Proceedings of the 3rd International Conference on Signal Processing and Machine Learning
DOI: 10.54254/2755-2721/5/20230694

Analysis of sentiment analysis model based on deep learning

Guanchao Wang[1]

[1]Department of Linguistics, University of British Columbia, 2613 West Mall, Vancouver, BC V6T 1Z4

guanch1@student.ubc.ca

Abstract. A traditional yet important topic in the study of natural language processing is sentiment analysis. Deep learning models have gradually taken over as one of the primary techniques for resolving sentiment analysis issues over the last ten years. Common deep learning models targeting sentiment analysis tasks include both recurrent and convolutional neural networks, as well as the BERT model. The current research examines the classificational accuracy of numerous deep learning models with diverse structural types in order to compare their performance in sentiment analysis. Results from the experiments suggest that the pre-training BERT model achieves the highest accuracy, while the convolutional neural network appears to sustain better results on sentiment analysis than standard recurrent neural networks.

Keywords: Sentiment analysis, natural language processing, deep learning.

1. Introduction

Sentiment analysis refers to the subfield of natural language processing (NLP) that focus on extracting and analyzing subjective information from text. A lot of focus has been paid to the task of sentiment analysis by both academia and the industry in recent years due to the surge of web text [1, 2]. The rapid development of artificial neural networks (ANN) in the recently has also onset the trend to apply deep learning (DL) methods in solving sentiment analysis tasks, those methods include Convolutional Neural Networks (CNN), Recurrent Neural Networks and Deep Belief Networks (DBN)[2]. Moreover, to address the problems with vanishing gradient in RNN, a modified RNN model named Long-Short Term Memory (LSTM) is suggested by Hochreiter and Schmidhuber, which constitutes the base for the newer model called the bidirectional LSTM (BiLSTM) [3]. With the recent development in the field of natural language understanding (NLU), contextualized language models like GPT-3, a transformer model pre-trained by OpenAI, and bidirectional encoder representations from transformers (BERT). Born with the purpose to outperform previous models, pre-trained language models like BERT and the robustly optimized BERT pretraining approach (ROBERTA) is better at capturing the contextual information of given texts, however, researchers have reported that they are not sufficient in extracting aspect-level information, which refers to the information concerning a single word or a little cluster of words [4, 5]. Specifically, models BERT alike tend to break what is syntactically a noun phrase into smaller elements, for instance, “bottle of water” would become “bottle of” and “water”. Some researchers believe that the lack of attention to syntax-related info of those models should be accountable for their weakness in spotting groups of words, to resolve this issue, Phan and Ogunbona suggested an aspect-extraction (AE) architecture named contextualized syntax-based aspect extraction (CSAE), which combine a ROBERTA network with part-of-speech (POS)

DOI: 10.54254/2755-2721/5/20230694

embedding, dependency-based embeddings and self-attention mechanism. In comparison, it is found that the CSAE model brings improvement in both AE and aspect sentiment classification (ASC) tasks[5].

There have also been some novel attempts to combine different DL architectures to make use of their respective advantages. For instance, Rhanoui et al. have proposed a neural network model called CNN-BiLSTM. In this model, the input text first entered CNN layers, in which some of the features are extracted[6]. After the results went through max pooling layers and a process of concatenation, the output of the CNN architecture then became the input of the BiLSTM layers. Rhanoui et al. compared the performance of the novel CNN-BiLSTM and previously established models including CNN, LSTM, BiLSTM and CNN-LSTM in a document-level sentiment analysis task, and the results show that out of all the models tested, the newly proposed CNN-BiLSTM achieves the highest accuracy rate. Although Rhanoui et al.'s model brings slight improvement over other models, it is suggested that the attention mechanism, which is lacking in the CNN-BiLSTM, should be incorporated in it in future improvement of the model. Following the same trend to improve performance by combining existing architectures, Basiri et al. suggested a new model termed the attention-based bidirectional CNN-RNN deep model (ABCDM)[7]. To excel in catching long-term dependencies as well as local features of input text, ABCDM's architecture is formed by incorporating the global vectors for word representation (Glove), bidirectional LSTM, bidirectional gated recurrent unit (GRU), CNN and an attentional component. By testing ABCDM's performance at document-level sentiment analysis tasks with data from a tweet, the authors found that ABCDM yields up-to-date results when classifying reviews of various sizes.

