Research and Construction of Word Vector Extension Structure Based on Context

Yiying Wang¹, Hongyun Ning²
¹ Tianjin Key Laboratory of Intelligence Computing & New Software Technology, Tianjin University of Technology, Tianjin, 300384, China
² Tianjin Key Laboratory of Intelligence Computing & New Software Technology, Tianjin University of Technology, Tianjin, 300384, China
*wang_yiying_olivia@163.com, nhyred@sina.com

Abstract. The CBOW (Continuous Bag-of-Words) model is a three-layer forward neural network that predicts the central word vector by the fixed-size window information. In fact, context has a very important role in understanding the meaning of words. However, the context information of the fixed window size is partial and it is not enough to represent the whole context. Due to the polysemy of Chinese words, the same word may have different semantics in different contexts, while the traditional CBOW method ignores the polysemy of words. Therefore, this paper proposes a context-based word vector extension structure for the above problems. The main contents are as follows: (1) Introducing the concept of context vector. The entire sentence in which the target word is located is represented by a vector. (2) Constructing the polysemy storage method of words, adding a contextual list to each word vector, so that the multiple semantics of the words can be effectively distinguished. (3) Based on the word vector extension structure of this paper, a new character vector generation model is proposed, and the effect of the new model and the traditional CBOW model in the news headline similarity sorting task is compared. The experimental results show that the new character vector generation model based on the extension structure of the word vector obtains better result.

1. Introduction

Text representation is the basic problem of natural language processing tasks. The original vector transformation method is one-hot, and its dimension lies on the size of the word dictionary. It has two problems. First, this representation has a very high dimension and great sparsity, which will lead to curse of dimensionality [1]. Second, this method cannot represent the complex semantic information of the language. In order to overcome these shortcomings, Bengio et al. proposed the concept of word vector, the word vector effectively avoids the dimensionality disaster and data sparsity, and it can calculate the semantic relevance between words.

In 2013, Mikolov[3] proposed the word vector generation model word2vec and model acceleration strategy[4] and it has been widely used. In word2vec model, the CBOW method predicts the target word through window words, the resulting vectors only contain the window-size information. In order to enrich the information of the word vector, sun et al.[5], li et al.[6] split all the characters into Chinese character component radical, puts forward the radical information as an additional semantics combined with words to express the rich character vector. Yu et al.[7] proposed a more detailed resolution, which breaks down Chinese characters into small modules, and combines the words,
characters and character parts together for joint learning. In terms of polysemy of characters and words, Chen et al.\cite{8} proposed to distinguish polysemy of characters according to the position of characters in different words. Hu et al.\cite{9} added part-of-speech information to obtain multiple semantic representation of polysemous words.

All of the above methods solve the problems of word vector information richness and word polysemy to different degrees, but they don't consider the context information on word vector generation. Therefore, this paper gives full consideration to the context on words, improves the traditional CBOW model and proposes a context-based word vector extension structure.

2. CBOW Principle and Existing Problems
The CBOW (Continuous Bag-of-Words) model was proposed by Mikolov et al.\cite{3} in 2013. Its structure is a three-layer feedforward neural network consisting of the inputLayer, hiddenLayer and outputLayer. Figure 1 shows the model of CBOW.

Figure1. Traditional BOW model

(1)The input layer of CBOW consists of C vectors, C is the number of window words, and the vectors are one-hot vectors. As shown in the figure above, assuming the window size is 2, when the prediction center word vector $X_t$ is predicted, the 4 $V$-dimensional one-hot vectors of its window words $X_{t-2}, X_{t-1}, X_{t+1}, X_{t+2}$ compose the input of this prediction.

(2)The hiddenLayer contains the vector $h_t$, which is the result of the average of adding the input vector and the input weight matrix multiple product. The input weight matrix $W_I$ is also called the "lookup-table", which is a $V \times N$ dimensional matrix, $V$ is the length of the vocabulary, and $N$ is the dimension of the word vector. Four input vectors are respectively multiplied by the input weight matrix $W_I$ to obtain four $1 \times N$-dimensional vectors, and add the four vectors, then divide the result by four to get the average, that is the hiddenLayer vector $h_t$.

