International Journal of Database Theory and Application
Vol.9, No.7 (2016), pp.31-38
http://dx.doi.org/10.14257/ijdta.2016.9.7.04

Incorporating Topic Priors into Distributed Word Representations

Xin Zhang[1], Bingquan Liu[1,*], Baoxun Wang[2], Xiaolong Wang[1] and Deyuan Zhang[3]

[1.] *School of Computer Science and Technology, Harbin Institute of Technology, 92 West Da Zhi St, Harbin, China*
[2.] *Application and Service Group, Microsoft, Beijing, China*
[3] *School of Computer, Shenyang Aerospace University, Shenyang, China*
Email:liubq@insun.hit.edu.cn

Abstract

Representing words as continuous vectors enables the quantification of semantic relationships of words by vector operations, thereby has attracted much attention recently. This paper proposes an approach to combine continuous word representation and topic modeling, by encoding words based on their topic distributions in the hierarchical softmax, so as to introduce the prior semantic relevance information into the neural networks. The word vectors generated by our model are evaluated with respect to word relevance and the document relevance. Experimental results show that our approach is promising for further improving the quality of word vectors.

Keywords*: distributed word representation; Continuous Bag-of-Words (CBOW); hierarchical softmax; latent dirichlet allocation*

1. Introduction

One of the recent Natural Language Processing (NLP) studies' concerns is the distributed representation of words, and it is also considered as a potential powerful technique that will help to provide better solutions to many NLP tasks [1-2]. The goal of the distributed representation (or word embedding) is to find a new semantic space for quantifying the semantic relevance of the words, that is, given the words' distributed representations, between the words in the vector space can reflect the relevance of the corresponding words.

Various word embedding methodologies can be taken to represent words as real-valued vectors [1, 3–5]. The neural network models presented by Mikolov *et al.* [4, 6] attract much attention recently for their efficiency and promising results. Two architectures (Continuous Bag-of-Words (CBOW) and Skip-gram) are introduced in his work to successfully model context information of a word for optimizing its vector representations. In general, a training process for obtaining such vector representation requires very large computation burden. Some strategies, *e.g.*, simplifying network structures or using the hierarchical softmax, are thus taken to speedup the training process.

Hierarchical softmax is a good choice for reducing the computational complexity when training the probabilistic neural network language models [7]. Generally, to adopt the hierarchical softmax method, the vocabulary should be reorganized via hierarchical word clustering [7-8], so as to achieve the binary encoding for each word. In the studies of Mikolov *et al.* [4, 6], the vocabulary is represented as a Huffman binary tree. Basically, the Huffman tree based strategy follows the observation that the frequency can be taken as a criterion for "classifying" the words, and it shows that the models presented by Mikolov *et al.* [4, 6] perform even better than the others, with the simple Huffmann tree vocabulary.

ISSN: 2005-4270 IJDTA
Copyright ⓒ 2016 SERSC

Although Huffman tree worked effectively, this frequency based word encoding approach usually neglects the semantic relevance among words. Apparently, two words with the same frequency will be allocated with similar codes based on the Huffman tree vocabulary, but the words are possible to be totally irrelevant. Noticing that some priori information can be implicitly introduced into the neural networks by means of word encoding, the quality of word vectors is expected to be further improved if semantic relations of words can be taken as a criterion for word encoding.

In this paper, we propose an approach to combine the continuous word representation model and the topic model, under the two architectures, namely, CBOW and skip-gram. A topic model is used to provide probability distribution information for each word, which is introduced into the architectures for word embedding as the prior semantic relevance knowledge. In detail, latent dirichlet allocation (LDA) is applied to generate word probability distributions over all topics for a given word. Then the topic distribution vectors are used to construct the binary codes for a word instead of the frequency based tree structure. Experimental results on an arbitrarily made evaluation data set have confirmed the effectiveness and validity of our approach. Moreover, a document representation method via the word vectors is also presented and evaluated.

The rest of the paper is organized as follows: Section 2 surveys the related work. Section 3 details our approach. Experimental results are given and analyzed in Section 4. Finally, conclusions and future directions are drawn in Section 5.

2. Related Work

The earliest research on representing words as continuous vectors dates back to 1986 [9]. Generally, the neural network language model proposed by [1] is considered as a typical approach that successfully introduces distributed word representations into NLP.

The development of deep learning techniques has brought new ideas and architectures to word representation studies. Collobert *et al.* [3, 10] present a deep architecture to learn the model for several NLP tasks jointly. In the meantime, word embedding results can also be obtained. The recurrent neural network is also taken to learn the continuous word vectors [5, 11], and the corresponding language models show promising performance.

