Proceedings of the 3rd International Conference on Signal Processing and Machine Learning
DOI: 10.54254/2755-2721/5/20230633

Performance analysis of sentiment classification based neural network

Jingyi Wang[1,3,†], Ruijie Xu[2,†]

[1]Queen's University, 99 University Ave, Kingston, ON Canada K7L 3N6, Canada
[2]University of the Fraser Valley, 33844 King Road, Abbotsford, BC, Canada V2S 7M8, Canada

[3]18jw111@queenu.ca
[†]These authors contributed equally

Abstract. Deep learning has more significant advantages for word embedding technology than sentiment analysis. This paper studies the application of deep learning on the word embedding problem in context, mainly discusses the RNN model with Word2Vec and without Word2Vec, then compares and analyzes their performance in the experiment, mainly evaluating the accuracy and test loss of seven models. This paper compares and illustrates the model which gets the different results in experiments, complementing the model and re-running the model, and analyzing the reasons for the difference in the performance of each model. The seven models are a single-layer neural network, multiple-layer (two and three) feed-forward neural networks, Convolutional Neural Network (CNN)- A feedforward neural network, which consists of single or multiple convolutional layers, pooling layers, and a fully connected layer on top, so this model is good at image processing. Long Short Term Memory (LSTM)- A temporal recurrent neural network, the advantage of the model is it could solve the gradient disappearance and explosion problem when it handles the long-sequence problem. Bi-directional Long Short Term Memory (Bi-LSTM)-Composed of forwarding LSTM and backward LSTM, it is very common for sequence labelling tasks that are related to the top and bottom, which are often used to model context information in NLP. Bi-directional Encoder Representation from Transformers (BERT)-A bidirectional language model. Finally, this paper analyses and evaluates these models with a specific illustration and research.

Keywords: Deep Learning, Sentiment classification, Classification, Word embedding, Machine Learning.

1. Introduction

The sentimental classification is a task which could output the conclusion depending on the context, the text could be a comment on a movie, a review of the production, or an attitude of the event. The output of the sentimental classification is usually divided into positive or negative. Because there are never like such a massive amount of opinion data has been stored in the digital form in human history, the beginning and rapid development of this field are consistent with the development of social media, such as comments, forums, blogs, Weibo, Twitter [1]. Sentiment classification is an essential task of natural language processing [2]. The term "sentimental classification" refers to the process of dividing a text

DOI: 10.54254/2755-2721/5/20230633

into two or more categories based on the meaning and emotional content it conveys. Tendency analysis is the process of categorizing the author's emotional tendency, viewpoint, or attitude. In e-commerce, sentiment classification can help businesses improve service and user experience by identifying user sentiment tendencies. Evaluation of ranking, sentiment analysis in customer service conversations, and sentiment classification evaluation are examples of application scenarios.

The current research focuses on three granularity levels of sentiment analysis: level of the document, statement, and aspect. Document-level sentiment classification divides documents with distinct opinions, such as movie comments, into two contrary opinions such as likes or dislikes [3]. It regards the entire document as the primary unit of information. It assumes that the document has a clear point of view, including the point of view of a single entity (for example, a specific mobile phone model). Aspect sentiment classification, entity extraction, and aspect extraction are just a few of the subtasks that make up aspect-based sentiment analysis. Statement-level sentiment classification classifies individual statements in a document. Based on the degree of dependence on labelled data, the existing emotion classification techniques are divided into supervised, semi-supervised, and unsupervised learning. Among them, the supervised learning method has the best performance at present. Supervision methods, including traditional machine learning methods, such as SVM or decision tree methods, have the advantage of simplicity. However, the disadvantage is that feature engineering is needed to extract discrete text features, such as n-gram. Moreover, deep learning methods have attracted attention in recent years, such as text-CNN, text-RNN, or more complex network structures. Compared with the traditional machine learning method, the advantage of the deep learning method lies in its powerful feature extraction ability. In addition, word embedding technology significantly improves sentence classification tasks' performance [4].

