Scholar Publishing Group
International Journal of Neural Network
https://doi.org/10.38007/NN.2021.020105
ISSN 2790-2692 Vol. 2, Issue 1: 35-43

Improved Text Classification Algorithm Based on Neural Network

Saravanan Saini*

Tennessee State University, USA

**corresponding author*

Keyword: Text Classification, Neural Network, Text Feature, Baseline Model

Abstract: In recent years, another idea of text modeling based on graph structured data has been continuously developed. Unlike Bert, graph neural network is a graph based deep learning network, which can capture the dependency relationship in a graph through message aggregation between neighboring nodes. It makes up for the problem that traditional deep learning networks can not process graph structure data, and is also increasingly used in text classification tasks. At present, there are still many problems in the text classification methods of graph networks. The graph convolution neural network method based on the global graph can not introduce the time series information contained in the text, and the graph neural network method based on the subgraph can not give different representations to the same word in different sentences. This paper proposes a text classification model based on graph neural network and fine-tuning Bert. Experimental results on multiple test data sets show that our model can learn more abundant text features. Compared with the baseline model, the accuracy is improved by 1.98%, and the F1 value is improved by 2.93%.

1. Introduction

Since the 21st century, with the rapid development of Internet technology and the surge in the number of users, the relationship between people and the Internet has become increasingly close. At present, the Internet is producing a large amount of data every day. More and more fields are constantly introducing Internet technology, and the scale of Internet data will continue to grow exponentially. As an important data form on the Internet, text data accounts for a large proportion, including news texts, government documents, film reviews, and so on [1].

The traditional deep learning model is difficult to capture the graph structure information contained in the text, which limits the effect of text classification to a certain extent. Text representation is to represent natural language text as numbers or vectors that can be understood by computers, and it is also the first step of natural language processing. A good text representation model can greatly improve the upper limit of text classification effect. In recent years, text representation methods have gradually become mature. Foreign scholars [2] proposed a BP neural language model based on multiple windows, which generated the word vector corresponding to each word in the process of model training, proving that the language training model using neural networks has better efficiency, and has become the basis of most neural network models at present. Foreign scholars have proposed the word2vec model [3], which uses the context relationship between words in the text to express words as low-dimensional vectors. Word2vec is divided into two calculation models: continuous word bag calculation model (cbow) and skip gram, which predict words through the context of words and predict the context through words [4]. Compared with previous word embedding methods, word2vec takes into account the context between words in the text, so the accuracy of word embedding will be higher. Glove model is a word vector representation tool based on global word frequency statistics. It makes full use of statistical information by using the co-occurrence matrix of words, and constructs the approximate relationship between the co-occurrence matrix and the word vector, which efficiently encodes the semantics [5].

Therefore, this paper uses graph neural network and Bert technology to classify the text, trying to mine deeper semantic structure information in the text. At the same time, it improves and optimizes the graph neural network model to try to further mine more abundant text graph structure information. On the one hand, it can provide more reliable models for the fields that need to use text classification technology, and on the other hand, it can promote the development of graph neural network in the field of text classification. Therefore, the research in this paper has certain theoretical and practical significance.

2. Overview of Related Concepts

2.1. Static Word Embedding Model

Word2vec is a word embedding learning method based on neural network language, which solves the problem that traditional text representation cannot characterize the similarity between words. Word2vec is mainly composed of two parts of models. Cbow model and skip gram model both realize the learning of text information by the model through the context relationship between words. The word embedding vector obtained by these two prediction models can-7 well represent the similarity between words, At the same time, it contains the context information of words to a certain extent [6, 7].

See the formula for details of the whole mode:

$$h = \frac{1}{c}\sum_{i=1}^{c} W_{v*N^{x_i}} \tag{1}$$

Where h refers to the hidden layer vector obtained by the hidden layer by multiplying the one hot vectors of the C words near the center word by the weight matrix W, and then adding and averaging.

