2020 International Conference on Big Data & Artificial Intelligence & Software Engineering (ICBASE) | 978-1-7281-9619-0/20/$31.00 ©2020 IEEE | DOI: 10.1109/ICBASE51474.2020.00093

Research on Mass News Classification Algorithm Based on Spark

Junyi Wang*
Information and Communication Company of State Grid Henan Electric Power Company
Zhengzhou, China
* Corresponding author: 763502997@qq.com

Fajia Ji
Information and Communication Company of State Grid Henan Electric Power Company
Zhengzhou, China
jifajia@ha.sgcc.com.cn

Boyu Liu
Information and Communication Company of State Grid Henan Electric Power Company
Zhengzhou, China
liuboyu@ha.sgcc.com.cn

Ning Wang
Information and Communication Company of State Grid Henan Electric Power Company
Zhengzhou, China
wangning2@ha.sgcc.com.cn

Hao Yin
Information and Communication Company of State Grid Henan Electric Power Company
Zhengzhou, China
yinhao1@ha.sgcc.com.cn

Feifei Zhang
Information and Communication Company of State Grid Henan Electric Power Company
Zhengzhou, China
1364427951@qq.com

***Abstract*—In recent years, with the explosion of the number of Internet news, people pay more and more attention to how to classify the mass of news. Therefore, this paper studies the mass news classification algorithm based on Spark, aiming at the problem of how to classify mass news data quickly and efficiently. In this paper, a large amount of news text is segmented based on Jieba segmentation tool, and several versions of stop words list are combined to remove stop words. Secondly, on the basis of traditional convolutional neural network, this paper proposes a news classification algorithm based on the combination of pre-trained Word2vec and improved CNN. In addition, the classification algorithm proposed in this paper is parallelized based on Spark, which improves the speed of mass news classification. In this paper, the standard data sets are used to compare and experiment the proposed news classification algorithm. The experimental results show that compared with the traditional algorithm, the news classification optimization algorithm designed in this paper has obvious improvement in multiple evaluation indexes such as accuracy, recall and F1. In addition, after parallel design of the algorithm proposed in this paper based on Spark, compared with the serial algorithm, the speed improvement effect is also more significant.**

Keywords- Spark; CNN; Classification; Word2Vec; Jieba

I. INTRODUCTION

In recent years, with the rapid development of the Internet, mass news on the Internet has become the primary source of information for people. However, due to the explosive growth of network news, people pay more and more attention to how to classify massive news quickly and accurately. Therefore, how to classify massive news has become the focus of academic researchers in various professional fields.

The rapid growth of Internet news has brought great difficulties to the classification of massive data. Some classical classification methods have not been able to meet the needs of the classification of massive data, so the classification of large-scale news data has become an important factor affecting the development of science and technology and society. Because Spark platform has efficient computing and processing capacity for massive data [1], mass news classification technology based on Spark arises at the historic moment. The birth of Spark alleviates many problems existing in the classification of mass news data. Therefore, applying Spark to the classification task of mass news, based on its advantages in large-scale data processing, can effectively alleviate the current overload of Internet news.

Therefore, in order to cope with the explosive growth of news and help people classify mass news quickly and effectively, this paper studies the mass news classification algorithm based on Spark.

II. RELATED WORKS

This paper makes an in-depth study of the theories and technologies involved in the mass news classification algorithm based on Spark and its research status at home and abroad.

In recent years, with the continuous development of Chinese word segmentation technology, a lot of word segmentation tools have been produced which are fully packaged and can be used directly. Among them, the most common ones mainly include THULAC, SnowNLP and Jieba. Jieba is the most widely used Chinese word segmentation software.

Nowadays, text models commonly used mainly include language model, topic model and vector space model [2].

978-1-7281-9619-0/20/$31.00 ©2020 IEEE
DOI 10.1109/ICBASE51474.2020.00093

Among these models, vector space model is the first one to be used. As technology evolves, academic researchers are constantly trying to use better models to represent text. Among them, Bun et al. [3] proposed a text vectorization algorithm based on TF-IDF, and the experimental results showed that the text representation method based on TF-IDF was more effective in news analysis. David (2003) [4] proposed the LDA topic model, which represents text as the distribution of topics. Naili et al. [5] conducted in-depth research on Word2vec based modeling method, and the research showed that compared with traditional modeling method, Word2vec based model representation effect is better.