2. Methods

This paper investigates the different DL models' performance over sentiment analysis tasks. The DL models presented in this paper include RNN, CNN, LSTM, BILSTM, BERT, and Feedforward Neural Networks (FNN). This chapter mainly introduces the definitions of the above six deep learning models.

2.1. Rnn

The recurrent neural network is a DL model mainly implemented to deal with sequential data, it tackles tasks common feedforward neural networks are incapable of doing since they cannot transmit and utilize historical data [8]. Let the word embedding expression of the existing text sequence be $x_{1,\dots,}x_{i,\dots,}x_t$. RNN model can update the hidden state of the moment through the input state $x_i$of the moment and the output state of the previous moment h_{i-1}. The hidden state expression is shown in Formula (1) and (2):

$$h_t = g(Uh_{t-1} + Vx_t + b_h) \quad (1)$$

$$o_t = g(Wh_t + b_o) \quad (2)$$

where g denotes the activation function, W, V, and U signify the parameter matrices, and the neurons all share the same information concerning each parameter. The output of the final moment then became each text sequence's feature vector, which is subsequently fed into the fully connected layer, which subsequently went through a SoftMax layer, after which the final classificational result is printed.

2.2. Cnn

A CNN is a type of FNN featuring a deep structure and convolutional calculations, which is an iconic algorithm in DL.

An input layer, an output layer, and several hidden layers between the first two layers make up the core parts of a CNN. Typical CNN input layers can handle input data with a dimensionality greater than one. Generally, a 1-dimensional(1-D) CNN's input layer would accept arrays, be it 1-D or 2-D, and a 1-D array usually denotes time or the spectrum samples. A 2-D array can comprise multiple channels. Similarly, a 2-D CNN's input layer can take 2-D or 3-D arrays. Furthermore, input layers of a 3-D CNN are capable of handling 3-D or 4-D arrays.

DOI: 10.54254/2755-2721/5/20230694

A CNN's hidden layer of starts with a convolutional layer, with a pooling layer attached to its end, which is then joined to a fully connected layer. The overall structure and principal mechanism of a CNN are essentially the same as an FNN since the upstream of an output layer in a CNN is often a fully connected layer.

2.3. Lstm

Because the input of all neurons in the RNN model contains the output of the previous unit when the sequence data grows to a certain length, the circulating neural network will have a "long-distance dependence problem", and the influence of the information input at an earlier time on the whole sequence data will be smaller and smaller. In the process of backpropagation, the gradient of every moment is transmitted to the prior neuron, thus leading to some issues called gradient disappearance and gradient explosion. In order to solve the problems mentioned above, which is what the cyclic neural network encounter when processing sequential data, researchers put forward a variant of RNN: LSTM [9]. LSTM adds a new gating mechanism, which controls the information flow through the forgetting gate, input gate, and output gate, and alleviates the problems with gradient disappearance and gradient explosion. The following is a detailed description of the gating mechanism.

Forgetting gate: The primary purpose of a forgetting gate is to control the retention degree of the information given by the previous moment. By combining the output h_{t-1} of the previous moment with the input data x_t of the current moment, the current moment acquires its input. The probability to forget the previous moment's information can be obtained by activating the sigmoid function. Details of this process are shown in Formula (3):

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{3}$$

Input gate: The input gate essentially regulates the information coming in from the neurons, including how current time data is processed and how prior time data is remembered. The current time data x_t is respectively subjected to the sigmoid function to acquire the outputting i_t and the Tanh activation function to get the result $\widehat{C_t}$. Specifically, the updated state C_t of memory cells at the current time can be expressed by the formula (4), (5) and (6):

