Proceedings of the 3rd International Conference on Signal Processing and Machine Learning
DOI: 10.54254/2755-2721/6/20230719

A comparison of multiple word embeddings and performance analysis

Jinghua Wang[1] and Zhongtian Lin[2]

[1]International School, Beijing University of Posts and Telecommunications, Beijing, 100089, China
[2]Computer and Big Data Institute, Fuzhou University, Fuzhou, 350001, China

wangjinghua@bupt.edu.cn

Abstract. Based on IMDB data sets for sentiment analysis, this research evaluates the performance of two types of neural networks, namely FNN and BERT. According to the experimental findings, both varieties of neural networks performed well (over 80%) on IMDB data sets. BERT performs the best of them all. The version of FNN improves when the number of layers is increased. The causes behind BERT's outstanding performance are then further examined in this research. In this article, it is hypothesized that a few factors, such as feature extraction capability and classification capability, account for BERT's exceptional performance. The results show that for the two coupling variables of BERT assumed in this paper, including feature extraction and classification ability, BERT has obvious advantages in feature extracting, and its classification ability has certain advantages over SVM and MLP. The deep learning model has achieved excellent performance in the sentiment classification task. The next step of this paper will focus on the robustness of the BERT model in sentiment analysis.

Keywords: deep learning, sentiment analysis, emotion recognition, BERT, FNN, IMDB.

1. Introduction

Text classification is one of the very basic tasks of natural language processing. In the past few years, deep learning models have become one of the first approaches to solve the task of text classification. Sentiment analysis is a typical case of text classification. Emotion analysis refers to the analysis of people's emotions and attitudes in their opinions, so as to divide the opinions expressed by people into positive emotions and negative emotions [1]. In recent years, deep learning models have achieved good classification results on sentiment analysis problems. Common deep learning models for sentimental analysis problems include Recurrent Neural Network, Convolutional Neural Network, and BERT.

BERT (Bidirectional Encoder Representation from Transformer) can greatly improve the prediction accuracy of the model. BERT is a model of linguistic representation based on prior training. The previous pre-trained language models are often limited by the structure of a single language model, which can only capture the context information in a certain direction. Therefore, the accuracy of the model is not good. The emergence of BERT breaks this limitation, which uses deeper bidirectional converter components to build the whole model, and finally forms a deep language representation that can fuse bidirectional information, which vastly enhances the prediction accuracy of the model. BERT model is widely used in text classification, question answering system, reading comprehension and other fields. It contains an attention mechanism and a bidirectional converter component and is one of the best models for NLP tasks today [2].

Compared with ordinary FNN, BERT has a more vital feature extraction ability, and it can also help the downstream sentiment analysis task to be complete with high quality. Previous studies have shown BERT's excellent performance in emotion analysis tasks, the results of the continuous evolution of model technology are inspiring, however, it is necessary to analyze further the reasons for BERT's outstanding performance. It is also helpful to find out the direction of further research and help researchers to achieve the balance between performance and cost.

Therefore, this paper mainly analyzes the performance of FNN and BERT models on sentiment analysis tasks based on the IMDB data set, examines the feature extraction ability of the two models on text information through experimental results, and verifies the influence of SVM and MLP classifiers on sentiment analysis task through comparative experiments.

2. Methods

Here we present key models and features used in this research.

2.1. Dataset

In this article, we selected IMDB data set, which is specially designed for emotional analysis. The data set contains a total of 50000 film reviews of emotionally polarized comments, of which 25000 labeled and 25000 unlabeled. Labeled comments will be used for training model and unlabeled comments will be used for testing. Training data set and testing data set each contains half positive and half negative comments. Emotional tendencies are labeled as train_labels and test_labels, lists of zeros and ones, with 0 being a negative emotion and 1 being a positive emotion.. The IMDB dataset has been integrated into the Keras library and has been preprocessed. It can be easily called

2.2. FNN

The Feedforward Neural Network (FNN) is among the most widely used and well- developed artificial neural networks. Neurons are arranged in layers, and individual neuron is only connected to the neuron from the previous or next layer, accept the output passed by the upper level and pass its own to the lower layer. There is absence of feedback among adjacent layers. Common feedforward neural networks include the perceptron network and the BP network. The perceptron network is the most basic feedforward neural network, aiming to solve separating hyper-planes that apply linear division to the labeled training data. Naturally, the loss function based on miss-classification that judges the performance of current weight emerges, while the gradient descending algorithm is utilized to conversely minimize the loss function to obtain an accurate perceptron model. Perceptron network mainly used for pattern classification and can also be used in learning control and multi-mode control based on pattern classification. The sensor network can be divided into single-layer sensor network and multi-layer sensor network. BP network refers to the feedforward network with a backpropagation learning algorithm. The BP network finally converges to the optimal weight value by continuously updating weight through the loss function. Unlike perceptron, the neuron transformation function of BP network assumes a sigmoid function, so it can realize any non-linear mapping from input to output [4].