At present, common classification algorithms include NB [6], KNN [7], SVM [8], etc. Among these classification algorithms, many have low complexity and simple structure. For small-scale data sets, classification effect is relatively good. In 2006, Hinton et al. brought the concept of deep learning into the public eye [9]. Among them, recursive neural network (RNN), long and short term memory network (LSTM) and convolutional neural network (CNN) [10] are several common deep learning algorithms at present.

Douglas et al. developed Hadoop computing framework on the basis of GFS and MapReduce [11]. In 2010, AMP Laboratory of University of California, Berkeley developed a parallel computing framework Spark [12], which has most of the advantages of traditional Hadoop MapReduce. Moreover, in large-scale data processing and analysis tasks, Spark performs better in algorithm running efficiency than Hadoop MapReduce.

In order to improve the effect and speed of mass news classification, this paper studies the mass news classification algorithm based on Spark.

III. News Classification Algorithm Based on the Combination of Word2Vec and CNN

In this section, a new classification algorithm based on the combination of pre-trained Word2vec and improved CNN is proposed.

A. Model design

The classification model structure designed in this paper is shown in Fig. 1.

1) Input layer design

For text data, the preprocessed text needs to be transformed into numerical data, so it is necessary to add an embedding layer in convolutional neural network. Embedding layer is the embedding layer of word vector. However, if embedding layer is embedding into the convolutional neural network for self-training, problems such as large number of training parameters and long training time will occur, and this may lead to the phenomenon of over-fitting of the model. Therefore, this paper embedding the trained Word2vec vector into the embedding layer, so as to reduce the training parameters and time consumption of the network model.

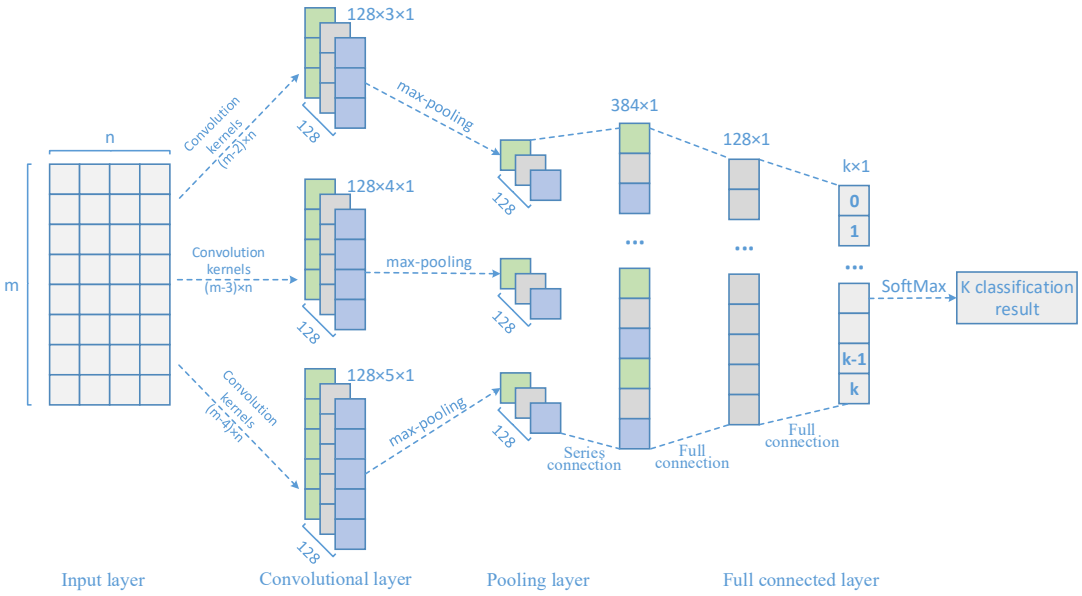

Figure 1. Improved convolutional neural network model diagram

Assuming that there are altogether d documents in the document data set, each document needs to be converted into an $m \times n$ matrix. m represents the rows in the two-dimensional matrix, and the mean value of the number of words in all the pre-processed documents is used as the value of m. If the number of words in the document is less than m, then 0 vector is used to complete it. If the number of words in the document is greater than m, then m words in the document are randomly selected. n is the vector dimension of the word. Therefore, the input layer in the CNN model is transformed into a two-dimensional feature matrix of $m \times n$. Compared with embedding layer's self-training, problems such as long training time, multiple training parameters and easy over-fitting of the model can be alleviated.

