Structured Vectors for Chinese Word Representations

Changliang Li, Bo Xu, Xiuying Wang, Gaowei Wu, Guanhua Tian, and Wendong Ge

***Abstract*—The use of word representations has been a key reason for the success of many NLP tasks. A lot of work has focused on improving the learning of word representations, and most approaches treat word as atomic unit. However, in some languages, for example Chinese, some words cannot be recognized correctly. This leads to the corruption of word embeddings' ability to capture semantic information. This paper addresses this shortcoming by proposing structured embeddings for word representations. Our method utilizes sub-word and atomic unit embeddings to represent word embeddings. We build structured vectors for Chinese word representations based on the method, and evaluateon SemEval-2012 Task 4: Measuring Chinese word similarity. The result shows that our method is remarkably effective in capturing semantic information and outperforms previous best performance by a large margin. Our method can be extended to the languages which do not have a trivial word segmentation process.**

***Index Terms*—Word embeddings, word segmentation, semantic information.**

I. Introduction

Word embeddings have recently demonstrated outstanding results across various NLP tasks, such as information retrieval [1], search query expansions [2], and representing semantics of words [3]. It is rewarding to obtain good performance word embeddings.

To the best of our knowledge, most of the existing learning methods aim to obtain English word embeddings and treat word as atomic unit [4]. In English and many other languages using some form of the Latin alphabet, the space is a good approximation of a word divider.

However, the equivalent to this character is not found in all written scripts, and without it word segmentation is a difficult problem. Languages which do not have a trivial word segmentation process include Chinese and Japanese, where sentences but not words are delimited, Thai and Lao, where phrases and sentences but not words are delimited, and Vietnamese, where syllables but not words are delimited [5].

In this case, it is hard to have a "good" segmenter. It is almost impossible to segment a sentence perfectly. In fact even human has trouble to segment some ambiguous sentences.

For example, the Chinese word "素食主义者", corresponding to English word "Vegetarian", is composed of five Chinese characters. To our best knowledge, even state of the art cannot recognize it correctly. It can only be recognized as three words: "素食", "主义" and "者". As a result, we cannot obtain the embeddings of words like "素食主义者".

When comes to this linguistic phenomenon, the existing method soflearning word embeddings are to replace the word that cannot be recognized correctly by a common "unknown" token. Under such methods, the ability of word embeddings to capture semantic information is corrupted.

In this paper, we propose novel structured word representations based on sub-word and atomic unit embeddings. The property of our method is to train language models on sub-word and atomic unit level. We define atomic unit as one character, and sub-word as the word that is more than one character. For example, "素食" and "主义" are sub-words and "者" is an atomic unit. And then we represent word embeddings as combing its sub-word and atomic unit embeddings.

We take Chinese as an example. We evaluate structured word embeddings on SemEval-2012 Task 4: measuring Chinese word similarity[1]. Despite its simplicity, our method works well in capturing words' semantic information. The experiments results show that using structured word embeddings on SemEval-2012 Task 4outperforms the previous best performance by a large margin.

II. Related Work

A word representation is a mathematical object associated with each word, often a vector. Each dimension's value corresponds to a feature and might even have a semantic or grammatical interpretation, so we call it a word feature [2]. A lot of work has been made on this task [6]-[8]. One common approach to inducing word representation is to use clustering, perhaps hierarchical. This technique was used by a variety of researchers [9]-[13]. This leads to a one-hot representation over a smaller vocabulary size.

Neural language models [14], [15], on the other hand, induce dense real-valued low-dimensional word embeddings using unsupervised approaches. Historically, training and testing of neural language models has been slow, scaling as the size of the vocabulary for each model computation. Many approaches have been proposed to eliminate that linear dependency on vocabulary size and allow scaling to very large training corpora.

Collobert and Weston presented a neural language model that could be trained over billions of words, because the gradient of the loss was computed stochastically over a small sample of possible outputs, in a spirit similar to Bengio [16]. This neural model of Collobert and Weston was refined and presented ingreater depth in [17], [18].

It was found that word representations could capture meaningful syntactic and semantic regularities in a very simple way, such as the singular/plural relation that V_{apple} -

Manuscript received October 11, 2014; revised April 21, 2015.

The authors are with the Institute of Automation Chinese Academy of Sciences 95 Zhongguancun East Road, 100190, Beijing, China (e-mail: changliang.li@ia.ac.cn, xubo@ia.ac.cn, xiuying.wang@ia.ac.cn, gaowei.wu@ia.ac.cn, guanhua.tian@ia.ac.cn, wending.ge@ia.ac.cn).