2.3. BERT

This section will provide a thorough introduction to the logical framework of the BERT model to help readers comprehend its fundamental operating principle. The Transformer framework serves as the foundation for the BERT universal language model, which can produce excellent representation vectors for texts. Each encoder in the BERT system is referred to as a Transformer block. The embedded vector X is created by calculating the input text's embedding layer, and it is then entered into the Transformer block for use. After repeated iterative calculations, the representation vector of the BERT model to the text is formed using the output vector of the first Transformer block as input into the second Transformer block [5].

The BERT model initially creates three N-dimensional vectors Q, K, and V for each self-Attention mechanism for each word in the text (that is, the dimensions of the final representation vector). The attention score is computed using these three vectors. This score value is used by the self-attention mechanism to compute the level of attention paid by other words in the text to the current term and create a representation vector with context. The weight matrices , W^Q,W^K, and W^V are multiplied by the embedded vectors x of the relevant words to produce the q, k, and v vectors, respectively. Before the BERT model's pre-training, W^Q,W^K, and W^V will be randomly initialized, and they will be updated continually while the BERT model is being trained [6].

A multi-attention mechanism is enhanced from the self-attention mechanism is added to the BERT model for parallel computing, which can increase the model's effectiveness and allow it to learn various sub-semantic spaces for words. In order to provide numerous output vectors with richer context information, the mechanism will build up multiple groups of W^Q,W^K, and W^V weight matrices. It will then utilize these groups of Q, K, and V vectors to compute the self-attention of words in the text at the same time. The output Z of the multi-head attention mechanism is then created by combining these output vectors into a matrix and performing dimension reduction.

The feedforward neural network in the Transformer block first calculates the output vector Z of the multi-head attention mechanism before generating the N-dimensional output vector R. The i+1 layer Transformer block will utilize the output r_i of the i+1 layer Transformer block as its input and carry out the same multi-head attention and feedforward neural network calculations. The final output r_k of the BERT model for the previous words may be obtained after this procedure iterates for k times [7].

The N-dimensional representation vector r_k corresponding to each word in the text may be derived when the computation is completed for each word in the text. In order to represent the text, BERT combines these vectors into a matrix with n dimensions and a sentence length matrix, where sentence length is the length of the text. BERT inserts a meaningless symbol [CLS] in front of each text in order to get around the classification layer's inability to accept a matrix as input. The word is then represented by the corresponding representation vector of this symbol. When in use, the BERT output matrix's first row vector only must be extracted and fed to the classification layer for training.

2.4. TF-IDF

In the vector spatial model, Term Frequency-Inverse Document Frequency (TF-IDF) is a massively used weighting methodology. TF-IDF algorithm is based on this assumption: for the optimal feature words, these feature words appear in a large number in a class or part of documents, but rarely or not in other documents. Therefore, that same text can be divided by using the term frequency TF.

In addition, considering the significance of a feature word in the text, it is assumed that the higher the frequency of feature words in a text, the more important the feature words are, so the inverse document frequency (IDF) is introduced. The product of the term frequency (TF) and the IDF is taken as the value measure of the vector space model. However, in essence, IDF is a noise-offsetting weighting method to avoid noise data. At the same time, it is obviously not completely correct to think that less text is important, and more text is unimportant. Therefore, the accuracy of this algorithm is not high.

TF-IDF calculation can be described as formula (1):

$$a_{ij} = tf_{ij} \times log\frac{N}{n_j} \quad (1)$$

In formula (1), tf_{ij} represents the frequency of term j in document i, and N represents the total amount of documents within the data set, and n_j represents the times of documents in which term j appears [8].