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{4}$$
$$\hat{C}_t = tanh(W_c \cdot [h_{t-1}, x_t] + b_C) \tag{5}$$
$$C_t = f_t^* C_{t-1} + i_t * \hat{C}_t \tag{6}$$

Output gate: The primary function of an output gate is to control the cells to discard some unnecessary memory information at the current moment. The details are shown in Formula (7) and (8):

$$o_t = \sigma(W_o * [h_{t-1}, x_t] + b_o) \tag{7}$$
$$h_t = o_t^* \, tanh(C_t) \tag{8}$$

2.4. Bilstm

BiLSTM model splices forward as well as the backward outputs of LSTM units at time t, as shown in Formula (9):

$$h_t = [\overrightarrow{h_t}; \overleftarrow{h_t}] \in R^n \tag{9}$$

BiLSTM layers take the output vector of the CNN layers as input and randomly initialize the normal assignment of weight matrix W and offset matrix B of the gate control mechanism. LSTM unit calculates the values of the forgetting, input gate, and output gates respectively based on the output information of the hidden layer at the last moment and the input information at the current moment and integrates them with the stored information of the LSTM unit at the last moment. Get the current output of the LSTM unit and update the hidden layer output and storage information so that the input for the LSTM unit the next time will be the current output[10]. Finally, feature vectors with

DOI: 10.54254/2755-2721/5/20230694

bidirectional semantics are produced by splicing the forward and backward LSTM unit output vectors. Moreover, the dropout mechanism is simultaneously presented in the input of the LSTM unit and the output of the hidden layer to solve the over-fitting problem caused by large parameters of the training model.

2.5. Bert

BERT model uses bidirectional embedding representation based on Transformer structure for text representation and adds sentence embedding and position embedding in addition to word embedding, which can well grasp the global information and the relationship between words and sentences, dramatically improving the original model, and is outstanding in all NLP tasks [11, 12]. The transformer layer is the main framework of BERT, which consists of several Encoders and Decoder[13]. The Encoder consists of four layers: the first layer is the Multi-Head Attention mechanism the second layer is a residual network, the third layer is a feedforward neural network, and the fourth layer realizes normalization operation at the end. Encoder adds encoder-Decoder attention layer based on the Encoder [14]. It realizes the decoding and re-sequencing of information, in which the multi-attention layer is the core of the Transformer layer. Its main idea is to adjust the weight of a word by calculating the correlation degree between words, which reflects the connection strength between the word and the adjacent words of the same sentence, and then reflects the importance of each word to the semantic expression of the sentence. First, the input sequence enters the Encoder. The matrix represents the target word, the matrix represents the context words, the original matrix represents the target word, and the context words are obtained through a linear transformation. Then, the self-attention value is obtained through scaling dot product operation, which reflects the correlation degree between a word and other words in the input sentence when the model encodes it. Finally, the self-attention value is spliced and linearly transformed to obtain an output vector of the same size as the text input by the model. This vector contains enhanced semantics, which can improve the overall effect of the algorithm. 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. The experimental outcome of the Bi-LSTM model appears to be 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 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 rather versatile NLP model at this stage. By the end of 2020, the BERT model will be used in almost all of Google's English query functions. Nevertheless, according to our research and analysis, it also has certain shortcomings that need to be continuously improved.

2.6. Nn

Common feedforward neural networks include the perceptron network and BP network. Perceptron network is the simplest feedforward neural network, which is 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 networks and multi-layer sensor networks. BP network refers to the feedforward network with a backpropagation learning algorithm. Unlike a perceptron, a BP network feeds the results of neuron transformations to a sigmoid function, thus, it is capable of mapping nonlinear relationships between input and output.

3. Results & analysis

This section presents implementations of several different models for the same sentiment analysis task, including RNN, CNN, LSTM, BILSTM, BERT, and Feedforward Neural Network (FNN) with 1, 2, and 3 layers, respectively. This section uses similar parameters and tools for all the models. All new models used in this section used Adam, an optimizer better than the commonly used SGD optimizer. In addition, BCEWithLogitLoss is used as the loss function for all models, and all models are trained

DOI: 10.54254/2755-2721/5/20230694

for five epochs. However, it should be noted that some models have different learning rates: while RNN, CNN, LSTM, and BiLSTM are implemented with a learning rate of 1.00E-03, the 3 FNN models all have a learning rate of 1.00E-01, while the BERT model alone has a learning rate of 2.00E-05. A comparison of the performance of different models is shown in figure 1 and table 1.

