Spark Parallel Optimization Algorithm Based on Improved BP Neural Network

Yaru Ge*  
1School of Software, Zhengzhou University, Zhengzhou, Henan, 450000, China  
*Corresponding author’s e-mail: geyaru@stu.zzu.edu.cn

Abstract. In view of the slow convergence rate and the local minimum problem of BP neural network when it is applied to big data classification, this paper proposes a spark parallel optimization algorithm based on Improved BP neural network. On the basis of MapReduce cloud computing platform, the parallel acceleration is realized by segmenting data sets, and the parallel data is processed by combining the Bagging algorithm, so that the network data of each node can complete the training independently until the convergence is realized, and finally the set becomes the data parallel classifier. The experiment results show that the parallel optimization algorithm proposed in this paper can effectively improve the parallel data processing speed, the classification accuracy is high, and has good adaptability in spark environment.

1. Introduction
BP neural network is widely used in the field of big data processing, such as data mining, data classification [1-2]. However, with the explosive growth of data sets, it is very important to improve the training speed of BP neural network when it is applied to large-scale data sets.

With the increasing popularity of big data platform Hadoop in recent years, the parallelization of BP neural network based on Hadoop platform has been greatly developed. In reference [3], in the map stage, the neural network on each node is locally updated one by one according to the local data, and after obtaining multiple neural network structures with local convergence, the neural network with the best classification effect in map stage is used as the classifier or as the initial state of the next iteration, realizing parallel optimization to some extent, but it still needs to read the disk many times, and the final data classification accuracy of BP neural network is not satisfactory. In this paper, we use the same memory based spark platform based on MapReduce to implement the method of reference [3], and at the end of the method of reference [3], we use different strategies: after local iteration, all the obtained neural network structures are integrated into a classifier for prediction by using Bagging algorithm. The experiment results show that the performance of method we proposed is better.

2. Theoretical Basis of Algorithm

2.1. Improved BP Neural Network
The basic idea of BP neural network is to repeatedly use chain rule to calculate the impact of each network weight on the final result error of the network. When the partial derivatives of all weights are obtained, the connected weights can be updated. In this paper, the elastic neural network algorithm is used to improve the BP neural network, so as to improve the convergence speed of the network. When the sign of partial derivative of weight is changed, it indicates that the last update amount is too large, skipping a local maximum point, so the update value of weight is multiplied by a number less than 1,
reducing the update amount of weight, and then converging to the maximum point. After the partial derivative of the network error relative to the ownership value is calculated by the chain rule and the gradient descent algorithm, the updating direction of each weight is determined according to the sign of the partial derivative.

2.2. Spark
Spark is an open-source cluster computing framework similar to Hadoop based on MapReduce. The advantages of Hadoop are different: Hadoop saves the intermediate results generated by jobs in HDFS (Hadoop distributed file system) after calculating each job. For iterative algorithm design, Hadoop often needs to repeatedly read the intermediate data saved in HDFS, while spark, as a more refined design of MapReduce, is based on memory computing, and each of them is calculated based on memory. The intermediate results are stored in memory as much as possible, so there is no need to read and write HDFS repeatedly, so spark can be better used in data mining, machine learning and other iterative algorithms [4-5].

2.3. Bagging Algorithm
Bagging algorithm is a commonly used algorithm to optimize the classification effect of classifiers. It can improve the classification performance by combining multiple classifiers with better classification effect than random classification to form a final classifier. Because of the parallel BP algorithm based on spark proposed in this paper, each classifier is obtained by training part of the data in the training set, and the classification performance of each classifier is not very high, which belongs to a weak classifier. Therefore, the classification performance of the classifier can be improved through the combination idea of Bagging [6].

3. Floww chart of parallel Optimization Algorithm
Spark adopts a typical master worker parallel mode, which generally has one master node and multiple worker nodes, that is, computing nodes. The master node coordinates the execution of the whole training task, while the actual performers of training are many worker nodes. Before running, the training set is divided into many blocks and stored on the distributed file system HDFS. Each calculation node reads its own part of the block according to the data locality and trains its own BP network structure independently. The specific algorithm flow chart is shown in Figure 1.

Figure 1. Flow chart of parallel optimization algorithm
4. Experiment Verification
The cluster environment configuration properties used in the experiment are: the number of nodes is 5, the cores of each node are 7, the spark version is 2.3.2, the spark mode is standalone, the JDK version is 1.8.0, the node CPU is Intel Xeon e3-1245 CPU 3.3GHz, and the node memory is 10GB. The BP network is trained by one processor and multiple processors respectively, and the training time and other results are counted. The majority voting method is selected for combination classification, and the algorithm proposed in reference [3] is used for combination classification, the results are shown in Figure 2. The number of selected iterations is 1000.

![Figure 2. Experiment results](image)

It can be seen from Figure 2(a) that the speedup ratio of the algorithm in spark environment is higher than that in Hadoop environment, which shows that the memory based spark platform is more suitable for iterative operation than the disk based Hadoop platform. At the same time, both of them obtain considerable speedup ratio, which proves the feasibility of the algorithm. Under the same experimental conditions, the BP spark algorithm is compared with the algorithm proposed in reference [3], and the accuracy results are shown in Figure 2(b). With the increase of the number of nodes, more and more base classifiers are combined in the algorithm proposed in this paper, and the accuracy is further improved. However, the algorithm proposed in reference [5] can not reflect the distribution of the whole data set well due to the reduction of the local data amount, and the accuracy is on the contrary decreasing trend. Finally, with 5 processors selected, the data volume of the training set increases gradually. The experimental results of processing time and data volume are shown in Figure 2(c). It can be seen from Figure 2(c) that the training time of the model is basically linear with the amount of training set data, which shows that the parallel optimization algorithm proposed in this paper has good scalability.

5. Conclusion
BP network has been widely used in a large number of practical projects for its good data processing performance. But the traditional BP algorithm has some disadvantages such as slow learning speed. This paper analyzes the limitations of BP algorithm, improves BP algorithm on the basis of parallel, and greatly improves the convergence speed of the network. However, there are still some shortcomings, such as how to select the most appropriate initialization parameters, which need further research and improvement.

Acknowledgments
This work was supported by Zhengzhou University.

References
[1] Yang, A., Zhuansun, Y. and Li, J., et al. (2019) Design of Intrusion Detection System for Internet of Things Based on Improved BP Neural Network. IEEE ACCESS., 7: 106043-106052.
[2] Wu, W., Feng, G. and Li, Z., et al. (2005) Deterministic convergence of an online gradient method for BP neural networks. IEEE Transactions on Neural Networks., 16: 533-540.

[3] Liu, M., Feng, G. and Zhou, J., et al. (2019) Joint Two-Tier Network Function Parallelization on Multicore Platform. IEEE Transactions on Network and Service Management., 16: 990-1004.

[4] Chen, J., Li, K. and Tang, Z, et al. (2017) A Parallel Random Forest Algorithm for Big Data in a Spark Cloud Computing Environment. IEEE Transactions on Parallel & Distributed Systems., 28: 919-933.

[5] Yan, Y., Chen, M. and Sadiq, S., et al. (2017) Efficient Imbalanced Multimedia Concept Retrieval by Deep Learning on Spark Clusters. International Journal of Multimedia Data Engineering & Management., 8: 1-20.

[6] Kurogi, S., Toidani, M.and Shigematsu, R., et al. (2018) Performance improvement via bagging in probabilistic prediction of chaotic time series using similarity of attractors and loocv predictable horizon. Neural Computing & Applications., 29: 341-349.