Then the hidden layer vector h is multiplied by the weight matrix to get the score of each word in the vocabulary. Finally, the probability distribution of all words is obtained through the softmax layer.

$$\mu_j = W_{N*V^h}{}' \tag{2}$$

$$P\left(x_j \middle| x_1, x_2, \ldots x_{j-1}, x_{j+1}, \ldots x_c\right) = y_j = \frac{\exp(\mu_j)}{\sum_{j'=1}^{V} \exp(\mu j')} \tag{3}$$

2.2. Dynamic Word Embedding Model

In the static word embedding model, each word only corresponds to a fixed word embedding representation, which will not change with the context of the word. However, the same word may express different semantics in different contexts, which to some extent affects the performance of natural language processing tasks [8]. Elmo model is a two-way language model. Its basic idea is to first learn the word embedding representation through a large-scale corpus. At this time, the word embedding representation can not distinguish the representation of the same word in different contexts, and then use the training data to fine tune the Elmo model, and use the training data to adjust the vector representation of words in the current context. This has also become the basis of many dynamic word embedding models [9]. Elmo model uses a two-layer bidirectional LSTM based network. The layers of LSTM are connected with each other using residuals. The low-level and high-level bidirectional LSTM layers can extract syntactic information and semantic information respectively.

The GPT model uses the decoder of the transformer instead of the bidirectional LSTM of the Elmo model. Because the transformer can better capture the semantic association information over a long distance through the self attention mechanism, and can be trained in parallel through the distributed GPU, it is more advantageous than the Elmo model in the efficiency of feature extraction and calculation. However, GPT is a one-way language model, some potential semantic information existing for some context related contexts may not be well captured [10, 11].

2.3. Traditional Deep Learning Model

(1) The word embedding layer is generally initialized by the pre trained word2vec word vector and glove word vector [12].

(2) The text representation layer, the most important layer in text classification, determines the effect of text classification. It is generally implemented by convolutional neural networks, cyclic neural networks and other networks based on deep learning [13].

(3) The classification layer is generally realized by a simple fully connected neural network after the text presentation layer, and finally by the sigmoid layer or softmax layer [14].

2.4. Graph Neural Network Model

Traditional neural networks have made great progress in computer vision and natural language processing. However, traditional neural networks are oriented to data with regular spatial structure, that is, European data. However, in real life, they are faced with many non European data such as [15]: chemical and molecular structures, social networks and so on. These data contain rich graph information, but can not be directly processed using traditional neural networks [16]. Graph neural network is a neural network specially for graph structure data. It can characterize and learn the nodes and edges in the graph, and also learn the node dependencies in the graph through the information transfer between nodes in the training process. Graph neural network is similar to people's ability to infer reliable information from daily experience. It can generate graphs from unstructured data. At present, the general structure of graph neural network can be divided into four steps: pre representation of nodes and edges in the graph, selection of subgraphs, feature extraction of subgraphs, generation and training of graph neural network.

(1) Pre representation of nodes and edges in the graph: embedding the nodes and edges in the graph is equivalent to initializing the features of nodes and edges in the graph.

(2) Subgraph selection: each node and the surrounding n nodes form an n-order subgraph.

(3) Feature extraction of subgraphs: extract local or global features of subgraphs by spectral method or spatial method.

(4) Generate and train the graph neural network: define the relevant parameters of the graph neural network, and input the data into the model for training.

With the great success of graph neural network technology in social networks and link prediction, more and more researchers apply graph neural network to natural language processing. Therefore, the main goal of text classification based on graph neural network at present is to construct the best text graph and fully characterize the constructed text graph [17].

Text GCN is a semi supervised text classification model based on graph convolution neural network. Its method is the proposed classical graph based spectral domain convolution method. This method defines the convolution network structure by defining the first-order approximation of spectral convolution, which simplifies the method used in spectral convolution [18]. See the formula for details:

$$H^{(l+1)} = \sigma(\tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}H^{(l)}W^{(l)}) \tag{4}$$

3. Text Classification Model Based on Neural Network

3.1. Model Introduction

The model proposed in this chapter is improved on the basis of text GCN, and mainly consists of two parts: the pre training bilstm model and the GCN model.

Pre trained bilstm model: this layer first initializes the words in the text using the pre trained word vector to obtain the initial text features, then inputs the text features into the bidirectional LSTM to complete the specified text classification task, and finally obtains the text and word features containing time series information.

Text map construction: graph convolution neural network can extract text features by convoluting text maps. The strategy of constructing text graph used in this paper is undirected graph, which reflects the potential graph information in the text through the edges between text and words and between words.