[1] http://www.cs.york.ac.uk/semeval-2012/task4/

DOI: 10.7763/IJMLC.2015.V5.538

$V_{\text{apples}} \approx V_{\text{car}}$ - V_{cars}. The regularities are observed as constant vector offsets between pairs of words sharing a particular relationship. It is also true for a variety of semantic relations, as measured by the Sem Eval 2012 task of measuring relation similarity [19].

Vector space word representations have been successfully at improving performances across a variety of NLP tasks. Much work has been focused on improving word embeddings. Socher *et al.*, recently proposed several kinds of recursive neural networks language models, such as RNN, MV-RNN, RNTN [20]-[24]. Mikolovetal, presented language model based on recurrent neural networks [25], [26]. Most of the work focuses on improvement of language model. Meanwhile, corpus also plays an important role in training word embeddings. There are several corpora publicly available, most of them are English.

III. Language Model

In this section, we describe model architecture for learning distributed representations of words that try to minimize computational complexity.

We employed skip-gram model to train sub-word and atomic unit embeddings due to its low computational complexity. It tries to maximize classification of a word based on another word in the same sentence. More precisely, it uses each current word as an input to a log-linear classifier with continuous projection layer, and predicts words within a certain range before and after the current word. Fig. 1 shows its graphical representation [26].

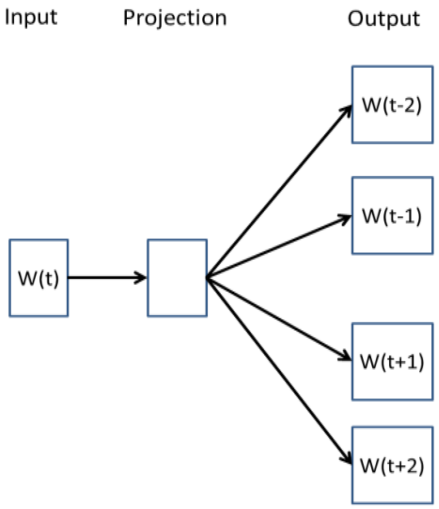

Fig. 1. Skip-gram model.

The objective function of Skip-gram model is to maximize the average log probability described as formula (1).

$$\frac{1}{T}\sum_{t=1}^{T}\sum_{j=-k}^{k}\log p(w_{t+j}|w_t) \qquad (1)$$

where k is the size of the training window. The inner summation goes from $-k$ to k to compute the log probability of correctly predicting the word w_{t+j} given the word in the middle w_t. The outer summationgoes over all words in the training corpus.The Skip-gram model is trained through using stochastic gradient descent, which is computed via back propagation algorithm [27].

IV. Structured Word Representations

We train language model on sub-word level and atomic unit level. Then we employ the obtained embeddings to represent structured word embeddings. The details of the method are described as follows.

A. Model

We firstly use maximum matching algorithm [28] to split W_i into one or more sub-words and atomic units. And then we represent word embeddings as combining its sub-word and atomic-unit embeddings. The final word embeddings$V(W_i)$ is represented as formula (2).

$$V(W_i) = Func\{\alpha_n \sum_{n=1}^{n=N} V(At\_U_i^n) + \beta_m \sum_{m=1}^{m=M} V(sub\_W_i^m)\} \qquad (2)$$

where $At\_U_i^n$ means the nth atomic unit of word W_i ; $sub\_W_i^m$ is the mth sub-word of word W_i ; $Func(.)$ is normalized function; N is the number of atomic units of the word; M is the number of sub-word of the word; α_n and β_m are two scaling parameters used to prevent either pair of sides from dominating the other. Sub-word and atomic unit embeddings are special cases of structured word embeddings. For example, when one word is composed of only one atomic, $N = 1$ and $M = 0$; while when one word is composed of only one sub-word, $N = 0$ and $M = 1$.

Fig. 2 shows the process of obtaining structured word representations. The role of WS model in Fig. 2 is to split word into sub-word(s) and atomic unit(s).