There are some words that have similar usage but it could express contrary emotions (like "bad" or "good") that could reflected to similar vectors in the embedding space, making it difficult to learn word embedding in context using conventional word methods like Skip-gram or CBOW for sentiment analysis. As a result, researchers also proposed embedding techniques for emotion-coding words [5]. Another paper, for instance, provides a model named emotional word embedding (EWE) for sentiment analysis. The experimental results recorded from the three real-world data sets could prove that the given EWE model performs better than other cutting-edge models for text sentiment prediction, and text similarity calculation [6]. For SA of unstructured data, a text normalization using a deep convolutional character level embedding (Conv-char-Emb) neural network model was proposed in another report. An efficient and effective method for SA that makes use of less learnable parameters in feature representation is character-based embedding in a CNN [7]. In addition, the author of one paper constructs a framework to prove the detail temporal dynamics of the embedding aid in quantifying shifts in attitudes and stereotypes toward women and ethnic minorities in the United States during the 20th and 21st centuries. To demonstrate that differences in the embedding closely follow shifts in demographic and occupation over time, we combine word embeddings trained on 100 years of text data with the US Census [8].

This paper explores and finds the RNN model with Word2Vec and without Word2Vec and analyses it in this process [9]. Then, seven models are evaluated: one-layer feed-forward, two-layer feed-forward, three-layer feed-forward neural network, CNN, LSTM, Bi-LSTM, and BERT. Experimental results show that introducing Word2vec into RNN can significantly improve the model's performance, proving word embedding technology's effectiveness [10]. Besides, the Loss of the One-layer feed-forward neural network is 0.562, the test set accuracy is 78.01%, the Loss of the Two-layer feed-forward neural network is 0.541, the accuracy is 80.02%. Three-layer feed-forward neural network is 80.02%, the Loss of model is 0.546, and the accuracy is 80.32%. Explained The impact of the different network layers on the model results. In the comparison of CNN, LSTM, BILSTM, and Bert, the performance of CNN is the worst. However, it is significantly higher than the three-layer fully linked neural network. The reason the test result of the Bi-LSTM model is better than LSTM is that it considers the sequence features of sentences more comprehensively. The more advanced multi-head attention mechanism and large-scale pre-training strategy, BERT performs best among the four models [11].

DOI: 10.54254/2755-2721/5/20230633

2. Methods

2.1. Word2vec

Before the word2vec, it should know about the one-hot encoding first. For example, there are six words: "I like to read the book. Each word could represent by a one-dimensional vector. To more precise, the "I" could be the [1,0,0,0,0,0]," like" [0,1,0,0,0,0] and so on. Only one position will be 1. Other positions will be 0. This kind of way to represent words is low efficiency because if a paper consists of 10000 different words, there are 10000 vectors to represent these words. Thus, there is another way to represent the word more efficiently, which is called word embedding. We can represent the different words with word embedding, it could give a specific dimension vector to represent the words, and the value of the dimension may be different depending on the index. The Word2vec devise by Google in 2013 to better represent word vectors. Word2vec applies skip-gram and Continuous Bag-of-Words (CBOW) generates word embedding. The skip-gram algorithm is predicted to the few nest words depending on the central word. The input layer is a target word, and the algorithm will output the specific size of words. The CBOW algorithm is predicted to the central word depending on the few nest words.

2.2. Forward neural network

Humanity's brains inspire the neural networks, and scientists create the neural network as they simulate the neural systems in the human brain. Humans' brain has many neurons to transmit electric signs. Thus, there are many neurons in the forward neural network, and each neuron could receive and process the input and then output. The neural network is very important in fields related the deep learning. The feed-forward neural network (FNN) transit the data in one direction, and they will not receive any feedback from the next layer. One-layer feed-forward is that a neural network only has an input layer and an output layer. The single-layer FNN network is commonly seen in its most straightforward as a single-layer perceptron.

$$Z = (x_1 * \omega_1 + x_2 * \omega_2 + \cdots + x_t * w_t) + b \quad (1)$$

In this model, each neuron's input x_t will be multiplied by weights, then sum it, and add the b(bias). The bias means that there are some inevitable errors in the prediction when we want to predict some results. The output will be processed by activating function. This kind of model is always applied to classification tasks. Unlike the one-layer FNN network, the multiple-layer FNN network has the hidden layers that will be more than one layer. The hidden layer is between with input layer and the output layer. The benefit of the hidden layer is that the input data can be processed to be more complex or more dimension data, which could get more prominent performance. Usually, the model results will be more accurate if we use more hidden layers. This paper only displays three types of the FNN.