2) Convolutional layer design

The design of the convolutional layer of this model is different from that of traditional CNN. The convolutional layer of traditional CNN uses the convolution kernel of the same size for convolution operation to extract data features and obtain a set of feature planes. The convolutional layer designed in this model performs convolution operations of three convolution kernels of different sizes on the input layer matrix respectively to obtain three sets of feature planes, which are placed in parallel, as shown in the Fig. 1. These three convolution kernels of different sizes are: *(m-2)×n*, *(m-3)×n*, and *(m-4)×n*, each with 128 convolution kernels. Therefore, the size of three groups of feature planes obtained by convolution operation with three convolution kernels of different sizes are 3×1, 4×1 and 5×1 respectively, among which 128 are in each group of feature planes.

3) Pooling layer design

As shown in the Fig. 1, the pooling layer samples the feature plane of the convolutional layer. The size of the sampling window is 3×1, 4×1 and 5×1 respectively. Three groups of 128 sampling planes were obtained, and the size of each sampling plane was 1×1 (only containing one neuron). Then, the three sets of sampling planes were connected in series to obtain a 384×1 sampling plane (namely 384 neurons).

In order to extract the best features of each feature plane, the max-pooling method is adopted to complete the sampling of the feature plane of the convolutional layer.

4) Full connected layer design

The first layer was fully connected with 128 neurons, while the second layer was fully connected with *k* neurons. Softmax was used as the output layer to complete *k* classification and obtain classification results.

Because the CNN model designed in this paper may have overfitting problems, this paper uses Dropout optimization strategy [13] to optimize the model, so as to suppress the occurrence of overfitting phenomenon. The core idea of this strategy is to break the connections between certain neurons in the hidden layer with a certain probability. In other words, these connections are temporarily disabled to simplify the complexity of the neural network model and avoid the overfitting of the model.

B. Parameter optimization

Gradient descent (GD) is a common method for parameter optimization of neural network models. According to the size of the text data set used in each iteration, gradient descent can be divided into three algorithms: stochastic gradient descent (SGD), batch gradient descent (BGD) and small batch gradient descent (MBGD).

Among the three algorithms, SGD is faster but less accurate. On the contrary, BGD and SGD have high accuracy, but slow training speed. MBGD combines SGD and BGD algorithms to effectively avoid their disadvantages.

Faced with the problem of setting learning rate, many optimization strategies based on MBDG algorithm emerge at the right moment. In order to improve the execution efficiency of CNN model, this paper chooses Adam [14] optimization strategy as the parameter optimization strategy of CNN.

The Adam optimization strategy combines momentum and Adadelta optimization strategy, and the parameter update mode of this optimization strategy is shown in (1).

$$\begin{cases} s_i = \gamma_1 s_{i-1} + (1-\gamma_1) g_i \\ r_i = \gamma_2 r_{i-1} + (1-\gamma_2) {g_i}^2 \\ \hat{s}_i = s_i / \left(1-{\gamma_1}^i\right) \\ \hat{r}_i = r_i / \left(1-{\gamma_2}^i\right) \\ \theta_i = \theta_{i-1} - \alpha \times \hat{s}_i / \left(\sqrt{\hat{r}_i} + \varepsilon\right) \end{cases} \quad (1)$$

Compared with other optimization algorithms, Adam has the advantages of easy implementation, low memory requirement and high computing efficiency. Most importantly, it is suitable for applications with large data and parameter scales. Because the effect of Adam optimization strategy is better than other strategies in many scenarios, this paper uses Adam strategy to optimize the parameters in the model training process, and takes Xavier as the weight parameter initialization method of the model.

C. Model training

The training of CNN can be divided into two processes: forward operation and reverse operation.

1) Forward operation

The input data is passed from the input layer to the output layer layer by layer, and finally the classification results are obtained. As shown in (2).

$$y_{pre} = f_n(w_n \cdots f_2(w_2 \times f_1(w_1 \times x + b_1) + b_2) \cdots + b_n) \quad (2)$$

Where *f* represents the activation function, *w* represents the weight value, and *b* represents the bias.

2) Reverse operation

Reverse operation refers to the calculation of the loss function based on the actual tag value and the prediction results obtained by passing the input data from the input layer to the output layer. Then the error is corrected according to the value of the loss function and the optimization algorithm of the network model. The error signal is reversely transferred from the output layer of the model to the input layer of the model, and the parameter gradient of each layer in the neural network model is calculated.

