersection set of words correctly predicted at the Top-10 with the representation learned with $AC3$,**ATWR**,**STWR** and

Table 3. The performance of **ATWR**,**STWR** and **ETWR**

Representation	ACC@1	ACC@5	ACC@10
AC1	0.096	0.204	0.251
AC2	0.110	0.216	0.280
AC3	0.126	0.225	0.292
SC1	0.096	0.204	0.251
SC2	0.103	0.209	0.269
SC3	0.108	0.221	0.274
ATWR	**0.131**	**0.255**	**0.314**
STWR	**0.132**	**0.251**	**0.307**
ETWR	**0.136**	**0.260**	**0.316**

Table 4. The average precision of L234 and L5 at Top-10

Representation	Prec@L234	Prec@L5
AC3	1.217	1.436
ATWR	1.233	1.513
STWR	1.272	1.532
ETWR	**1.281**	**1.544**

ETWR as the evaluation set. Then we choose two subsets $\{L234, L5\}$ from this evaluation set according to the number of translation for a word, where $L234$ means the word has 2, 3 or 4 translations and $L5$ means the word has at least 5 translations. In this setting, $L234$ has 253 words and $L5$ has 261 words. From the Table 4, we can see the performance of all the temporal word representations proposed in this paper can outperform the baseline $AC3$, and the **ETWR** gains the most significant improvement.

5 Related Works

In the BLE task from comparable corpora, most of the previous methods are based on the distributional hypothesis that a word and its translation tend to appear in similar contexts across languages [2–4,13]. Based on this assumption, generally an unsupervised standard approach [9] using **Co-occurrence Word Representation** calculates the context similarity and then extract word translation pairs with high similarity.

Another interesting word representation is topic word representation in [16,17]. They train a cross-language topic model on the document-aligned comparable corpora. It attempts to abrogate the need of seed lexicon. However, the bilingual topic representation must be learned from aligned documents.

Recently some supervised approaches have been tried to solve this task. An linear classifier in [7] and a Random Forest classifier [8] are used to automatically decide if two words in source language and target language are translated each other. In [11], a linear transform is learned to project semantically identical words from one language to another. In this approach, the word is represented

with a continues and dense vector i.e. word embedding. It is surprising that this approach achieved a high accuracy on a bilingual word translation than the standard approach.

All the above methods just learn the translation for a word with a single representation. Its use is problematic when a word has several translations. Discovering multiple senses embedding per word type is the focus of [6,12]. Compared with these context-based approaches, our method is based on the observed fact on the corpora that the word meaning often varies at the different time slot. We exploit the time information to learn multiple word representations.

Temporal information is firstly used in [14]. And a similar approach in [7] uses the frequency distribution on the corpora at the different time stamp which is estimated and as a feature of a classifier. Compared with these methods, our approach exploits temporal distributed word representation which is more robust and continuous.

6 Conclusions

We presented a simple but effective method that exploiting the temporal distributed word representation to learn the linear transform matrix. Three temporal distributed word representations are used for this purpose. This method can learn multiple translations for polysemous words better. Experiments conducted on an English-Chinese comparable corpora indicate that the three temporal word representations all improve the baseline significantly and **ETWR** performs best. By measuring the average unnormalized precision in the Top-10 list, it's better shown that the temporal distributed word representation is effective for the translation of polysemous words.

Acknowledgments. This work is supported by the project of National Natural Science Foundation of China (61173073, 61272384) and International Science and Technology Cooperation Program of China (2014DFA11350).

References

1. Bengio, Y., Courville, A., Vincent, P.: Representation learning: A review and new perspectives. IEEE Transactions on Pattern Analysis and Machine Intelligence **35**(8), 1798–1828 (2013)
2. Chiao, Y.C., Zweigenbaum, P.: Looking for candidate translational equivalents in specialized, comparable corpora. In: Proceedings of the 19th International Conference on Computational Linguistics, vol. 2, pp. 1–5. Association for Computational Linguistics (2002)
3. Emmanuel, M., Hazem, A.: Looking at unbalanced specialized comparable corpora for bilingual lexicon extraction. In: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL), pp. 1284–1293 (2014)
4. Fung, P., Yee, L.Y.: An ir approach for translating new words from nonparallel, comparable texts. In: Proceedings of the 17th International Conference on Computational Linguistics, vol. 1, pp. 414–420. Association for Computational Linguistics (1998)

5. Huang, D., Zhao, L., Li, L., Yu, H.: Mining large-scale comparable corpora from Chinese-English news collections. In: Proceedings of the 23rd International Conference on Computational Linguistics: Posters, pp. 472–480. Association for Computational Linguistics (2010)
6. Huang, E.H., Socher, R., Manning, C.D., Ng, A.Y.: Improving word representations via global context and multiple word prototypes. In: Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers, vol. 1, pp. 873–882. Association for Computational Linguistics (2012)
7. Irvine, A., Callison-Burch, C.: Supervised bilingual lexicon induction with multiple monolingual signals. In: HLT-NAACL, pp. 518–523. Citeseer (2013)
8. Kontonatsios, G., Korkontzelos, I., Tsujii, J., Ananiadou, S.: Using a random forest classifier to compile bilingual dictionaries of technical terms from comparable corpora. In: Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics: Short Papers, vol. 2, pp. 111–116 (2014)
9. Laroche, A., Langlais, P.: Revisiting context-based projection methods for term-translation spotting in comparable corpora. In: Proceedings of the 23rd International Conference on Computational Linguistics, pp. 617–625. Association for Computational Linguistics (2010)
10. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space (2013). arXiv preprint arXiv:1301.3781
11. Mikolov, T., Le, Q.V., Sutskever, I.: Exploiting similarities among languages for machine translation (2013). arXiv preprint arXiv:1309.4168
12. Neelakantan, A., Shankar, J., Passos, A., McCallum, A.: Efficient non-parametric estimation of multiple embeddings per word in vector space (2015). arXiv preprint arXiv:1504.06654
13. Rapp, R.: Automatic identification of word translations from unrelated english and german corpora. In: Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics on Computational Linguistics, pp. 519–526. Association for Computational Linguistics (1999)
14. Schafer, C., Yarowsky, D.: Inducing translation lexicons via diverse similarity measures and bridge languages. In: Proceedings of the 6th Conference on Natural Language Learning, vol. 20, pp. 1–7. Association for Computational Linguistics (2002)
15. Turian, J., Ratinov, L., Bengio, Y.: Word representations: a simple and general method for semi-supervised learning. In: Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, pp. 384–394. Association for Computational Linguistics (2010)
16. Vulić, I., De Smet, W., Moens, M.F.: Identifying word translations from comparable corpora using latent topic models. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers. vol. 2, pp. 479–484. Association for Computational Linguistics (2011)
17. Vulić, I., Moens, M.F.: Detecting highly confident word translations from comparable corpora without any prior knowledge. In: Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics, pp. 449–459. Association for Computational Linguistics (2012)