2.3. Convolutional neural network

If the input is the image, the full connection will be too complex and low efficiency when we convert the image to vectors. Therefore, The CNN model consists of multiple-layer feed-forward neural networks, including the input layer, hidden layers, and output layers. The biggest different with the forward neural network is that the input is three-dimensional (height, width, depth). The depth is not the number of the layer of the neural network, but it is the third dimension of the activation. However, also, there can separate into three parts: convolution, pooling, and fully connected. More specifically, the convolution and pooling layer will extract the image features if an image is inputted to the model. The pooling layer could classify max pooling and average pooling. The max pooling will extract the biggest value from the upper layer, and the average will calculate the average from the upper layer. The last layer will reflect those features in the output, such as the classification task. This kind of model is highly effective in image processing.

2.4. Recurrent Neural Network

DOI: 10.54254/2755-2721/5/20230633

The recurrent neural network (RNN) also belongs to one type of the neural network. It not only could connect to the next neuron but also could connect to the last neuron, then generate a cycle. The RNN model could receive feedback from the subsequent neurons. Thus, it could adjust the parameter better. The LSTM is one kind of RNN, and it receives feedback from the next layer. The standard LSTM consists of a cell and serval gates: forget gate, output gate and input gate. The cell will store the information that crosses the whole neural network. Besides, the other gates will decide whether the information needs to be stored in the cell. For example, the forget gate will get output according to the processing. If the output is 0, then the cell will not store it. On the contrary, if the output is 1, the cell will store it.

$$f_t = sigmoid(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{2}$$

f_t is the forget gate, $W_f$is currently weight, the h_{t-1} is the last layer feedback, x_t is the input, b_f is the bias. The Bi-LSTM model is still a kind of RNN model and could have two directions of data flowing. One data flow from the first to the end, and another flows from the end to the first. In this way, the RNN could store all information not only from the pre-neurons but also it could store the information from the later neurons. The dual-date flowing benefit is that the cell could store more comprehensive information.

2.5. BERT

This model is based on a Transformer and is mainly used for NLP prediction. The Transformer includes an encoder that can read the input and a decoder that produces a prediction for the output. The BERT model applies the Masked language model to produce the representation of the sentences. Meanwhile, the Transformer is not a model that reads the context from the first to the end. It could read from two sides. Therefore, this mechanism allows the model to understand the context more comprehensively. The BERT model is used to produce a lot of presentations of sentences, and then users can use the presentation in other NLP missions. We call this process pre-train. To be specific, if users train the model in mission A, then users want to apply the model in mission B, but users cannot obtain enough number of sample for B. In this situation, the model that pre-trains with A will be better than the model that only trains with B.

3. Experimental results and analysis

3.1. Data description

The IMDB dataset consists of 50000 comments, and each comment contains obviously sentiment. The positive sentiment occupies 50% of comments, and the negative sentiment occupies 50% of comments too. The data set was divided into two-part, the train part and the test part, and each part had 25000 samples. Each dataset is consists of a gzipped, and tab-delimited value (TSV) format file in the UTF-8 character set. The first line of each file has a title that describes the content in each column. " \N" is used to indicate that the title/name is missing or null. In the experiment of this paper, we use the train part to train the different models then use the test part to test the models.

3.2. Results and analysis

Table 1. The performance of RNN (Epoch=100, batch_size = 64).

Model Name	Test Loss	Test Acc
RNN (basic)	0.706	48.45%
RNN (with word2vec)	0.67	56.50%

As shown in Table 1, using word2vec as word embedding to initialize the RNN model has achieved significant performance improvement (+8%). This paper holds that the experimental results show the importance of word embedding technology for natural language processing. The large-scale pre-training model represented by BERT currently adopts more advanced training skills, thus obtaining word

DOI: 10.54254/2755-2721/5/20230633

embedding with contextual semantic information. This shows good migration in many downstream tasks. In addition, comparative learning has recently received attention in expressive learning. In the future work of this paper, more detailed experiments will be set up to verify the relationship between variables such as training methods, corpus size, and word embedding quality.

Table 2. The performance of seven models (Epoch=100, batch size = 128).