In this paper, it is assumed that the objective function is defined as cross entropy cost function, as shown in (3).

$$J(\theta) = -\frac{1}{n}\sum_{i=1}^{n}\begin{bmatrix} y^{(i)} \log\left(h_\theta\left(x^{(i)}\right)\right) + \\ (1-y^{(i)}) \log\left(1-h_\theta\left(x^{(i)}\right)\right) \end{bmatrix} \quad (3)$$

Where, y is the expected output value, and its value range is 1,2,3,…,*k*, that is, there are *k* types of output, *x* is the input value, and h is the expected output value.

Softmax regression functions are generally used as the last layer to complete the classification operation. The hypothesis function h of Softmax is shown in (4).

$$h_\theta(x^{(i)}) = \begin{bmatrix} p(y^{(i)}=1|x^{(i)};\theta) \\ p(y^{(i)}=2|x^{(i)};\theta) \\ \cdots \\ p(y^{(i)}=k|x^{(i)};\theta) \end{bmatrix} = \frac{1}{\sum_{j=1}^{k} e^{\theta_j^T x^{(i)}}} \begin{bmatrix} e^{\theta_1^T x^{(i)}} \\ e^{\theta_2^T x^{(i)}} \\ \cdots \\ e^{\theta_k^T x^{(i)}} \end{bmatrix} \quad (4)$$

Equation (4) is substituted into (3), and the loss function obtained is shown in (5).

$$\begin{aligned} J(\theta) &= -\frac{1}{n}\left[\sum_{i=1}^{n}\sum_{j=1}^{k} 1\{y^{(i)}=j\} log p(y^{(i)}=j|x^{(i)},\theta)\right] \\ &= -\frac{1}{n}\left[\sum_{i=1}^{n}\sum_{j=1}^{k} 1\{y^{(i)}=j\} log \frac{e^{\theta_j^T x^{(i)}}}{\sum_{l=1}^{k} e^{\theta_l^T x^{(i)}}}\right] \end{aligned} \quad (5)$$

As in (5), the value of 1{expression is true}=1, the value of 1{expression is false}=0.

The gradient formula can be obtained by taking the derivative of (5), as shown in (6).

$$\nabla_{\theta_j} J(\theta) = -\frac{1}{n}\sum_{i=1}^{n}\left[x^{(i)}\left(1\{y^{(i)}=j\} - \frac{e^{\theta_j^T x^{(i)}}}{\sum_{l=1}^{k} e^{\theta_l^T x^{(i)}}}\right)\right] \quad (6)$$

The classification model designed in this paper is trained iteratively. When the set threshold is met or the specified number of iterations is reached, the neural network model training is completed.

IV. News Classification Algorithm Combined by Word2Vec and CNN Based on Spark

In order to speed up the training speed of the classification model designed in this paper, this section carries out parallel design of the classification model with the help of the high-speed computing capability of Spark bigdata platform.

The parallel design flow of the training phase is shown in Fig. 2. First, the model training data set is uniformly divided into n data slices. Then the initialization model parameters are broadcast and distributed to each node in the Spark cluster. In addition, each data slice is also sent to each node for a Map operation. During the Map phase, all the nodes in Spark have an identical classification model. For each working node, data slices should be input into the classification model for training. After forward propagation and error back propagation, the weights and bias parameters of the classification model are obtained. After the model training of each node is completed, the network model parameter results obtained after the training of all nodes need to be summarized in the Reduce phase to obtain the average model parameter value. Finally, the average model parameter is used as the new model parameter of the classification model, and the new model parameter is broadcast and distributed to each work node for training again until the set maximum iteration number or the set error threshold is reached.