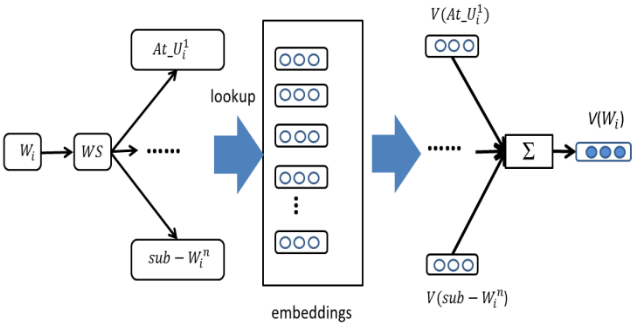

Fig. 2. Of obtaining structured word representations.

For example, the Chinese word “素食主义者” (corresponding to English word “vegetarian”) cannot be recognized correctly. Instead, it is recognized as two sub-words and one character (atomic unit), “素食” (corresponding to “vegetables” in English), “主义” (corresponding to “doctrine” in English), and “者” (corresponding to “person” in English). The embeddings of Chinese word “素食主义者” is represented as formula (3) using our method.

$$V(\text{"素食主义者"}) = \text{Func}\{\beta_1 V(\text{“素食”}) + \beta_2 V(\text{“主义”}) + \alpha 1 V\text{“者”} \qquad (3)$$

Fig. 3 shows the example on vector space. The vectors were projected down to two dimensions using PCA algorithm.

Each part of one word’s components, sub-word or atomic unit, has much relation with the word. Our method uses structured embeddings to combine the word's all components embeddings. So the obtained word embeddings are supposed to capture the word's semantic information. This is in line with the original idea of word embeddings. Each dimension of the embeddings represents a latent feature of the word, hopefully capturing useful syntactic and semantic properties.

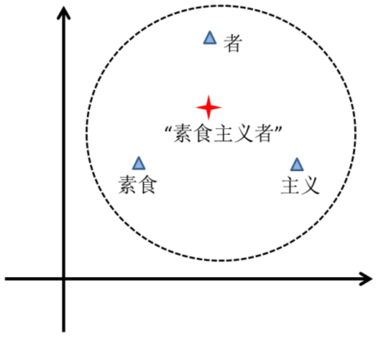

Fig. 3. Example of structured word embeddings.

B. *Learning*

Word similarity is employed to evaluate the quality of word embeddings. It is thought that the word embeddings is reliable in capturing semantic information, if the word similarity score obtained via using word embeddings is close to human annotators' judgments. The details about learning parameters α_n and β_m in formula (2) are described as follows.

W_i^1 and W_i^2 mean a pair of words. Their center word embeddings, $V(W_i^1)$ and $V(W_i^2)$, are obtained via formula (2). X_i is the similarity score given by using cosin function on the pair or words, as illustrated in formula (4).

$$X_i = \cos\left(V(W_i^1), V(W_i^2)\right) \tag{4}$$

While Y_i is the similarity score assigned by human annotators. Both X_i and Y_i are normalized. The objective function to maximize is described as formula (5).

$$\min \sum_{i=1}^{N} |Y_i - X_i|^2 \tag{5}$$

N is the total number of word pairs. There are various algorithms to search the optimal value. We take the derivative with respect to the parameters using stochastic gradient descent.

V. Experiment

In order to illustrate the effectiveness of our method, we evaluate the word embeddings obtained through our method on word similarity task. We collected Baidu Baike [2] documents as the corpus due to its wide range of topics and word usages, and its clean organization of document by topic.

And we employed ICTCLAS[3] as word segmentation tool. We employed word2vec[4] tool to compute baseline word (including sub-word and character) embeddings. Word2vecprovides an efficient implementation of skip-gram architectures discussed in previous section. Since we had obtained baseline word embeddings, we used formula (2) to build structured word embeddings. We used 50-dimensional embeddings.

A. *SemEval-2012 Task 4*

SemEval-2012 Task 4 provides a benchmark for evaluating the performance of semantic similarity calculating approaches for Chinese word pairs. Trial dataset includes 50 word-pairs, which are used to learn parameters in our work; Test dataset includes 297 word pairs with similarity scores estimated by humans [29].

It also provides an order list, in which all word pairs are listed in descending order with similarity scores (i.e., the gold order). And any candidate tested approach is requires to give similarity scores for each test word pairs, and then list them in descending order (i.e., the predicted order). And the better the predicted order preserves the order of word pairs in the gold order, the better the performance of the tested semantic similarity approach is.