Basically, the continuous word vectors can be achieved by introducing the 'lookup table' into the neural network trained for the given NLP task and performing the optimization on it, whose non-linear hidden layer leads to high computational complexity. In [4, 6], the simpler but effective networks are built by adopting log-linear architectures purely for the word embedding goal. The word vectors generated by such models have been utilized on the machine translation [2] and visual recognition [12-13].

Organizing the vocabulary as the hierarchical structure is the essential procedure of hierarchical softmax, which plays a great role in constructing the efficient word embedding models. [7] Have given a hierarchical word clustering method based on the prior knowledge extracted from WordNet. A data-driven approach is proposed in [8] to cluster the words hierarchically for training the model. The vocabulary in the models presented in [4, 6] is based on the words' frequencies, which is actually a strategy without expert knowledge.

3. Approach

In this section, we will describe our approach to merging the topic model with the log-linear networks for the distributed word representation. The motivation of our work is to provide the Continuous Bag-of-Words (CBOW) and the Skip-gram architectures with the prior word relevance information. As mentioned above, the word encoding process can implicitly introduce certain prior knowledge to the word embedding model. The reason is the binary codes are considered as the targets during the word vector learning via hierarchical softmax, thus heuristically if two words' semantic relevance can be partly

 Copyright ⓒ 2016 SERSC

represented by means of their binary codes, the training results of the continuous word vectors will be influenced by the semantic relevance information consequently. Comparing with the frequency based encoding strategy, we believe that the topic relevance information will deliver reasonable prior semantic knowledge.

3.1. Word Encoding via Topic Distributions

In the original word embedding architectures proposed by Mikolov *et al.* [4, 6], the word encoding procedure for hierarchical softmax is conducted based on the Huffman tree, which obviously takes words' frequencies as the basis. Although the current CBOW and Skip-gram architectures have got promising results, we aim to introduce more word relevance knowledge into the models as expected by [7], so as to further improve the quality of the learnt word vectors The essence of our approach is to generate the words' binary codes according to their topic distributions achieved by LDA, and such codes will take the place of the ones obtained based on the Huffman Tree. The illustration of our topic distribution based word encoding approach is given in Figure 1. The functions of the projection layer and the output layer are the same as the models in [4], which means that the projection layer is taken as the main component for modeling the context of the given words for different architecture (CBOW or Skip-gram), and hierarchical softmax is performed on the output layer.

For each word *w* in the vocabulary, we use LDA to get its probabilities of being generated by the topics, and the probabilities can be seen as their 'distributions' on such topics. The real-valued topic distributions of words are transformed to the binary vectors by the sampling process. A real-valued 'base' vector is adopted for the transformation by the value comparison on the corresponding dimensions. Actually there are various methods for obtaining the base vector, and in this paper this vector is simply computed by averaging each dimension of the original vectors. After topic sampling, the binary codes are considered as the targets for hierarchical softmax, and the rest processes are almost the same with the original CBOW and Skip-gram architectures except the strategy to select the output neuron.

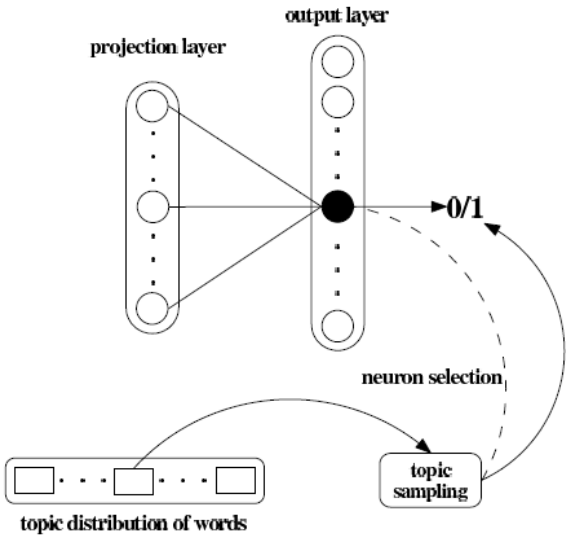

Figure 1. Illustration of the Topic Distribution Based Word Encoding

It should be noted that only one unit in the output layer is activated when performing softmax on each hierarchy, thus the proper neuron selection method is needed. For the model with the vocabulary organized by Huffman tree, the neuron selection is based on

Copyright ⓒ 2016 SERSC

the array structure of the Huffman tree. In our methodology, the output unit to be activated is selected according to the sampling base vector as follows:

$$Neuron_{w,i} = (N \times Base_i + hash(w)) \bmod K \quad (1)$$

Where Neuron $_{w,i}$ denotes the index of the output unit to be activated when predicting the *i*th element of the binary code of word *w*. *N* denotes a large multiplier, and *K* is for the number of the units in the output layer. We use $Base_i$ to denote the *i*th element of the base vector in the sampling process, and *hash (w)* stands for the hash value of word *w*.