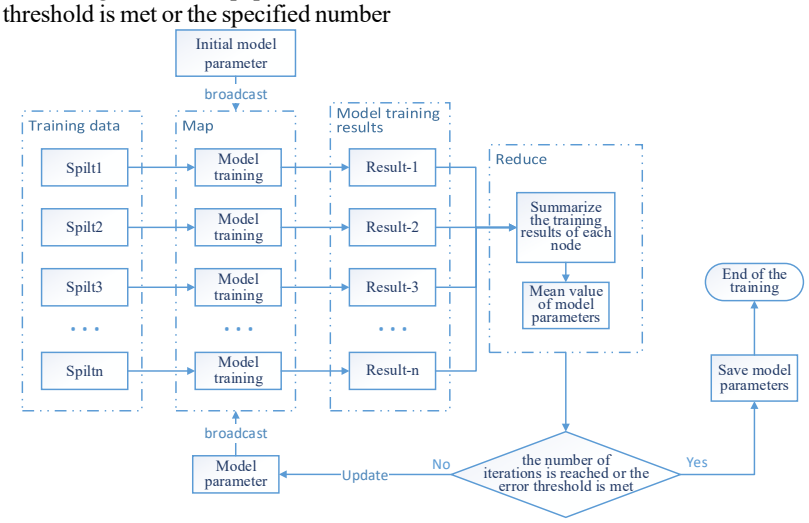

Figure 2. Parallel design flow chart for training stage of classification model based on Spark

The parallel design flow of the test stage is shown in Fig. 3. It can be seen from the figure that the parallel testing process of the classification model designed in this paper is similar to the training stage. First, the input data is also divided into m identical data slices. Then, the trained model parameters are broadcast and distributed to each work node, and all data slices is distributed to each node in the Spark cluster for Map operation. In the Map phase, all nodes have a trained news classification model, and the task of each working node is to input test data into the classification model for testing. Finally, after the Map operation is completed, the test results on all nodes will be summarized in the Reduce phase to obtain the final model test results.

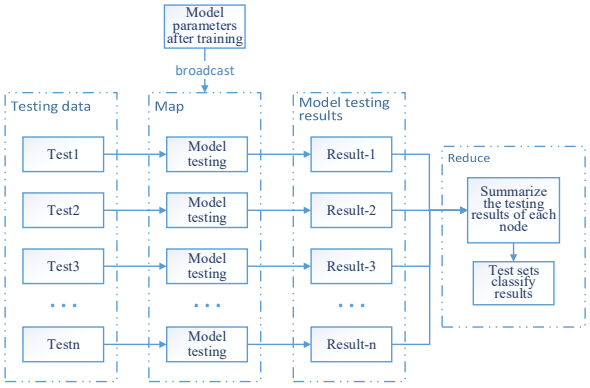

Figure 3. Parallel design flow chart for testing stage of classification model based on Spark

V. EXPERIMENT

A. DataSet

In order to verify the effect and efficiency of the news classification algorithm designed in this paper, this section selects some data from THUCNews data set of Tsinghua University for experiment. The selected data set includes 10 categories, including sports, entertainment, home, real estate, education, fashion, current affairs, games, technology and finance, and 13,200 pieces of news are selected for each category, totaling 132,000 pieces of news.

B. Experimental Design

Experiments in this paper are all run on 4-node Spark cluster consisting of 4 virtual machines, and serial algorithm experiments are all run on Slave1 node. Before starting the experiment, first use the Jieba word segmentation tool for word segmentation. Secondly, several versions of the stop words list are combined to remove the stop words from the data set. Finally, the news data set is trained based on Word2Vec model, and the word vector dimension is set as 128-dimension in this experiment.

1) Effect comparative experiment of news classification algorithm:

The experimental data are divided into training sets and test sets in a ratio of 5:1. In order to verify the effect of the classification algorithm, the classification model designed in this paper was compared with the traditional CNN classification model, and the average value of multiple results was taken as the final result. In this experiment, accuracy rate, recall rate and F1 value were used as evaluation indexes of news classification effect.

2) Accelerated comparative experiment of news classification algorithm:

This experiment records the running time of the news classification algorithm based on Spark and the running time of the serial classification algorithm, and then calculates the acceleration ratio between the parallelization algorithm and the serial algorithm, so as to compare the performance of the two algorithms. The experiment takes the average of each running time as the final result, and then calculates the acceleration ratio of serial and parallel algorithms. In this experiment, the running time and acceleration ratio of serial and parallel algorithms are used as evaluation indexes.

C. Experimental Results and Analysis

1) Effect comparative experiment of news classification algorithm

The specific results obtained by the effect comparative experiment are shown in Table I.