In order to make this comparison easy to calculate, SemEval-2012 Task 4 provides a measure, the Kendall Rank Correlation Coefficient:

$$\tau = 1 - \frac{2S(\pi,\sigma)}{N(N-1)/2} \tag{6}$$

where N is the number of objects.π and σ are two distinct orderings of an object in two ranks. $S(\pi,\sigma)$ is the minimum number of adjacent transpositions needing to bring π and σ [30]. In this metric, τ 's value ranges from -1 to +1, and -1 means that the two ranks are inverse to each other and, +1 means the identical rank [29].

B. *Results*

We evaluate the quality of structured word embeddings on SemEval-2012 Task 4. Detailed performance of structured word embeddings, referred as Structured-E, is given in Table I. We also report the previous results.

TABLE I: Results of Different Methods on SemEval-2012 Task 4

Method	*Kendall'sτ coefficient*
Structured-E	**0.0643**
MIXCC	0.05
MIXCD	0.04
Guo-ngram	0.007
Guo-words	-0.011

From the result, we can see that using structured word embedding soutperforms the previous best performance MIXCC by a large margin.

The result shows that structured word embeddings is remarkably effective in capturing semantic information.

TABLE II: Results of Structured and Baseline Word Embeddings on SemEval-2012 Task 4

Method	*Spearman's correlation*	*Kendall'sτ coefficient*
Structured-E	**52.79**	**0.0643**
Base-E	50.58	0.0303

Furthermore, we compare the quality of structured word embeddings to baseline word embeddings (referred as Base-E) using both Kendall'sτ coefficient and Spearman's rank correlation $\rho \times 100$. Spearman's rank correlation is

[2] http://baike.baidu.com /
[3] http://www.ictclas.org/
[4] http://word2vec.googlecode.com/svn/trunk/

used to gauge how well the relationship between two variables, the similarity scores given by various methods and the human annotators. The result is shown in Table II.

The result highlights the fact that structured word embeddings can capture more semantic information than baseline word embeddings. This is expected since baseline word embeddings cannot represent the words that cannot be recognized correctly by word segmentation.

VI. Conclusion

This paper has presented novel structured word embeddings. Its property is to represent one word’s embeddings as combing its sub-word(s) and atomic unit(s) embeddings. Our method eliminates the impact brought by word segmentation, which is one of difficult problems in NLP tasks. The results have shown that structured word embeddings is remarkably effective in capturing semantic information.

In this paper, we took Chinese as example. However, it can be used to extend to any language which does not have a trivial word segmentation process. A promising direction for our further work is to utilize structured word embeddings to improve performance of many NLP tasks.

Acknowledgment

This work is supported by National Program on Key Basic Research Project (973 program) under Grant: 2013CB329302 and the National Natural Science Foundation of China under Grant No. 61175050&NO.61203281&No.61303172.

References

[1] M. P. Ca, D. Lin, J. Bigham, A. Lifchits, and A. Jain, “Names and similarities on the web: Fact extractionin the fast lane,” *ACL*, 2006.

[2] R. Jones, “Generating query substitutions,” in *Proc. the 15th International Conference on worldwide Web*, 2006.

[3] J. Reisinger and R. J. Mooney, “Multi-prototypevector-space models of word meaning,” *NAACL*, 2010.

[4] J. Turian, L. Ratinov, and Y. Bengio, “Word representations: A simple and general method for semisupervised learning,” *ACL*, 2010.

[5] From Wikipedia, the Free Encyclopedia. [Online]. Available: http://en.wikipedia.org/wiki/Text_segmentation

[6] T. Mikolov, M. Karafi ́at, L. Burget, J. Cernock ́y, and S. Khudanpur, “Recurrent neural network based language model,” *Interspeech*, 2010.

[7] A. Mnih and G. E. Hinton, “A scalable hierarchical distributed language model,” *NIPS*, pp. 1081-1088, 2009.

[8] M. Luong, R. Socher, and C. Manning, “Better word representations with recursive neural networks for morphology,” *CONLL*, 2013.

[9] Y. Bengio, J. Louradour, R. Collobert, and J. Weston, “Curriculum learning,” *ICML*, 2009.

[10] T. Koo, X. Carreras, and M. Collins, “Simple semi-supervised dependency parsing,” *ACL*, pp. 595-603, 2008.

[11] S. Miller, J. Guinness, and A. Zamanian, “Name tagging with word clusters and discriminative training,” *HLT-NAACL*, pp. 337-342, 2004.