3.2. Topic Modeling

A classic LDA [14] is performed on experimental corpus for retrieving word topics. LDA is a generative model that allows sets of observations to be explained by unobserved groups that explain why some parts of the data are similar.

Given a topic model trained on a corpus with *K* topics, all the units in the corpus are assigned with a word-topic probability ϕ_i corresponding to topic *k*. Therefore, a word *w* has a topic vector $\vec{\phi}_w = \{\phi_1, \phi_2, ..., \phi_k\}$.

4. Experiments

4.1. Experimental Setting

This paper takes the 20-newsgroups dataset as our corpus for learning the word vectors and conducting the corresponding evaluating tasks. The corpus contains approximately 20,000 newsgroup documents, partitioned (nearly) evenly across 20 different newsgroups. For the word embedding models, the smaller-scale corpus is more challenging and simulates the cases that the scale of the corpus is limited.

We choose the hierarchical softmax as the learning algorithm, and the dimension of the word vectors are set as 100. The context size is 5 and 10 for CBOW and Skip-gram respectively.

4.2. Evaluation Tasks

Two tasks are defined to evaluate the quality of the word vectors learned by our models and the original architectures proposed by Mikolov *et al.* [4, 6].

4.2.1. Relevance Quantification for Word Pairs

The word embedding approaches aim to map the words into a continuous semantic space, that is, if the words are semantically relevant with each other, they will be closer in the semantic space. Intuitively, the performance of the word embedding models can be evaluated by quantifying the relevance for the semantically relevant word pairs based on the vectors generated by the models.

For this evaluating task, a set of 200 highly relevant word pairs is manually built, and another 200 word pairs are randomly selected. Orienting to the corpus, the selected word pairs have covered the categories of dataset, while the randomly selected ones can be considered as irrelevant pairs. In this task, we evaluate the quality of the word vectors by computing the average cosine similarity of the word pairs. It is expected that the vectors with higher quality will deliver higher average similarity on the semantically relevant pairs.

4.2.2. Relevance Quantification for Document Pairs

The ultimate goal of the continuous word representation research is to promote the solutions to various NLP tasks, by providing the word vectors with which the words' relationships can be quantified reasonably. Nevertheless, the research on utilizing the

 Copyright ⓒ 2016 SERSC

word vectors in the specific NLP problems has not been fully carried out. This task is designed to further evaluate the quality of the word vectors. As a simple solution to the document representation problem, the vectors of the words appearing in the document are added to get the real-valued document vectors with the same dimension.

Two document sets are prepared to test our approach in this task. From each group in the 20-newsgroup dataset, we randomly select a pair of documents as the relevant documents to form a test set including 40 documents. Another test set is comprised of the irrelevant document pairs which are randomly selected from different news groups. Finally, the average cosine similarities are calculated on both the sets.

4.3. Results and Discussion

Table 1 lists the average cosine similarities calculated on the manually selected relevant word pairs and the randomly selected ones, whose word vectors are generated by the models adopting Huffman tree and LDA respectively. It can be seen that all the average similarities of the relevant word pairs are around 0.5. By contrast, the similarities of the randomly selected pairs are much lower, since random selection will always get the irrelevant pairs. This observation indicates that our LDA based architectures meet the basic requirement of the word embedding model.

Table 1. Average Cosine Similarities of Word Pairs

Model	Avg. Cosine Similarity	
	Relevant	Random
CBOW-Huffman	0.4812	0.0796
Skip-gram-Huffman	0.5093	0.0770
CBOW-LDA	0.5457	0.0631
Skip-gram-LDA	0.5595	0.0727

The results in Table 1 have suggested that comparing with the classic models with the Huffman tree vocabulary; the architectures adopting LDA (CBOW-LDA and Skip-gram-LDA) tend to generate the vectors with higher similarity for the given semantically relevant word pairs. It should be noted that the models for continuous word representation map the words to a new semantic feature space indeed, thus the average similarity of the highly relevant words is a factor to be taken when estimating the reasonableness of the semantic space. On the other hand, it is expected that the irrelevant word vectors can be separated in the space, and the results show that our method afford the characteristic.

