ISSN 2616-5775 Vol. 6, Issue 13: 92-98, DOI: 10.25236/AJCIS.2023.061314

BiLSTM Text Classification Incorporating Attentional Mechanisms

Hao Qin*

Tianjin University of Commerce, Tianjin, 300134, China
**Corresponding author: qhao063@163.com*

Abstract: *Most of the current research on text classification studies feature extraction or global information extraction at the surface level, which considers all features as important features, greatly increasing the model computational power. The existing text classification model bi-directional long short-term memory (BiLSTM) can learn text contextual information, but cannot be targeted to the extraction of important features and special attention. This paper incorporates the attention mechanism into the BiLSTM model, so that the model in the learning of contextual information, while the model This paper integrates the attention mechanism into the BiLSTM model, which makes the model learn contextual information while extracting locally important feature information, reduces the number of meaningless text features, and finally speeds up the convergence of the model, and ultimately achieves a higher accuracy of the classification results.*

Keywords: *BiLSTM, text classification, attention mechanism*

1. Introduction

Currently, the Internet has entered the era of big data, a large amount of information is constantly generated, including text, audio, pictures, video, etc., of which the largest number of text information, access to a wide range of ways to network data, social media, user comments, including a number of paths can provide a good source of information for text data. However, the biggest difference between text data and other data types is its unstructured nature, i.e., the inner connection of each text is hidden in the semantic information, and it is not possible to obtain the internal semantic information from the external structure. Therefore, the key to solving the text classification problem lies in how to effectively obtain semantic information and effectively classify text information is particularly important, which also gave birth to the birth of text classification technology [1].

The purpose of text classification technology is to achieve automatic classification of text, to solve the problem of information clutter, to provide an efficient method of information classification and access to information, while text classification technology is also one of the basic technologies of data mining, which can initially process the text information, tag it with category labels, and provide a kind of coarse-grained textual semantic information [2].

In specific domains, text categorisation techniques can also provide fine-grained text semantic information. For example, in the categorisation of reviews on shopping websites, text categorisation techniques can provide information about the emotional tendency of the reviews. In addition, text classification techniques have a fundamental role in the research of other areas of data mining. For example, some algorithms and models of text categorisation techniques have good results in the field of sentiment analysis, so text categorisation techniques are an area of great research value.

2. Literature review

Traditional text classification methods manually extract the features which are later fed into the classifier for training. Wang introduce a convolutional recurrent neural network for text classification, which enjoys both the advantages of convolutional neural networks for extracting local features from text and also those of recurrent neural networks (LSTM) in memory to connect the extracted features[1]. Yao et. Al propose to use graph convolutional networks for text classification[2]. Experimental results show that the improvement of Text GCN over state-of-the-art comparison methods become more prominent as (Yao et. al., 2019) lower the percentage of training data, suggesting the robustness of Text

ISSN 2616-5775 Vol. 6, Issue 13: 92-98, DOI: 10.25236/AJCIS.2023.061314

GCN to less training data in text classification. Recently, deep neural networks have achieved promising performance in the text classification task compared to shallow models. Despite of the significance of deep models, they ignore the fine-grained (matching signals between words and classes) classification clues since their classifications mainly rely on the text-level representations. To address this problem Du introduce the interaction mechanism to incorporate word-level matching signals into the text classification task[3]. Bao Al explore meta-learning for few-shot text classification, demonstrate that the model consistently outperforms prototypical networks learned on lexical knowledge (Snell et al., 2017) in both few-shot text classification and relation classification by a significant margin across six benchmark datasets (20.0% on average in 1-shot classification)[4]. To tackle the problems, Huang propose a new GNN based model that builds graphs for each input text with global parameters sharing instead of a single graph for the whole corpus[5]. Experiments show that the model outperforms existing models on several text classification datasets even with consuming less memory.The key technology for gaining the insights into a text information and organizing that information is known as text classification. Shah study a comparative analysis of logistic regression, random forest and knn models for the text classification. A BBC news text classification system is designed[6].The task of text classification is usually divided into two stages: text feature extraction and classification. Inspired by the current trend of formalizing NLP problems as question answering tasks Chai propose a new framework for text classification, in which each category label is associated with a category description[7].Deep learning based models have surpassed classical machine learning based approaches in various text classification tasks, including sentiment analysis, news categorization, question answering, and natural language inference.Minaee provide a comprehensive review of more than 150 deep learning based models for text classification developed in recent years, and discuss their technical contributions, similarities, and strengths[8].The task of text classification is usually divided into two stages: text feature extraction and classification. Inspired by the current trend of formalizing NLP problems as question answering tasks Wu propose a new framework for text classification, in which each category label is associated with a category description[9].Deep learning--based models have surpassed classical machine learning--based approaches in various text classification tasks, including sentiment analysis, news categorization, question answering, and natural language inference. Minaee provide a comprehensive review of more than 150 deep learning--based models for text classification developed in recent years, and discuss their technical contributions, similarities, and strengths[10].