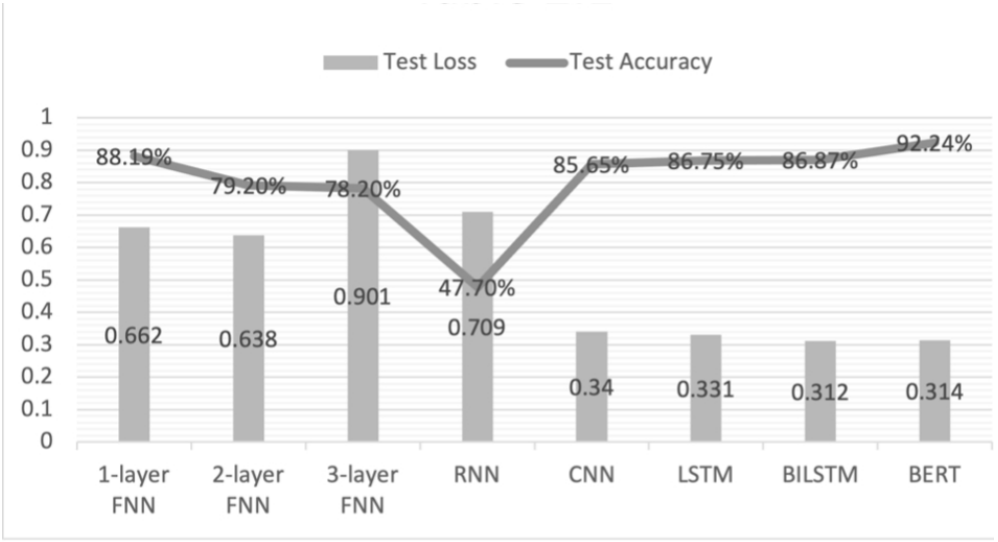

Figure 1. model-wise comparison.

As figure 1 shows, the RNN model used in Assignment 1 has the lowest test accuracy. This is because the RNN model itself is quite simple. It cannot extract sequential information. Furthermore, the RNN model suffers from gradient vanishing when dealing with long sequences, and thus, can't get the optimized solution. CNN appears to perform better in sentiment analysis than RNN since the accuracy it provides is 37.95% higher than that of RNN. This is because a CNN model uses convolutional layers of various sizes to extract semantic information, which resembles the characteristics of N-gram models and thus outperforms RNNs at grasping the meaning of a sentence. LSTM has a satisfactory test accuracy of 86.75% because it uses the mechanisms of cell and gate to assuage the problem of gradient vanishing. The BiLSTM is better than LSTM at gripping the semantics of a sentence since it can catch the bidirectional meaning of a sequence. Coming up next is the 1-layer FNN, which has an astonishing test accuracy of 88.19%. However, as the numbers of layers increase, the accuracy rate drops significantly due to overfitting. To verify this, we applied the method of dropout to the 3-layers FNN and improved the accuracy rate to 89.12% while keeping all other parameters unchanged. Finally, table 1 shows that BERT, a Transformer model–based pre–trained language model, yields the best results among all models used. This paper concludes that the high accuracy of BERT is due to its use of attention and the masking mechanism.

Table 1. Experimental results.