The key of our approach is to encode words in a vocabulary with their topic distributions generated by LDA. It is necessary to investigate how the encoding is affected by different topic numbers. Figure 2 shows the relevant word pairs' average similarities when the number of topics changing from 10 to 50. The first observation is that the Skip-gram architecture performs better than CBOW, which can be attributed to the fact that the frequencies of the words are limited due to the scale of the training corpus, and the current empirical studies indicate that the skip-gram architecture will get better performance on infrequent words.

As shown in the figure above, both the models obtained the best performance when LDA provides the 40-dimensional topic distribution to each word for binary encoding. Generally, the topics of the topic model do not always match the real categories of the dataset. It is possible that the 40-dimensional topic distribution has offered plenty of prior semantic relevance information for word encoding, while avoiding the confusion brought by the redundant dimensions.

Copyright ⓒ 2016 SERSC

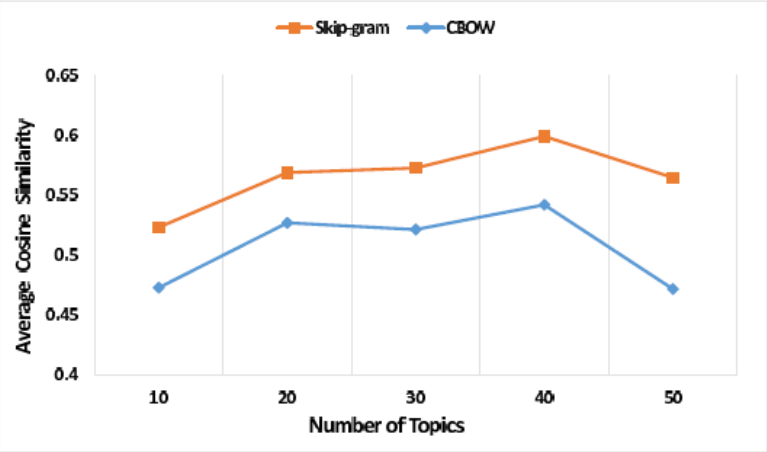

Figure 2. Average Similarities of the Word Vectors Generated by the Models with Different Dimensional LDA Outputs

The experimental results in Table 1 and Figure 2 have indicated that our topic distribution based word encoding approach is promising for improving the quality of the continuous word vectors. We attribute the improvement to the words' semantic relevance implicitly introduced by the topic model. As mentioned in Section 3, the word's probabilistic distribution gives a preferable way to preliminarily model the semantic relevance of words. Apparently, the similar distributions on each of the topics can suggest that the corresponding words are likely to be relevant, but even equivalent frequencies can not lead to this inference. Basically, our new word encoding approach does not change the main training process, thus the context oriented word semantic estimation is still maintained. In this case, the context information and the prior semantic knowledge are fused and improve the quality of the word vectors.

Table 2. Examples of the Word Pairs in Different Categories

	Science	**Politics**	**Sport**	**Computer**
CBOW-LDA CBOW-Huffman	artificial - intelligence artificial – associate	gun - handgun gun - guns	sport - championships sport - trek	ati - graphics ati - gup
CBOW-LDA CBOW-Huffman	chemistry - physics chemistry – paperback	rifle - caliber rifle - recruitment	hockey - ecac hockey - fans	ram - dram ram - meg
CBOW-LDA CBOW-Huffman	scientist - research scientist – associate	iraq - kuwait iraq - senseless	baseball - basketball baseball - fans	harddrive - hardisk harddrive – leebr

To further illustrate the performance of our approach, Table 2 lists some examples of the word pairs in different categories. Each example is produced by joining the top-ranked neighbor with the given query word, and the neighborhood is measured with the cosine similarities of the word vectors generated by the classic CBOW architecture and the one with LDA based word encoding respectively. It can be seen that most of the pairs generated by CBOW-LDA involve semantically relevant words, whose relationships

 Copyright ⓒ 2016 SERSC

seems interesting and reasonable, *e.g*., "*scientist - research*" and "*hockey - ecac*". In the results generated by the classic model, by contrast, the top-ranked neighbors of *hockey* and *baseball* are both *fans*, and such results are obtained based on the context relationship obviously. For some NLP tasks like text classification, discovering such relationships are more meaningful than the syntactic (*e.g*., "*apple - apples*") or the obvious semantic (*e.g*., "*France - Paris*") relationships.