TABLE I. EFFECT COMPARATIVE EXPERIMENT RESULTS

Category	Traditional CNN			Pre-trained Word2vec + improved CNN		
	Accuracy rate	*Recall rate*	*F1*	*Accuracy rate*	*Recall rate*	*F1*
Sports	0.984	0.993	0.988	0.996	0.997	0.996
Entertainment	0.976	0.971	0.973	0.995	0.964	0.979
Home	0.864	0.978	0.917	0.978	0.933	0.955
Estate	0.991	0.995	0.993	0.981	0.979	0.980
Education	0.924	0.939	0.931	0.966	0.971	0.968
Fashion	0.982	0.927	0.954	0.948	0.991	0.969
Current affairs	0.934	0.949	0.941	0.987	0.951	0.969
Games	0.974	0.979	0.976	0.969	0.993	0.981
Technology	0.961	0.988	0.974	0.964	0.974	0.969
Finance	0.992	0.879	0.932	0.971	0.994	0.982
Total	0.9582	0.9598	0.9579	0.9755	0.9747	0.9748

The comparison results of News classification algorithm on F1 values are shown in Fig. 4.

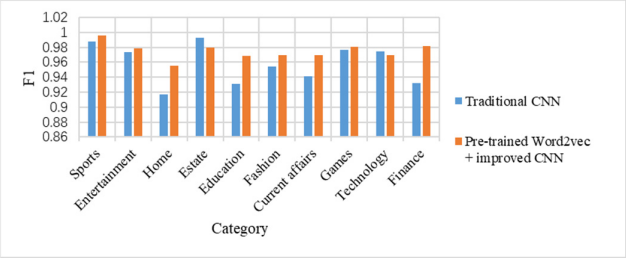

Figure 4. The comparison results of News classification algorithm on F1 values

The overall graphical comparison results of different news classification algorithms are shown in Fig. 5.

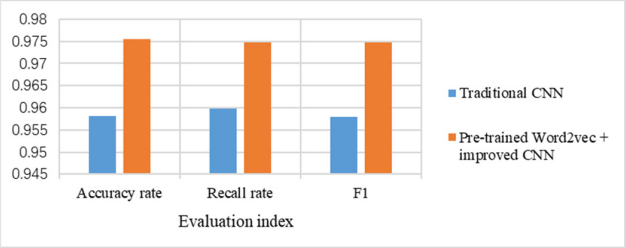

Figure 5. The overall graphical comparison results of different news classification algorithms

It can be seen from Table I and Fig. 4 that, from the results of each category, the classification algorithm designed in this paper has improved the accuracy, recall rate and F1 value in most categories. In addition, it can be seen from Table I and Fig. 5 that, from the overall classification results, the accuracy and recall rates of the classification model designed in this paper reach 0.9755 and 0.9747 respectively. Compared with the traditional CNN model, the accuracy rate and recall rate are improved by 1.73% and 1.49% respectively. Most importantly, compared with the traditional CNN model, the F1 value of the CNN model designed in this paper increased from 0.9579 to 0.9748, with an increase of 1.69%. It can be seen that the classification algorithm designed in this paper has significantly improved the accuracy rate, recall rate, F1 value and other evaluation indexes.

2) Accelerated comparative experiment of news classification algorithm

The results of running time and acceleration ratio obtained by the accelerated comparison experiment of news classification algorithm based on Spark are shown in Fig. 6 and Fig. 7.

As can be seen from Fig. 6 and Fig. 7, when the experimental data scale is relatively small, the time required for the news classification algorithm based on Spark and the serial CNN classification algorithm to complete the operation is not significantly different. With the increase of experimental data scale, the running time of serial algorithm increases more than that of parallel algorithm, so the acceleration ratio of the two algorithms increases. However, with the continuous increase of the data size, the acceleration ratio of the two becomes smaller. This is mainly because when the data grows to a certain scale, the acceleration ratio of the algorithm will gradually reach its peak, and the time consumed in the process of data transmission and communication between clusters will increase continuously, so the acceleration ratio will be reduced to some extent. Even so, the parallel algorithm maintains a good acceleration ratio compared with the serial algorithm, and has a good acceleration effect all the time. Therefore, the news classification algorithm based on Spark designed in this paper can improve the speed of mass news classification to some extent.

Therefore, through two groups of comparative experiments, it can be seen that the news classification algorithm based on Spark designed in this paper has been significantly improved in both algorithm effect and speed.