3. LSTM model based on attention mechanism

3.1 The Long Short-Term Memory (LSTM)

The Long Short-Term Memory (LSTM) network is a variant of the Recurrent Neural Network (RNN), a network structure originally introduced by Hochreiter et al. and gradually refined as a tool for dealing with time-series prediction problems.LSTM is different from the internal structure of the RNN network, which is subject to long term dependencies, whereas the internal structure of the network of the LSTM is more complex, and this is achieved by introducing the sigmoid function in combination with the tanh function. LSTM is different from the internal structure of RNN network, and the internal structure of LSTM network is more complex, by introducing the sigmoid function and combining it with tanh function, which is achieved by using the input gate, forgetting gate, and output gate, and the repeating module does not just have a simple tanh structure, so as to avoid the long term dependence of RNN, LSTM can regulate the state at any time through the structure of gates, and by adding the summation operation, LSTM has solved the problem of the disappearance of the gradient successfully, by connecting the short term memory and the long term memory. LSTM has the ability to delete or add nodes, the information is passed to the cell state through three gates, where the forget gate decides whether the previous cell information is passed to the current cell, the input gate determines the information update of the cell from the current input, and the output of the current cell is obtained based on the newest state, the previous output and the current input, which is also known as the output gate, and each of the gates is represented by the neural network, which has an input layer, a hidden layer, a hidden memory, and a hidden network, and each of them is represented by a neural network. Each gate is represented by a neural network with input, hidden and output layers.

At time t, the LSTM has three input values, namely the input of the current network, the output of the previous network h_{t-1}, and the state of the previous cell c_{t-1}. At the same time, the LSTM network produces two output values, namely, the output of the current moment h_t, and the state of the current

ISSN 2616-5775 Vol. 6, Issue 13: 92-98, DOI: 10.25236/AJCIS.2023.061314

cell c_t. The input gates control whether the information in the state cell is retained or deleted, and the output gates control the long-term state information of the state cell as the output of the current network. Long-term state information is controlled as the output of the current network. The following equation represents the update of each state cell at time t of the long and short term memory network.

$$\begin{aligned} f_t &= \sigma\left(W_{fx}x_t + W_{fh}h_{t-1} + b_f\right) \\ i_t &= \sigma(W_{ix}x_t + W_{ih}h_{t-1} + b_i) \\ \widetilde{c}_t &= \tanh(W_{cx}x_t + W_{ch}h_{t-1} + b_c) \\ c_t &= f_t * c_{t-1} + i_t * \widetilde{c}_t \\ o_t &= \sigma\left(W_{ox}x_t + W_{oh}h_{t-1} + b_o\right) \\ h_t &= o_t * \tanh\left(c_t\right) \end{aligned} \tag{1}$$

where σ is the sigmoid function, x_t is the input at time t, f_t, i_t, and o_t represent the forgetting gate, input gate, and output gate, respectively, W is the weight matrix, and b is the bias matrix.

The backpropagation algorithm is one of the most commonly used algorithms in training artificial neural networks, and it is also used for the training of LSTM. Specifically, the training algorithm of LSTM can be divided into three main steps.

First, the outputs of all neural units are computed for the LSTM network forward, i.e., the outputs of all neural units are computed by Eq. (1) calculate the value of f_t, i_t, o_t, h_t, c_t;

Second, the backward computation yields the error terms for all neural units δ

Third, the weight gradient is calculated from the error term.

Finally after processing all the time steps, we get a sequence of hidden states $h = \{h_1, ..., h_n\}$.