Table 3. Average Cosine Similarities of the Document Pairs

Model	Avg. Cosine Similarity	
	Relevant	Random
CBOW-Huffman	0.6138	0.3223
Skip-gram-Huffman	0.6109	0.3095
CBOW-LDA	0.6473	0.2251
Skip-gram-LDA	0.6559	0.2208

The experimental results of quantifying the documents' relevance with the continuous document vectors are given in Table 3. As shown by the table, the vectors obtained by simply adding the vectors of the words can be used to quantify the relevance for the documents, for the obvious gap of the average cosine similarities between the relevant documents and the irrelevant ones can be seen. Our proposed CBOW-LDA and Skip-gram-LDA architectures have done better work than the models with Huffman tree, since they have provide more distinguishable word vectors. However, the similarity gap is not as large as that of the word pairs, which means the document representation needs further investigation.

5. Conclusions

In this paper, we proposed an approach to encode words based on their topic distributions in a hierarchical softmax procedure. The contributions of this paper can be summarized as follows: (1) Our approach improves the quality of word vectors via integrating the context information and the prior word relevance knowledge of a word learnt by topic modeling. (2) We proposed an effective method to represent documents via our word embedding results. Experimental results showed that document vectors are reasonable to quantify the relevance of documents.

There will be two directions of future work. First, we will investigate other methods to represent text fragments or documents via word vectors. Second, further studies will be conducted to introduce other kinds of prior knowledge to the continuous word representation model.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (61272383, 61300114 and 61572151).

References

[1] Y. Bengio, R. Ducharme, P. Vincent and C. Janvin, "A Neural Probabilistic Language Model", Journal Mach. Learn. Res., vol. 3, **(2003)**, pp. 1137–1155.
[2] T. Mikolov, Q. V. Le and I. Sutskever, "Exploiting Similarities among Languages for Machine Translation", CoRR, abs/1309.4168, **(2013)**.
[3] R. Collobert, J. Weston, L. Bottou, M. Karlen, L. Kavukcuoglu and P. Kuksa, "Natural Language Processing (Almost) from Scratch", Journal Mach. Learn. Res., vol. 12, **(2011)**, pp. 2493-2537.

Copyright ⓒ 2016 SERSC

[4] T. Mikolov, K. Chen, G. Corrado and J. Dean, "Efficient Estimation of Word Representations in Vector Space", CoRR, abs/1301.3781, **(2013)**.
[5] T. Mikolov, W. T. Yih and G. Zweig, "Linguistic Regularities in Continuous Space Word Representations", Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies Association for Computational Linguistics: Atlanta, Georgia, **(2013)**, pp. 746–751.
[6] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado and J. Dean, "Distributed Representations of Words and Phrases and their Compositionality", In Advances in Neural Information Processing Systems 26. Burges C, Bottou L, Welling M, Ghahramani Z, Weinberger K., Eds., **(2013)**, pp. 3111-3119.
[7] F. Morin and Y. Bengio, "Hierarchical probabilistic neural network language model", AISTATSar05, **(2005)**, pp. 246-252.
[8] A. Mnih and G. E. Hinton, "A Scalable Hierarchical Distributed Language Model", In Advances in Neural Information Processing Systems 21. Koller D, Schuurmans D, Bengio Y, Bottou L, Eds., **(2008)**, pp. 1081–1088.
[9] D. E. Rumelhart, G. E. Hinton and R. J. Williams, "Nature", **(1986)**, pp. 533–536.
[10] R. Collobert and J. A. Weston, "Unified Architecture for Natural Language Processing: Deep Neural Networks with Multitask Learning", Proceedings of the 25th International Conference on Machine Learning ACM: New York, NY, USA, ICML '08, **(2008)**, pp. 160-167.
[11] T. Mikolov, M. Karafiát, L. Burget, J. Cernocký and S. Khudanpur, "Recurrent neural network based language model. INTERSPEECH, **(2010)**, pp. 1045-1048.
[12] A. Frome, G. S. Corrado, J. Shlens, S. Bengio, J. Dean, M. A. Ranzato and T. Mikolov, "DeViSE: A Deep Visual-Semantic Embedding Model", In Advances in Neural Information Processing Systems 26 . Burges C, Bottou L, Welling M, Ghahramani Z, Weinberger K, Eds., **(2013)**, pp. 2121–2129.
[13] C. Junjun, Y. Linshen and L. Peng, "Image Clustering via combined visual and annotation information", Journal of Harbin University of Science and Technology, vol. 19, no. 2, **(2014)**, pp. 57-62.
[14] D. M. Blei, A. Y. Ng and M. I. Jordan, "Latent Dirichlet Allocation", Journal Mach. Learn. Res., vol. 3, **(2003)**, pp. 993-1022.

 Copyright ⓒ 2016 SERSC