2.5. SVM

Support Vector Machine is a mechanism of machine learning that was initially introduced by Vapnik et al. and is based on statistical learning theory. The best classification hyperplane must be found in order to accurately distinguish the two different types of data points and maximize the margin of classification interval. Making a restricted optimization problem is the solution. an issue with limited quadratic programming that must be solved in order to produce a classifier. Given a sample set(x_i, y_j) $x_i \in R^n, y_i \in \{1, -1\}$, the expression y_i denotes the category to which x_i belongs. When $y_i = 1$, anything is said to belong to the positive class, while when$y_i =- 1$, it is said to belong to the negative class. There is a classification hyperplane: $w \cdot x + b = 0$ if the sample points are linearly separable. For each sample point that has been separated, $y_i[(w \cdot x_i) + b] >= 1$, where w is the classification hyperplane's n-dimensional normal vector and b is the offset Because if the classification interval is the biggest, $Margin = 2/\| w \|$, $min \| w \|^2/2$will be solved [9]

Lagrange multiplier is presented for use in this optimization issue. The general form of the problem description is indicated in formula when converted to its dual problem (2):

$$max \sum_i \alpha_i - \frac{1}{2}\sum_{i,j} \alpha_i \alpha_j y_i y_j K(x_i, x_j) \tag{2}$$

The constraints are $0 \leq \alpha_i \leq c$ and $\sum_i \alpha_i y_i = 0$ in nature. In formula (4), the categorization function is displayed:

$$f(x) = sgn[\sum_i \alpha_i y_i (x_i, x) + b] \tag{3}$$

where c is the error penalty component, which regulates the severity of the penalty for selecting the incorrect sample. In order to make the nonlinearly separable sample points linearly separable in high-dimensional space, the kernel function $K(x_i, x)$ translates all sample points to high-dimensional space.

Text classification based on SVM includes two parts: training and classification. In the training procedure, the training text is generated into a document vector by referring to the feature dictionary. Input SVM learner to obtain classification model; In the classification stage, the document vector of the text to be classified is input into the classification model to get the classification result [10]

2.6. MLP

Multilayer perceptron is an evolutionary version of perceptron. Its characteristic is that it can process nonlinear data with more high-dimension parameters and achieve higher accuracy, which is supported by multiple interconnected neuron layers. It is composed of an input layer, output layer and hidden layers between them, in which the number of hidden layers should be determined by specific classification tasks. The connection between input layer and hidden layer utilizes as a fully connected layer, and that connects hidden layer and output layer is functionalized as a classifier. In constructing the MLP model, weight W and bias B will be randomly initialized in advance, and then the perceptron will learn. The learning process is to make W and B constantly improve in the direction of reducing errors, that is, to minimize the loss of function value. Firstly, according to the input training data X, the forward input and output of all layers are calculated, then the error terms of the output layer are calculated, and the error terms are pushed backward layer by layer. Finally, the partial derivatives of the cost function about each weight and each bias are calculated, and then use gradient descent algorithm iterations to update the parameters. Using cost function by mean square error is simple and easy to understand, but it is prone to saturation and limited to the local machine. The improved cost function method is to use the cross entropy J(0), and the formula (4) is as follows：

$$J(\theta) =- \frac{1}{n}\sum_{k=1}^{n} [\hat{y}_k lny + (1 - \hat{y}_k) ln(1 - y)] \tag{3}$$

After "cross entropy" is adopted as the cost function, the partial derivative of the cost function about the weight value leaves the error term between the output and the label. If the error is more significant, the correction term will be more extensive, the parameter update will be faster, and the training speed will be faster [11].

3. Experimental results and analysis

The experiment of testing models consists of two parts. Firstly, the performance of FNN and BERT models is tested based on the IMDB dataset. The FNN includes three experimental settings with different depths: one-layer, two-layer, and three-layer. The experimental results can be seen in Table 1.

Table 1. Accuracy of multi-layer FNNs and BERT.

Model	Accuracy
FNN (one-layer)	81.31%
FNN (two-layer)	84.40%
FNN (three-layer)	87.66%
BERT	91.12%

As can be seen from Table 1, both types of neural networks have achieved good performance (over 80%) on IMDB data sets. Among them, BERT has the best performance, accuracy 9.81%, 6.72% and 3.46% higher than FNNs respectively. With the deepening of the number of layers, the version of FNN becomes better. This is in line with the expectation of this article. Generally speaking, the deeper the layers of the neural network model, the stronger its fitting ability to the distribution of data features. Although FNN may appear to be over fitted in multiple layers, it does not appear because layer didn’t exceed 3. BERT uses the most advanced multi-head attention mechanism not only present, which far exceeds FNN in model scale, but also pre-trains large-scale corpus. Therefore, the model performance is far superior to FNN.