In traditional RNN and LSTM models, the acquired information can only be propagated forward, resulting in the state of moment t only depending on the information before it. For natural language processing tasks, the context of the words are critical, in order to obtain the contextual information contained in moment t, a bidirectional long short-term memory neural network (BiLSTM) model is proposed to capture the contextual information [1],the structure is shown in Fig. 1.The principle of BLSTM is to read the training data from both temporal directions and train the neural network. In BiLSTM, the prediction is done by connecting the left and right summary vectors. Since BiLSTM acquires both before and after information, it can produce better a prediction.

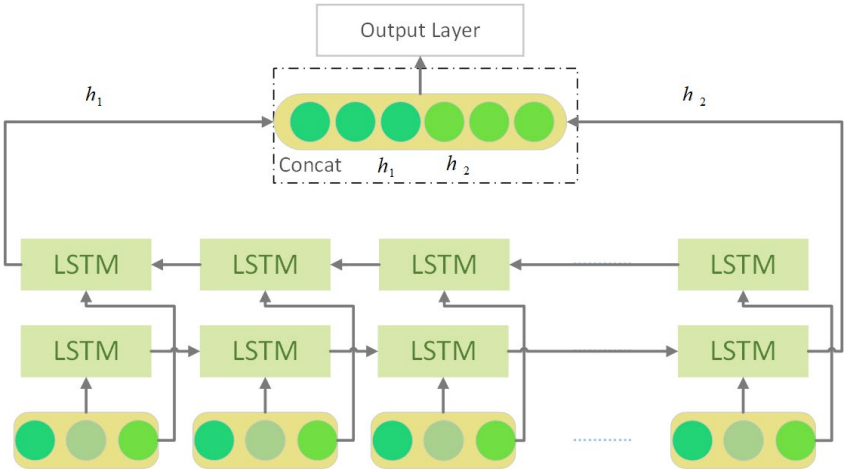

Figure 1: BiLSTM

ISSN 2616-5775 Vol. 6, Issue 13: 92-98, DOI: 10.25236/AJCIS.2023.061314

3.2 Attention mechanism

Attention is an important cognitive function inherent in humans, which serves to enable individuals to selectively focus on the information they need while ignoring unimportant information [1]. In the case of limited computational resources, the attention mechanism becomes an efficient resource allocation tool, which helps to focus limited resources on processing more important information, thus effectively alleviating the information overload problem. This mechanism not only plays a key role in human cognition, but also has been applied in neural networks.

Neural networks, as information processing systems, also need to cope with the challenge of large amounts of input information. By introducing a mechanism similar to human attention, neural networks can cope with information input in complex environments more effectively. Focusing on the processing of critical information can help to improve the efficiency of neural networks and enable them to learn and infer more accurately.

The process of calculating the attentional mechanism is shown in equation (2), h_{it} is the ith text representation vector at time t obtained by LSTM, W_w is the weighting matrix, α_{it} weighting of attention, s_i is the weighted vector representation.

$$u_{i,t} = \tanh\left(W_w h_{i,t}\right)$$
$$\alpha_{i,t} = \frac{\exp\left(u_{i,t}^{T} u_w\right)}{\sum_t \exp\left(u_{i,t}^{T} u_w\right)} \tag{2}$$
$$S_i = \sum_t \alpha_{i,t} h_{i,t}$$

The collation design of the model in this paper is shown in Figure 2.