Model	Learning Rate	Loss	Test Accuracy
1-layer FNN	1.00e-01	0.662	88.19%
2-layer FNN	1.00e-01	0.638	79.20%

DOI: 10.54254/2755-2721/5/20230694

Table 1. (continue)

Model	Learning Rate	Loss	Test Accuracy
3-layer FNN	1.00e-01	0.901	78.20%
RNN	1.00e-03	0.709	47.70%
CNN	1.00e-03	0.34	85.65%
LSTM	1.00e-03	0.331	86.75%
BILSTM	1.00e-03	0.312	86.87%
BERT	2.00e-05	0.314	92.24%

4. Conclusion

In this paper, RNN, CNN, LSTM, BiLSTM, BERT, and FNN are used to verify the accuracy of the sentiment analysis task. The final statistics indicate that the BERT model yields the optimum accuracy under the same experimental configuration as possible, and the accuracy rate on the test set reaches 0.9224. Compared with RNN, CNN has achieved better results, and the CNN model's accuracy on the test set has improved by 37.95%. BiLSTM is slightly better than LSTM, and its accuracy on the test set is improved by 0.12% compared with LSTM. To sum up, in the sentiment analysis task, the semantic information can be better extracted using the convolutional layer, and the text classification model using pre-trained word vectors can achieve higher accuracy than the model without pre-trained word vectors. However, there is still much room for improvement in how to design an appropriate word vector embedding method, and how to design a more complex model for specific tasks will continue to advance as a future research task.

References

[1] Hoang, Mickel, Oskar Alija Bihorac, and Jacobo Rouces. "Aspect-based sentiment analysis using bert." In Proceedings of the 22nd nordic conference on computational linguistics, pp. 187-196. 2019.

[2] Birjali, Marouane, Mohammed Kasri, and Abderrahim Beni-Hssane. "A comprehensive survey on sentiment analysis: Approaches, challenges and trends." Knowledge-Based Systems 226 (2021): 107134.

[3] Hochreiter, Sepp, and Jürgen Schmidhuber. "Long short-term memory." Neural computation 9, no. 8 (1997): 1735-1780.

[4] Araque, Oscar, Ganggao Zhu, and Carlos A. Iglesias. "A semantic similarity-based perspective of affect lexicons for sentiment analysis." Knowledge-Based Systems 165 (2019): 346-359.

[5] Phan, Minh Hieu, and Philip O. Ogunbona. "Modelling context and syntactical features for aspect-based sentiment analysis." In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pp. 3211-3220. 2020.

[6] Rhanoui, Maryem, Mounia Mikram, Siham Yousfi, and Soukaina Barzali. "A CNN-BiLSTM model for document-level sentiment analysis." Machine Learning and Knowledge Extraction 1, no. 3 (2019): 832-847.

[7] Basiri, Mohammad Ehsan, Shahla Nemati, Moloud Abdar, Erik Cambria, and U. Rajendra Acharya. "ABCDM: An attention-based bidirectional CNN-RNN deep model for sentiment analysis." Future Generation Computer Systems 115 (2021): 279-294.

DOI: 10.54254/2755-2721/5/20230694

[8] Zaremba, Wojciech, Ilya Sutskever, and Oriol Vinyals. "Recurrent neural network regularization." arXiv preprint arXiv:1409.2329 (2014).

[9] Zia, Tehseen, and Usman Zahid. "Long short-term memory recurrent neural network architectures for Urdu acoustic modeling." International Journal of Speech Technology 22, no. 1 (2019): 21-30.

[10] Wang Ruojia， Wei Siyi， Wang Jimin. Research on the application of BiLSTM-CRF model in Chinese electronic medical record named entity recognition[J]. Journal of Literature and Data， 2019, 1 (2): 53-66

[11] Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. "Bert: Pre-training of deep bidirectional transformers for language understanding." arXiv preprint arXiv:1810.04805 (2018).

[12] Han, Kai, An Xiao, Enhua Wu, Jianyuan Guo, Chunjing Xu, and Yunhe Wang. "Transformer in transformer." Advances in Neural Information Processing Systems 34 (2021): 15908-15919.

[13] Badrinarayanan, Vijay, Alex Kendall, and Roberto Cipolla. "Segnet: A deep convolutional encoder-decoder architecture for image segmentation." IEEE transactions on pattern analysis and machine intelligence 39, no. 12 (2017): 2481-2495.

[14] Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. "Attention is all you need." Advances in neural information processing systems 30 (2017).