[12] P. Liang, “Semi-supervised learning for natural language,” Master’s thesis, Massachusetts Institute of Technology, 2005.

[13] Y. Bengio, *Neural Net Language Models Scholar Pedia*, vol. 3, no. 3881, 2008.

[14] Y. Bengio, R. Ducharme, and P. Vincent, “A neural probabilistic language model,” *NIPS*, 2001.

[15] Y. Bengio, R. Ducharme, and P. Vincent, and C. Jauvin, “A neural probabilistic language model,” *Journal of Machine Learning Research*, vol. 3, pp. 1137-1155, 2003.

[16] R. Collobert and J. Weston, “A unified architecture for natural language processing,” *Deep Neural Networks with Multitask Learning*, 2008.

[17] F. Morin and Y. Bengio, “Hierarchical probabilistic neural network language model,” *AISTATS*, 2005.

[18] T. Mikolov, W. T. Yih, and G. Zweig, “Linguistic regularities in continuous space word representations,” *NAACL-HLT*, 2013.

[19] S. Miller, J. Guinness, and A. Zamanian, “Name tagging with word clusters and discriminative training,” *HLT-NAACL*, pp. 337-342, 2004.

[20] R. Socher, J. Pennington, E. H. Huang, A. Y. Ng, and C. D. Manning, “Semi-supervised recursive auto encoders for predicting sentiment distributions,” *EMNLP*, 2011.

[21] R. Socher, C. Manning, and A. Ng, “Learning continuous phrase representations and syntactic parsing with recursive neural networks,” presented at NIPS* 2010 Workshop on Deep Learning and Unsupervised Feature Learning, 2010.

[22] R. Socher, C. C. Lin, A. Ng *et al*., “Parsing natural scenes and natural language with recursive neural networks,” in *Proc. the 28th International Conference on Machine Learning*, 2011, pp. 129-136.

[23] R. Socher, J. Bauer, C. D. Manning *et al*., “Parsing with compositional vector grammars,” in *Proc. the ACL Conference*, 2013.

[24] T. Mikolov, S. Kombrink, L. Burget, J. Cernock ́y, and S. Khudanpur, “Extensions of recurrent neural network language model,” *ICASSP*, 2011.

[25] T. Mikolov and G. Zweig, “Context dependent recurrent neural network language model,” *SLT*, 2012.

[26] T. Mikolov, K. Chen, G. Corrado, and J. Dean, “Efficient estimation of word representations in vector space,” *ar Xiv preprintar Xiv*, vol. 1301, no. 3781, 2013.

[27] D. E. Rumelhart, G. EHinton, and R. J. Williams, “Learning representations by back-propagating errors,” *Nature*, vol. 323, no. 6088, pp. 533-536, 1986.

[28] H. Guo, Z. Y. Shu, W. Wang, and J. Chui, *An Augmented MM Algorithm of Word Segmentation Microcomputer Application*, vol. 18, no. 1, pp. 13-15, 2002.

[29] P. Jin and Y. Wu, “*Sem*eval-2012 task 4, evaluating chinese word similarity,” in *Proc. The First Joint Conference on Lexical and Computational Semantics*, Association for Computational Linguistics, vol. 1, 2012.

[30] M. Lapata, “Automatic evaluation of information ordering: Kendall's tau,” *Computational Linguistics*, vol. 32, no. 4, pp. 471-484, 2006.

Changliang Li was born in 1985. He received the B.S. degree in communication science from Northeast University. Now, he is a Ph.D. in Institute of Automation, Chinese Academy of Sciences. His research interest includes natural language processing, big data, deep learning.

He has published several papers on improving word embeddings, including English and Chinese. His work has been employed in industry task. Mr. Lionce was honored as the “excellent student” and “excellent Ph.D.”.

Wendong Ge is an assistant professor in Institute of Automation, Chinese Academy Sciences (IACAS). He received the B.S. degree in biomedical engineering from Tianjin Medical University in 2006, the M.S. degree in information and communication engineering from Southeast University (NIT) in 2009, and Ph.D. degrees in information and communication engineering from Beijing University of Posts and Telecommunications (BUPT) in 2012. His current research interests include spoken dialogue systems, dialogue management, reinforcement learning and deep learning.

He was involved in several projects funded by the national high technology research and development program of China and national natural science foundation of China. From November 2010 to November 2011, he was a joint Ph.D. student at the University of British Columbia, Vancouver, BC, supported by China Scholarship Council.