Model Name	Test Loss	Test Acc
FNN (one layer)	0.562	78.01%
FNN (two layer)	0.541	80.02%
FNN (three layer)	0.546	80.32%
CNN	0.372	84.66%
LSTM	0.339	86.56%
Bi-LSTM	0.286	89.39%
BERT	0.207	92.75%

The experimental results in Table 2 show: 1) For FFN, with the deepening of layers, the performance of FFN on the test set is gradually improved. In this paper, one to three layers of FNN are tested. Another test that deserves more attention is to continue to increase the number of layers of FFN. It is worth discussing whether the relationship between its performance and the number of layers can show a simple linear relationship. According to the specific task and data scale, achieving the most cost-effective model method is the next focus of this paper. 2) In comparing CNN, LSTM, BiLSTM, and BiLSTM, the performance of CNN is lower than that of LSTM and BILSTM models. This paper thinks that the reason is that RNNs are more suitable for dealing with sequential features. Compared with LSTM, BiLSTM has more advantages. This is because BiLSTM incorporates more sequence inforation. 3) BERT has the best performance among all models. The large-scale pre-training stage has brought BERT excellent performance and mobility, making it the mainstream model in natural language processing, but its shortcomings cannot be ignored. The computational complexity of the attention mechanism and the requirement of computing resources and data scale in the pre-training stage make BERT challenging to use on mobile devices.

4. Conclusion

With the experiment and analysis, this paper found that in the comparison of CNN, LSTM, BILSTM, and Bert, the performance of CNN is the worst. However, it is significantly higher than the three-layer fully linked neural network. In the experiment, the result of Bi-LSTM model is better than LSTM because it considers the sequence features of sentences more comprehensively. BERT performs best among the four models with the more advanced multi-head attention mechanism and large-scale pre-training strategy. The BERT model consists of the pre-training model and the downstream task model, in this situation, besides, it can run synchronously with downstream, meanwhile the model could support classification tasks of text, so when it processes text classification tasks there is unnecessary to modify it. Therefore, BERT is a widely used natural language processing model at this stage. By the end of 2020, the BERT model is used in almost all of Google's English query functions. But according to our research and analysis, it also has certain shortcomings that need to be continuously improved.

References

[1] F. Poecze, C. Ebster, and C. Strauss, "Social media metrics and sentiment analysis to evaluate the effectiveness of social media posts," *Procedia Computer Science*, vol. 130, pp. 660–666, 2018.

[2] M. Thomas and L. C.A, "Sentimental analysis using recurrent neural network," *International Journal of Engineering & Technology*, vol. 7, no. 2.27, p. 88, 2018.

[3] S. Sukheja, S. Chopra, and M. Vijayalakshmi, "Sentiment analysis using Deep Learning – A Survey," *2020 International Conference on Computer Science, Engineering and Applications (ICCSEA)*, 2020.

[4] K. Cho, B. van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio,

DOI: 10.54254/2755-2721/5/20230633

"Learning phrase representations using RNN encoder–decoder for statistical machine translation," *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014.

[5] X. Rong, "[PDF] word2vec parameter learning explained: Semantic scholar," *undefined*, 01-Jan-1970. [Online]. Available: https://www.semanticscholar.org/paper/word2vec-Parameter-Learning-Explained-Rong/940e8c477f3e7ddb1d3aa2f216a38c8f9486e544. [Accessed: 01-Nov-2022].

[6] Q. Zeng, X. Zhao, X. Hu, H. Duan, Z. Zhao, and C. Li, "Learning emotional word embeddings for sentiment analysis," *Journal of Intelligent & Fuzzy Systems*, vol. 40, no. 5, pp. 9515–9527, 2021.

[7] M. Arora and V. Kansal, "Character level embedding with deep convolutional neural network for text normalization of unstructured data for Twitter sentiment analysis," *Social Network Analysis and Mining*, vol. 9, no. 1, 2019.

[8] N. Garg, L. Schiebinger, D. Jurafsky, and J. Zou, "Word embeddings quantify 100 years of gender and ethnic stereotypes," *Proceedings of the National Academy of Sciences*, vol. 115, no. 16, 2018.

[9] W. zhang, S. Jiang, S. Zhao, K. Hou, Y. Liu, and L. Zhang, "A Bert-BILSTM-CRF model for Chinese Electronic Medical Records named entity recognition," *2019 12th International Conference on Intelligent Computation Technology and Automation (ICICTA)*, 2019.

[10] Z. Dai, X. Wang, P. Ni, Y. Li, G. Li, and X. Bai, "Named entity recognition using Bert BiLSTM CRF for Chinese Electronic Health Records," *2019 12th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, 2019.

[11] Q. Zhang, Y. Sun, L. Zhang, Y. Jiao, and Y. Tian, "Named entity recognition method in health preserving field based on bert," *Procedia Computer Science*, vol. 183, pp. 212–220, 2021.