GCN model: this model builds a large heterogeneous text map for the corpus, realizes the learning of node features in the text map through two-layer graph convolution, and converts high-dimensional sparse node features in the text map into low-dimensional dense node features. The purpose of this GCN layer is to further extract text and word features obtained from the pre trained bilstm model, so as to obtain text feature representations with stronger representation ability, and obtains the text classification result through the representation.

3.2. Experimental Environment

The experiments in this chapter are carried out under Google's colab. The accessories include a CPU with a frequency of 2.30ghz and a memory of 12.72gb. The graphics card is Tesla P100 and the memory is 16GB. The above algorithms are implemented using Python development language. The main development tool is the open source Python machine learning library pytorch. The version of pytorch is 1.7.0.

4. Numerical Analysis Results

4.1. Data Set Introduction

Table 1. Introduction of experimental data set

Dataset	Docs	Training	Test	Words	Classes	Average Lengeth
R8	7674	5485	2189	7688	8	65.72
R52	9100	6532	2568	8892	52	69.82
Ohsumed	7400	3357	4043	14157	23	135.82

As shown in Table 1, in this paper, experiments were carried out on multiple data sets to verify the effectiveness and performance of the model. A total of three data sets were used, namely R8, R52 and OHSUMED data sets.

OHSUMED is a classification data set established on the basis of a large biomedical data information base. Since MEDLINE is designed for multi label classification, this paper deletes two or more tagged texts. After processing, OHSUMED contains 23 categories.

4.2. Comparative Experiment of Model Classification Effect

Table 2. Comparison of accuracy between bilstm GCN and different models

Model	R8	R52	Ohsumed
CNN	95.92	91.99	60.19
LSTM	96.04	90.67	57.17
Bi-LSTM	96.51	92.19	60.29
Fasttext	96.27	92.75	59.23
Graph-CNN	96.62	92.85	63.54
Text-GCN	97.07	92.95	68.24
BiLSTM-GCN	97.58	93.65	68.21

Table 3. The comparison between BiLSTM-GCN and F1 of different models

Model	R8	R52	Ohsumed
CNN	84.21	54.87	43.03
LSTM	86.23	62.35	46.31
Bi-LSTM	87.13	62.58	47.68
Fasttext	82.55	64.21	48.12
Graph-CNN	93.32	63.21	59.55
Text-GCN	91.32	62.58	63.25
BiLSTM-GCN	92.58	65.89	63.21

As shown in Table 2 and Table 3, in terms of data sets, R8 and R52 are two news text classification data sets, and OHSUMED is a medical text classification data set. The performance of the model in this chapter is improved more on R8 and R52, which indicates that the model in this chapter is more suitable for news text classification tasks. Medical texts contain more professional terms, and it is difficult for the model to obtain its semantic information. Compared with other models, the semantic information learned by the model in this chapter is also relatively limited, although the performance of the model has been improved, the improvement is not significant.

4.3. Comparison of Model Parameters

As shown in Table 4, the parameter quantity of the model is related to the memory consumed by the model. The smaller the parameter quantity, the less the memory consumed by the model, and the larger the parameter quantity, the more memory consumed by the model. This experiment compares the parameters of different models. The specific comparison is shown in Table 4. First, the CNN model has the least parameters on the R8 and R52 data sets, the Bi LSTM model has the least parameters on the OHSUMED data set, and the text GCN model has the most parameters on the three data sets. Compared with the graph neural network, the traditional deep learning model has less parameters, Therefore, the traditional deep learning model has more advantages in memory consumption. Compared with text GCN, the bilstm GCN model proposed in this chapter has greatly reduced the parameters of the GCN model. The parameters on the R8 dataset have been reduced by 51.20%, the parameters on the R52 dataset have been reduced by 52.67%, and the parameters on the OHSUMED dataset have been reduced by 53.04%.

Table 4. BiLSTM-GCN and the comparison of different model's data

Model	R8	R52	Ohsumed
CNN	1295874	1552584	1974321
LSTM	1752581	1684521	1954122
Bi-LSTM	1357423	1597423	1971254
Text-GCN	2958713	3505414	4216321
BiLSTM-GCN	1235821	1695841	1954123

As shown in Figure 1 and Figure 2, generally speaking, the larger the dimension of the word vector, the more it can represent the information contained in the word, so the better the classification effect will be. In this chapter, glove pre training word vectors of 100, 200 and 300 dimensions are used for experiments on R8 and R52 data sets. As shown in Figure 1, with the improvement of word vector dimension, the accuracy and F1 value of the model have improved to a certain extent.