(3)The outputLayer is the predicted value of the target word vector. The hidden layer vector $h_t$ is multiplied by the output weight matrix $W_O$ to obtain the predicted value $X_t$. $X_t$ is compared with the one-hot vector of the central word and the difference is calculated. Then update the input weight matrix $W_I$ and the output weight matrix $W_O$ by back propagation. Finally, the input weight matrix $W_I$ will contain the word vectors of all words in the corpus.

According to the analysis of the above principle, CBOW only uses the window words of the target word when generating the target word vector. However, when we understand the meaning of a specific word, we need to take the whole context of the word into account. In addition, due to the polysemy of Chinese words, the same word may correspond to different semantics. However, the traditional CBOW model doesn't consider the polysemy of the word, and ignores the differentiation of multiple semantics of the word.

3. Context-based Word Vector Extension Structure

3.1. Definition of context vector
The sentence in which the target word is located is represented by a vector, which becomes the context vector of the word in the current context. Because the context has a very important role in the
understanding of a word, the traditional CBOW model uses the partial context information is one-sided, not enough to represent the semantics of the whole sentence. Therefore, the whole sentence is represented by a context vector. Among them, the calculation of the context vector is obtained by adding and averaging the word vectors of all the words in the sentence. For example, the sentence $S$ is composed of $k$ words, and $X_1, X_2, \ldots, X_k$ represent the word vectors of the $k$ words, and then the context vector of the word $X_i$, $i \in \{1, k\}$ is
\[
S_{\alpha} = \frac{X_i + X_{i+1} + \ldots + X_k}{k}.
\]

3.2. Definition of The Word Vector Extension Structure Based on Context Vector

The word vector extension structure refers to adding a context list to each word vector based on the original word vector, so that multiple semantics of the word can be stored in the context vectors of the word vector extension structure. Due to the polysemy of Chinese words, words have different semantics in different contexts, and the expressions of different semantics should be distinguished. However, since the traditional CBOW method has no repetition when establishing the vocabulary, each word in the training result only corresponds to one vector expression, thus ignoring the polysemy of the word. Therefore, the polysemy storage method of the words constructed in this paper stores different contexts of words. For example, the extended structure of the word vector $X_i$ can be expressed as $(X_i, S_{X_{i1}}, S_{X_{i2}}, \ldots, S_{X_{in}})$, where $S_{X_{ij}}$, $j \in \{1, n\}$ represents the $j$th context vector of word vector $X_i$.

3.3. The Construction of The Extension Structure

3.3.1 Pseudocode

Function1:
CreateYJ()
    for each sentence
        for each word
            Calculate context vector $S_{X_{ij}}$
            updateYJ()
    Function2:
updateYJ()
    for calculate similarities between $S_{X_{ij}}$ and existed vectors
    if the similarity $> \Delta$
        insert $S_{X_{ij}}$
    else $S_{X_{ij}} = \frac{1}{2}(S_{X_{ij}} + \text{the most similar vector})$
    insert $S_{X_{ij}}$

It can be known from the above pseudo code that in order to construct the extended structure of the word vector, it is first necessary to define the context vector constructing function CreateYJ(). For each sentence in the corpus, traverse each word in the sentence, and the word vector of each word are obtained by CBOW. According to the word vectors, calculate the context vectors of each word in different sentences, and then calls the context vector update function updateYJ(). In the context vector update function, for the new context vector $S_{X_{ij}}$ of the word $X_i$, calculate its similarity with other existing context vectors of $X_i$, and compare the similarity value with the threshold $\Delta$, if the similarity is greater than the threshold, add the new context vector to the extended structure of the current word, if the similarity with an existing context vector is less than the threshold, the average the value of the context vector $S_{X_{ij}}$ and the less similarity, use the average value to update the original context vector.
3.3.2 The Storage of The Extension Structure

![Diagram of the Storage of The Word Vector Extension Structure](image)

The storage diagram of the word vector extension structure is shown in Figure 2. It is a chain storage table composed of the word name wordi, the word vector Xi and its corresponding context vector SXij. The word vector Xi is obtained by CBOW, and the context vector SXij is obtained by the CreateYJ() function in the above pseudo code, that is, the sentence vector of each sentence is calculated according to the word vectors obtained by CBOW, and the update of the context vector of each word is called the context update function updateYJ(), which updates the context list by comparing the similarity between the new context vector and the existing context vectors.