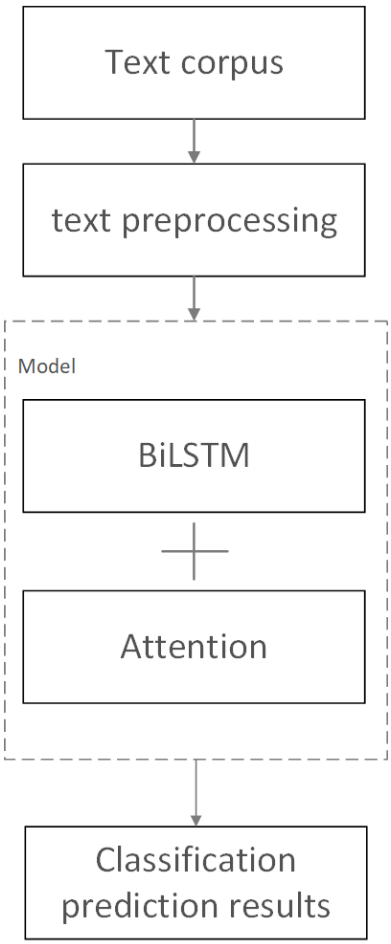

Figure 2: Text classification process

ISSN 2616-5775 Vol. 6, Issue 13: 92-98, DOI: 10.25236/AJCIS.2023.061314

4. Data pre-processing

4.1 Data set production

The data used in this paper is comment text data with two categories: positive and negative, where 0 represents negative and 1 represents positive. The form of the dataset is shown in Figure 3.

```
<Polarity>1</Polarity>
<text>great bagels made the old-fashioned way.</text>

<Polarity>0</Polarity>
<text>i will never visit this restaurant again.</text>

<Polarity>0</Polarity>
<text>avoid this place!</text>

<Polarity>1</Polarity>
<text>i also recommend the rice dishes or the different varieties of congee (rice porridge).</text>

<Polarity>0</Polarity>
<text>bartender was unable to tear himself away from friends at bar.</text>
```

Figure 3: Comment dataset

Use python language to clean the text data, use regular expression to extract the text content and category respectively, when processing the text, the length of each tweet is controlled by setting the variable pad_size, in this paper the size of pad_size is set to 20, then calculate the length of each tweet. If the length of the text is less than 20, fill the <PAD> mark at the end of the text, so that the text reaches the specified length, if the length of the text is more than 30, only the former pad size part is intercepted, and through this processing, all of the text is processed into a length of 20. After that, all the text will be converted into lowercase letters to complete the dataset production. In this paper, the training set, validator and test set are divided according to the ratio of 7:1:2.

4.2 Word vector transformation

After extracting the text and categories based on the original data, it is necessary to generate word vectors based on the text, i.e., to establish the one-to-one correspondence between words and vectors, and at the same time to establish the semantic connection between words. In this paper, we use word2vector to transform the text data into word vectors and retain the similarity of words and other properties. At the same time a dictionary matrix containing all the text sentences is generated, where the keys are the words and the values are the corresponding word vector tensors, and then by traversing the text in the dataset, the word vectors corresponding to each word are found and they are combined into vectors that represent the whole text. Also because the produced dataset is a supervised learning task with labels, the labelled data also needs to be processed and converted into the corresponding tensor representation. Finally, the text vectors and labels in the dataset are combined into training samples for neural network model training. The word vector form is shown in Figure 4.

```
tensor([[-0.3965,  0.7025,  0.1428,  ..., -0.6899,  0.0155, -0.2765],
        [-0.4372,  0.7589,  0.1490,  ..., -0.7336,  0.0018, -0.3006],
        [-0.3921,  0.7137,  0.1245,  ..., -0.6743,  0.0200, -0.2944],
        ...,
        [-0.0089,  0.0286,  0.0123,  ..., -0.0167,  0.0063, -0.0060],
        [ 0.2961,  0.5166,  0.2517,  ...,  0.8363,  0.9010,  0.3950],
        [ 0.8809,  0.1084,  0.5432,  ...,  0.1735,  0.9247,  0.6166]])
```

Figure 4: Word vector

4.3 Parameterisation

When training the model, the first step was to set up the model hyperparameters. The training period was set to 50, the training batch was set to 128, the word vector dimension was set to 100 dimensions, the hidden layer neurons were set to 128, the layer layer was set to 2, and the Adam optimisation algorithm was used to update the neural network weights. The model hyperparameters are set as shown in Table 1.

ISSN 2616-5775 Vol. 6, Issue 13: 92-98, DOI: 10.25236/AJCIS.2023.061314

Table 1: Hyperparameter settings

parametric	Value
Word embedding	100
BiLSTM hidden size	128
n_layers	2
Epochs	50
Batch size	128
Dropout	0.5
Learning rate	0.001

Training Periods (Epochs): In this paper, the training period of the model is set to 50 rounds, which is based on the training process of the model. During the experimental period, this paper conducted several trials on the model and observed the changes of training loss and validation loss. It was found that when the training period of the model reached 50 rounds, the performance of the model gradually stabilised and converged. Therefore, 10 rounds are chosen as the number of training cycles for the model proposed in this paper to adequately train the model and obtain the best performance.

In conclusion, the selection of parameters in this paper is based on the results of repeated testing and observation of model performance, and efforts are made to ensure that these parameters can achieve the best model performance under the given hardware and dataset conditions in the experimental process, and satisfactory results have been achieved in the experiments[11-14].