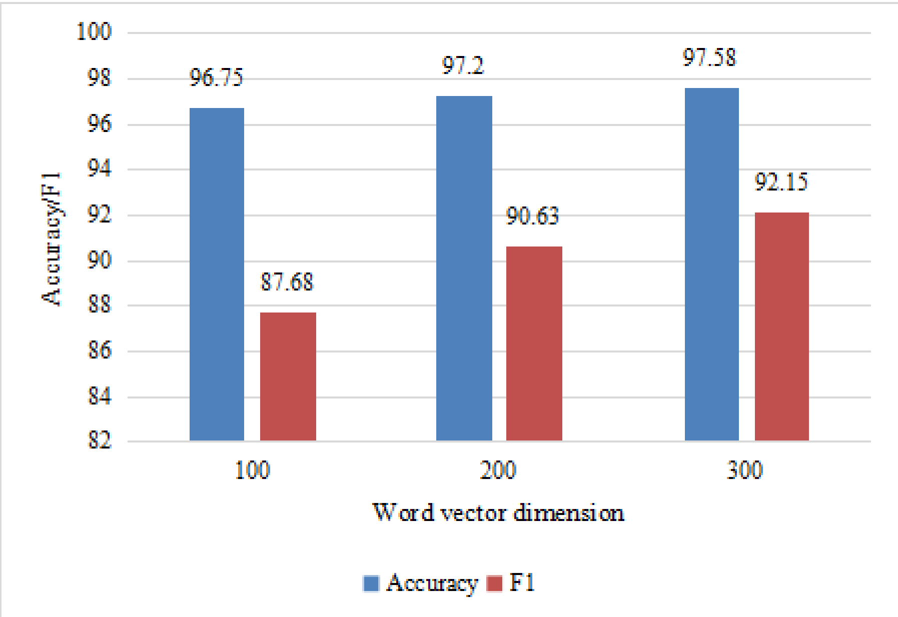

Figure 1. Comparison of accuracy and F1 value of bilstm GCN model in R8

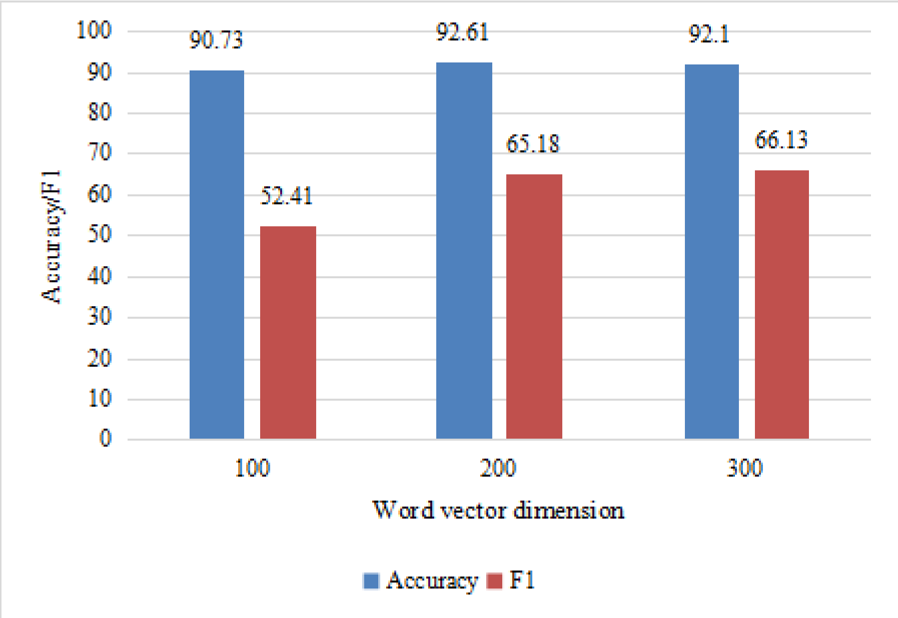

Figure 2. Comparison of accuracy and F1 value of bilstm GCN model in R52

4.4. Influence of Porter's Stemming Algorithm on Model Classification

As shown in Figure 3, porter's stemming algorithm is a stemming algorithm based on suffix stripping, which is widely used in the preprocessing of natural language texts. This algorithm can extract stemming well and help improve the performance of the model. In this chapter, a comparison experiment is set up on the R8 and R52 data sets to compare the impact of whether to use Porter's stemming algorithm on the classification effect of the model, as shown in the figure below. It can be seen from Figure 3 that the accuracy and F1 value of the model have been improved after using Porter's stemming algorithm.