3.4. Improved CBOW model and its application

3.4.1 Improved CBOW model

Based on the above ideas, the improved CBOW model after adding the context-based word vector extension structure is as follows:

![Diagram of Improved Word Vector Generation Model](image)

As shown in figure 3, in the improved context-based word vector generation model, a dictionary is first constructed after the word segmentation of the corpus, a prediction vector of the central word is obtained through the window words, and then the difference between the predicted value and the target value are calculated. Update the input weight matrix and the output weight matrix by back propagation to obtain the word vectors of the V words in the dictionary. What differ from the traditional CBOW model is, each time the central word is predicted according to the input context, the output layer includes not only the prediction vector Xt of the central word but also the context vector SXt of the sentence in which the word is located.

3.4.2 Application

In the process of learning Chinese characters, words composed of the characters are often used to help understand the meaning of the word. In the case of polysemy, contextual information is usually taken into account to judge the semantic of the character in the current context. Based on this idea, on the basis of the extended structure of word vector proposed in this paper, a new context-based character vector generation method can be obtained. The new method extracts all words containing the target character, calculates the similarity between the context vector in the extended structure and the current character, and selects the word that is most similar to the current context. The calculation combined with the attention mechanism, assigns weights to the words participating in the
calculation according to the character and word context similarity, and weights the words and their weights to obtain the final character vector.

4. Experiment and Analysis

4.1. Experiment Process

This paper uses the extrinsic evaluation task, constructs the corpus, and obtains all the character vectors to represent the news titles. After calculating the similarities, the 10 news titles with the highest similarity to the specified title are found and printed. This paper crawls 300 titles from Sina News education and examination column, and cleans the original data for this experiment.

① Firstly, use the CBOW method, construct the character vector table and establish the similarity search algorithm based on the table. That is, for each title in the corpus, the vector of each character in the title is found from the character vector table. Calculate their similarity with the title X, respectively, then take out the top 10 news titles most similar to X and print.

② Then use the context and words oriented character vector generation method to obtain the word vector extended context table. Based on the table, establish the character vector calculation function to obtain the character vectors, and then call the above function “similarity search algorithm”, extract the first 10 are extracted. X most similar news headlines and output.

4.2. Results and Analysis

Let the title X = "高考如何才能成功逆袭".

The top 10 news titles most similar to X obtained by the traditional CBOW are shown as figure 4. And the top 10 news titles obtained by the new method are shown as figure 5.

Contrasting the experimental results 1 and 2, it can be seen that the result 2 is better than the result 1. As shown in the experimental results, the context is more concentrated in the field of examination. Compared with result 1, the title classification is more obvious. This is because the character vectors in method 1 only cannot fully represent the semantic information of a character. However, the character vectors in method 2 is based on words and context, and it can distinguish the different semantics of the words according to the context information. When generating the target character vector, the context of the character is fully considered, and the most similar words will be chosen to construct the final character vector, which can more accurately describe the semantics of the character in a specific context. Therefore, character vectors that take into account words and context information perform better in this task.

References

[1] Bellman R E. Dynamic Programming[M]. [S.l.]: Dover Publications, Incorporated, 2003.
[2] Yoshua B, Réjean D, Pascal V, et al. A Neural Probabilistic Language Model[J]. Journal of Machine Learning Research, 2003, 3: 1137–1155.

[3] Mikolov T , Chen K , Corrado G , et al. Efficient Estimation of Word Representations in Vector Space[J]. Computer Science, 2013.

[4] Mikolov T , Sutskever I , Chen K , et al. Distributed Representations of Words and Phrases and their Compositionality[J]. Advances in Neural Information Processing Systems, 2013, 26:3111-3119.

[5] Sun Y , Lin L , Tang D , et al. Radical-Enhanced Chinese Character Embedding[J]. Lecture Notes in Computer Science, 2014, 8835:279-286.

[6] Li Y , Li W , Sun F , et al. Component-Enhanced Chinese Character Embeddings[J]. Computer Science, 2015.

[7] Yu J , Jian X , Xin H , et al. Joint Embeddings of Chinese Words, Characters, and Fine-grained Subcharacter Components[C]// Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing. 2017.

[8] Chen X , Xu L , Liu Z , et al. Joint learning of character and word embeddings[C]// International Conference on Artificial Intelligence. AAAI Press, 2015.

[9] Hu Baotian. Text representation based on deep neural network and its application [D]. Harbin Institute of Technology, 2016.