5. Experimental results

The task of this paper is to binary classify the review data and use the accuracy to evaluate the model performance.

(1) accuracy

The proportion of correct predictions out of the total of all predictions was used to evaluate the experimental results with the following formula.

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{3}$$

In this paper, we finally get 96.69% accuracy on the test set, and the accuracy of the training set, validator, and test set during the training period is shown in Figure 5.

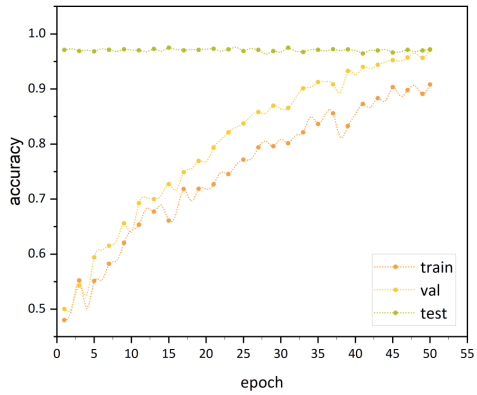

Figure 5: Accuracy of each stage of model training process

6. Conclusion

In this paper, by integrating the attention mechanism into LSTM not only can extract the long-distance contextual information in the text, but also can extract the key local information in the tweets, which optimises the feature extraction ability of the model, so that the attention of the model can be more

ISSN 2616-5775 Vol. 6, Issue 13: 92-98, DOI: 10.25236/AJCIS.2023.061314

focused on the extraction of the important information, and ultimately achieve a higher accuracy of the classification effect.

References

[1] Yan Jiayu. Research on Chinese text classification based on improved recurrent neural network [D]. Nanjing University of Information Engineering, 2023.DOI:10.27248/d.cnki.gnjqc.2022.000576.
[2] Zhang, Chong. Research on text classification techniques based on Attention-Based LSTM model [D]. Nanjing University, 2016.
[3] Ruishuang Wang; Zhao Li; Jian Cao; Tong Chen; Lei Wang; "Convolutional Recurrent Neural Networks for Text Classification",2019 International Joint Conference On Neural Networks ..., 2019.
[4] Liang Yao;Chengsheng Mao;Yuan Luo; "Graph Convolutional Networks For Text Classification", AAAI, 2019.
[5] Cunxiao Du; Zhaozheng Chen; Fuli Feng; Lei Zhu; Tian Gan; Liqiang Nie; "Explicit Interaction Model Towards Text Classification", AAAI, 2019.
[6] Yujia Bao; Menghua Wu; Shiyu Chang; Regina Barzilay; "Few-shot Text Classification With Distributional Signatures", ARXIV-CS.CL, 2019.
[7] Lianzhe Huang; Dehong Ma; Sujian Li; Xiaodong Zhang; Houfeng WANG; "Text Level Graph Neural Network For Text Classification", EMNLP, 2019.
[8] Kanish Shah; Henil Patel; Devanshi Sanghvi; Manan Shah; "A Comparative Analysis of Logistic Regression, Random Forest and KNN Models for The Text Classification", AUGMENTED HUMAN RESEARCH,2020.
[9] Duo Chai; Wei Wu; Qinghong Han; Fei Wu; Jiwei Li; "Description Based Text Classification With Reinforcement Learning", ARXIV-CS.CL, 2020.
[10] Shervin Minaee; Nal Kalchbrenner; Erik Cambria; Narjes Nikzad; Meysam Chenaghlu; Jianfeng Gao; "Deep Learning Based Text Classification: A Comprehensive Review", ARXIV-CS.CL, 2020.
[11] Wei Wu; Duo Chai; Qinghong Han; Fei Wu; Jiwei Li; "Description Based Text Classification with Reinforcement Learning",ICML,2020.
[12] Shervin Minaee; Nal Kalchbrenner;Erik Cambria; Narjes Nikzad; Meysam Chenaghlu; Jianfeng Gao;"Deep Learning--based Text Classification",ACM COMPUTING SURVEYS (CSUR),2021.
[13] Kang Lei. Deep neural network in short text classification [D]. Lanzhou University,2021. DOI:10.27204/d.cnki.glzhu.2021.002138
[14] W.Q. Xu. Research on text classification algorithm based on graph neural network[D]. Jilin University, 2022.DOI:10.27162/d.cnki.gjlin.2022.007439