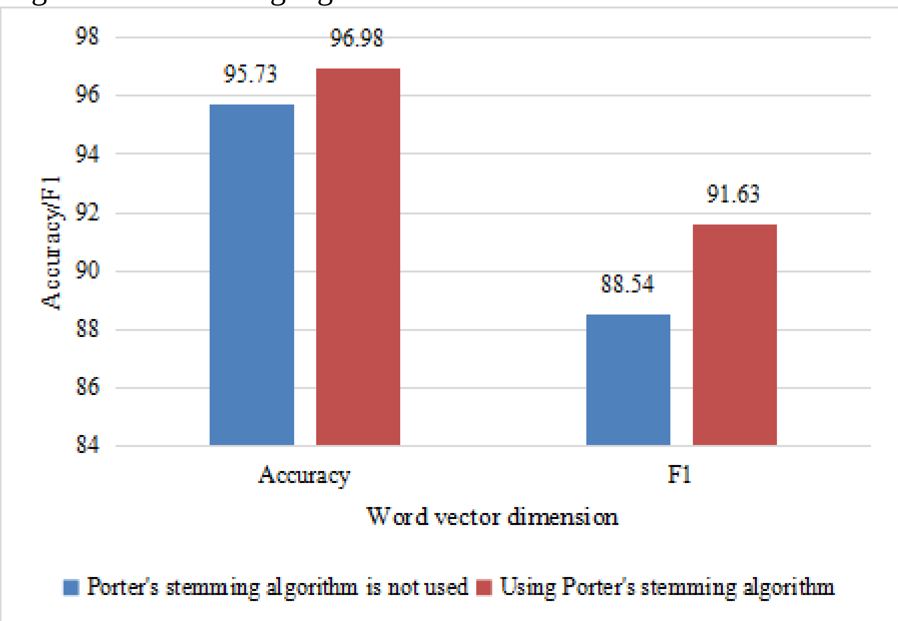

Figure 3. Comparison of accuracy and F1 value of bilstm GCN model with or without Porter's stemming algorithm in R8

5. Conclusion

For the task of text classification, a text classification model based on bilstm GCN is proposed. First, the specific method of the model is introduced. First, the words and text features containing time series are extracted through the pre trained bilstm model, and then these features are input into the two-layer GCN to get the final classification label. The model proposed in this chapter solves the disadvantage that text GCN cannot learn time series information to a certain extent. Then the experimental environment, experimental parameter settings, comparative experiments and evaluation indicators are briefly introduced. The experimental results show that, compared with the traditional deep learning text classification model and the baseline GCN model, the model proposed in this chapter has a certain improvement in the accuracy and F1 value of the three published text classification data sets, and the number of parameters is reduced compared with the baseline GCN model. At the same time, the different dimensions of word vectors and whether to use Porter's stemming algorithm also have a significant impact on our proposed model.

Funding

This article is not supported by any foundation.

Data Availability

Data sharing is not applicable to this article as no new data were created or analysed in this study.

Conflict of Interest

The author states that this article has no conflict of interest.