Then, this paper further analyzes the reasons for BERT's excellent performance. In this paper, it is considered that the reason for BERT's outstanding performance is a couple of variables, including feature extraction ability and classification ability. Specifically, this paper uses BERT trained in Table 1 to generate word embedding contained in each sentence in the IMDB data set. And the average operation is performed on word embedding, which constitutes a sentence, as the vector of each sentence. Then, this paper selects SVM and MLP models to analyze the above sentence vectors. The experimental outcomes are presented in Table 2.

Table 2. Accuracy of BERT with different classifiers.

Sentence embedding	Classifier	Accuracy
BERT	SVM	90.23
BERT	MLP	90.01
TF-IDF	SVM	80.60

The experiment in Table 2 is divided into two parts. The first part uses two different classifiers, including SVM and MLP, to classify Bert Sentiment Embedding. The training results show that the accuracy of SVM and MLP based on Bert Sentiment Embedding is over 90%. Compared with the original BERT, the performance is reduced by 0.89% and 1.22%, respectively. This paper holds that the experimental results show BERT's powerful feature extraction ability. Then, TF-IDF is used to generate sentence embedding, and SVM is used to classify it as the control group. The results show that the quality of Sentiment Embedding caused by TF-IDF is far lower than that of BERT, and the performance of TF-IDF+SVM drops to 0.963% compared with BERT+SVM. The above results prove that for the two coupled variables of BERT assumed in this paper, including feature extraction ability and classification ability, BERT's feature extraction ability has apparent advantages, and its classification ability has certain benefits compared with SVM and MLP.

4. Conclusion

This paper tests the performance of two types of neural networks, including FNN and BERT, based on IMDB data sets for sentiment analysis. The experimental results show that both types of neural networks have achieved good performance (over 80%) on IMDB data sets. Among them, BERT has the best performance. With the deepening of the number of layers, the version of FNN becomes better. Then, this paper further analyses the reasons for BERT's excellent performance. In this paper, it is considered that the reason for BERT's outstanding performance is a couple of variables, including feature extraction ability and classification ability. The results show that for the two coupled variables of BERT assumed in this paper, including feature extraction ability and classification ability, BERT's feature extraction ability has apparent advantages, and its classification ability has certain benefits compared with SVM and MLP. The deep learning model has achieved excellent performance in emotion classification tasks. The next step of this paper will focus on the robustness of the BERT model in emotion analysis.

References

[1] Hirschberg J, Manning C D. Advances in natural language processing[J]. Science, 2015, 349(6245): 261-266.

[2] Devlin J, Chang M W, Lee K, et al. Bert: Pre-training of deep bidirectional transformers for language understanding[J]. arXiv preprint arXiv:1810.04805, 2018.

[3] Svozil D, Kvasnicka V, Pospichal J. Introduction to multi-layer feed-forward neural networks[J]. Chemometrics and intelligent laboratory systems, 1997, 39(1): 43-62.

[4] Wolf T, Debut L, Sanh V, et al. Transformers: State-of-the-art natural language processing[C]//Proceedings of the 2020 conference on empirical methods in natural language processing: system demonstrations. 2020: 38-45.

[5] Sun C, Qiu X, Xu Y, et al. How to fine-tune bert for text classification?[C]//China national conference on Chinese computational linguistics. Springer, Cham, 2019: 194-206.

[6] Voita E, Talbot D, Moiseev F, et al. Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned[J]. arXiv preprint arXiv:1905.09418, 2019.

[7] Ramos J. Using tf-idf to determine word relevance in document queries[C]//Proceedings of the first instructional conference on machine learning. 2003, 242(1): 29-48.

[8] Noble W S. What is a support vector machine?[J]. Nature biotechnology, 2006, 24(12): 1565-1567.

[9] Tong S, Koller D. Support vector machine active learning with applications to text classification[J]. Journal of machine learning research, 2001, 2(Nov): 45-66.

[10] Riedmiller M. Advanced supervised learning in multi-layer perceptrons—from backpropagation to adaptive learning algorithms[J]. Computer Standards & Interfaces, 1994, 16(3): 265-278.