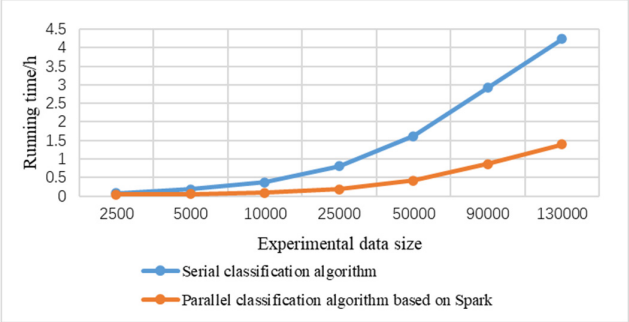

Figure 6. Comparison results of running time of classification algorithm

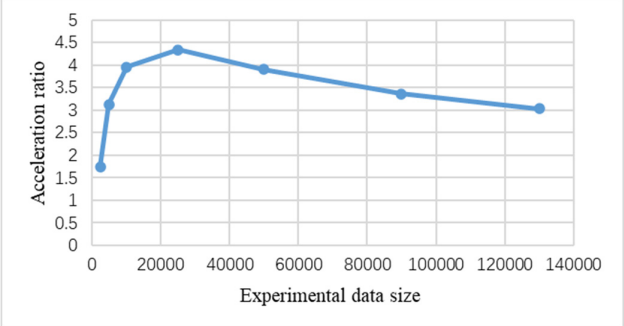

Figure 7. Results of acceleration ratio of classification algorithm

VI. Conclusiong

With the explosion of the number of Internet news, people pay more and more attention to how to classify the mass of news. Therefore, this paper studies the mass news classification algorithm based on Spark, aiming at the problem of how to classify mass news data quickly and efficiently.

In this paper, a large amount of news text is segmented based on Jieba segmentation tool, and several versions of stop words list are combined to remove stop words. Secondly, on the basis of traditional convolutional neural network, this paper proposes a news classification algorithm based on the combination of pre-trained Word2vec and improved CNN. This algorithm imbedded the pre-trained Word2vec into the input layer of the improved convolutional neural network. The experimental results show that this algorithm not only improves the classification effect, but also reduces the parameters and time of model training. In addition, the classification algorithm proposed in this paper is parallelized based on Spark, which improves the speed of mass news classification.

This paper has made some achievements in the research on mass news classification algorithm based on Spark, which has been significantly improved in both algorithm effect and processing speed. However, due to the limited time, the research in this paper is not in-depth enough, and further optimization and improvement are needed in the aspects of pre-processing operation, algorithm effect, calculation speed and so on.

REFERENCES

[1] Meng X, Bradley J, Yavuz B, et al. MLlib: machine learning in apache spark[J]. Journal of Machine Learning Research, 2015, 17(1):1235-1241.

[2] Salton G, Wong A, Yang C S. A Vector space model for automatic indexing[J]. Communications of the ACM, 1975, 18(11):613-620.

[3] Bun K K, Ishizuka M. Topic extraction from news archive using TF*EDF algorithm[C]// International Conference on Web Information Systems Engineering. IEEE, 2003.

[4] Blei D M, Ng A Y, Jordan M I, et al. Latent Dirichlet Allocation[J]. Journal of Machine Learning Research, 2003, 3:993-1022.

[5] Naili M, Chaibi A H, Ben Ghezala H H. Comparative study of word embedding methods in topic segmentation[J]. Procedia Computer Science, 2017, 112:340-349.

[6] Li J, Sun L, Zhang Q, et al. A Naive Bayes classifier in Text Processing [J]. Journal of Harbin Engineering University, 2003(01):74-77.

[7] Zhou Z. Machine Learning [M]. Beijing: Tsinghua University Press, 2016.

[8] Zhang A, Liu G, Liu C. Research on Multi-Class Text Classification based on SVM [J]. Journal of Intelligence, 2004(09):7-8+11.

[9] Hinton G E, Osindero S, Teh Y W. A Fast Learning Algorithm for Deep Belief Nets[J]. Neural Computation, 2006, 18(7):1527-1554.

[10] Liu J, Liu Y, Luo X. Research Progress of Deep learning [J]. Computer Application Research, 2014(07):7-16+28.

[11] Eadline D. Hadoop 2 Quick-Start Guide: Learn the Essentials of Big Data Computing in the Apache Hadoop 2 Ecosystem[M]. 2015.

[12] Zaharia M, Chowdhury M, Franklin M J, et al. Spark: cluster computing with working sets[C]// Usenix Conference on Hot Topics in Cloud Computing. 2010.

[13] Hinton G E, Srivastava N, Krizhevsky A, et al. Improving neural networks by preventing co-adaptation of feature detectors [J]. Computer Science, 2012, 3(04):212-223.

[14] Kingma D, Ba J. Adam: A Method for Stochastic Optimization [J]. Computer Science, 2014:1-15.