References

[1] Du J H. Automatic Text Classification Algorithm Based on Gauss Improved Convolutional Neural Network. Journal of Computational Science. (2017) 21(jul.):195-200. https://doi.org/10.1016/j.jocs.2017.06.010

[2] Kruthiventi S, Ayush K, Babu R V. DeepFix: A Fully Convolutional Neural Network for Predicting Human Eye Fixations. IEEE Transactions on Image Processing. (2017) 26(9):4446-4456. https://doi.org/10.1109/TIP.2017.2710620

[3] Acharya U R, Fujita H, Lih O S, et al. Automated Detection of Arrhythmias Using Different Intervals of Tachycardia ECG Segments with Convolutional Neural Network. Information Sciences. (2017) 405:81-90. https://doi.org/10.1016/j.ins.2017.04.012

[4] Mishkin D, Sergievskiy N, Matas J. Systematic Evaluation of Convolution Neural Network Advances On the Imagenet. Computer Vision and Image Understanding. (2017) 161(aug.):11-19. https://doi.org/10.1016/j.cviu.2017.05.007

[5] Kahng M, Andrews P Y, Kalro A, et al. ActiVis: Visual Exploration of Industry-Scale Deep Neural Network Models. IEEE Transactions on Visualization & Computer Graphics. (2018) (99):1-1. https://doi.org/10.1109/TVCG.2017.2744718

[6] Quan H, Srinivasan D, Khosravi A. Short-Term Load and Wind Power Forecasting Using Neural Network-Based Prediction Intervals. IEEE Transactions on Neural Networks & Learning Systems. (2017) 25(2):303-315. https://doi.org/10.1109/TNNLS.2013.2276053

[7] Hou R, Chen C, Shah M. Tube Convolutional Neural Network (T-CNN) for Action Detection in Videos[C]// IEEE Computer Society. IEEE Computer Society. (2017):5823-5832. https://doi.org/10.1109/ICCV.2017.620

[8] Chatterjee S, Sarkar S, Hore S, et al. Particle Swarm Optimization Trained Neural Network for Structural Failure Prediction of Multistoried RC Buildings. Neural Computing & Applications. (2017) 28(8):2005-2016. https://doi.org/10.1007/s00521-016-2190-2

[9] Bangalore P, Tjernberg L B. An Artificial Neural Network Approach for Early Fault Detection of Gearbox Bearings. IEEE Transactions on Smart Grid. (2017) 6(2):980-987. https://doi.org/10.1109/TSG.2014.2386305

[10] Hodo E, Bellekens X, Hamilton A, et al. Threat Analysis of Iot Networks Using Artificial Neural Network Intrusion Detection System. Tetrahedron Letters. (2017) 42(39):6865-6867.

[11] Meng H, Bianchi-Berthouze N, Deng Y, et al. Time-Delay Neural Network for Continuous Emotional Dimension Prediction from Facial Expression Sequences. IEEE Transactions on Cybernetics. (2017) 46(4):916-929. https://doi.org/10.1109/TCYB.2015.2418092

[12] Ansari G J, Shah J H, Farias M, et al. an Optimized Feature Selection Technique in Diversified Natural Scene Text For Classification Using Genetic Algorithm. Ieee Access. (2021) (99):1-1. https://doi.org/10.1109/ACCESS.2021.3071169

[13] Ahmed A. A Novel Statistical Method for Scene Classification Based on Multi-Object Categorization and Logistic Regression. Sensors. (2020) 20(14). https://doi.org/10.3390/s20143871

[14] Fadzal A N, Puteh M, Rahman N A. Ant Colony Algorithm for Text Classification in Multicore-Multithread Environment. Indonesian Journal of Electrical Engineering and Computer Science. (2020) 18(3):1359. https://doi.org/10.11591/ijeecs.v18.i3.pp1359-1366

[15] Hamid Z, Khafaji H K. A General Algorithm of Association Rule-Based Machine Learning Dedicated for Text Classification. Journal of Physics Conference Series. (2021) 1773(1):012011. https://doi.org/10.1088/1742-6596/1773/1/012011

[16] Hindi K E, Shawar B A, Aljulaidan R, et al. Improved Distance Functions for Instance-Based Text Classification. Computational Intelligence and Neuroscience. (2020) (2020) (2):1-10. https://doi.org/10.1155/2020/4717984

[17] Alsammak I, Sahib H, Itwee W H. An Enhanced Performance of K-Nearest Neighbor (K-NN) Classifier to Meet New Big Data Necessities. IOP Conference Series: Materials Science and Engineering. (2020) 928(3):032013 (15pp). https://doi.org/10.1088/1757-899X/928/3/032013

[18] Samarthrao K V, Rohokale V M. Enhancement of Email Spam Detection Using Improved Deep Learning Algorithms for Cyber Security. Journal of Computer Security, (2021) (1):1-34.