Random forest algorithm in big data environment

Yingchun Liu *

School of Economics and Management, Beihang University, Beijing 100191, China

Received 1 September 2014, www.cmnt.lv

Abstract

Random forest method is one of the most widely applied classification algorithms at present. From the actual big data scene and requirements, the application of random forest method in the big data environment to conduct in-depth study. Due to the big data needs to process a huge number of features at the same time, and the data pattern changes constantly over time, the accuracy of a random forest algorithm without self-renewal and adaptive algorithm will gradually reduce over time. Aiming at this problem, analysis on the characteristics of random forest method, presents how to realize the self-adaptation ability with random forest method in similar situations, and verified the feasibility of the new method of using the actual data, and analysis and discussion of how to further research and improve the random forest method in big data environment.

Keywords: decision tree, random forest, big data

1 Introduction

Owing to the enough accumulated data over the years in this sector, big data has gained many practical application scenarios. The whole range from vast amounts of information on the Internet to supermarket shopping bills contains significant commercial value. Rapid growth in the amount of data has overstepped the bearing capacity of traditional data analysis, which accelerates the urgency in development of big data analysis tools suitable for various application areas.

Typical features of big data include huge data amount, numerous data types, high requirement for processing speed and high analysis value return. From the perspective of application scenarios, the demands for large data analytics mainly concentrate in several major categories, such as classifier, association rules and clustering [1]. Classifier technology is one focus of data mining research, and the famous classification algorithms covers association rules [2], Bayes [3], decision trees [4], neural networks, rule learning, K-means, genetic algorithms, rough sets, fuzzy logic [5, 6] and other directions.

In the scenarios of big data, the algorithm complexity caused by data amount will, however, rapidly increase, thus rejecting applicability of abovementioned classification algorithms in dealing with massive data. Common classification methods for massive data include decision tree algorithms like SPRINT [7] and BOAT, naive Bayesian algorithm, k-nearest neighbor algorithm, and classification algorithms based on association rules discovery, etc.

As a common method of data mining, Random forest method [8] has been proved to be a state-of-the-art of learning model, which not only have well classification, regression, performance and fast and efficient operations, and random forest can effectively handle multiple classification problems, also has obvious advantage in dealing with the noise. Random forest method that is not subject to memory limitations and featured with rapid processing speed and good parallel scalability, is an excellent classification tool to handle massive data and a typical decision tree classification algorithm.

These papers propose a self-adaptive random forest method. In this method, the trees in the random forest are not changeless but constantly updated by pruning bad trees and add more accurate trees. According to the comparative experiment on testing data sets, the new method has higher classification accuracy than the traditional random forest. The new method is more suitable for current big data scene in which data pattern will gradually change with time.

2 Analyses by random forest method

2.1 INTRODUCTION OF RANDOM FOREST METHOD

Random forest method is a combination classification method proposed by Breiman in 2001. Using bagging method, random forest method will draw multiple training sample sets that are different from each other. Every single sample set builds a decision tree with randomly selected attributes [9].

Random forest uses CART algorithm for building trees. Considering the large number of built trees, random forest method is characterized with good ability to resist noise and outstanding performance in the classification capability.

Random forest method is defined as a set of decision trees $\{h(x,\theta_k),k=1,...\}$, where $h(x,\theta_k)$ is a meta-classifier, namely, a unpruned decision tree created using CART

* Corresponding author's Email:lyc1031@yeah.net

algorithm; *x* serves as the input vector, while $\{\theta_k\}$ is an independent and identically distributed random vector. They determine the growth process of each decision tree. According to the input data, each decision tree will give a result; integration of multiple results will give the final output of a random forest.

In the random forest, the growth process of a single decision tree is as follows:

1. For the original training sets, bagging method is used to select random data with replacement and thus form training sets with difference.
2. The features are also selected using sampling approach. If it is assumed that a data set has N features, then M features will be sampled from N, where $M<<N$. For each extracted training set, only randomly selected M features rather than all N features will be used for node splitting in building trees.
3. All built decision trees will grow freely without pruning.

The final result can be integrated using simple majority voting method (for classification problems) or mathematical average method (for regression problems) performed among the results of the decision trees.

2.2 ADVANTAGES OF RANDOM FOREST METHOD

Random forest method can be deemed as a combining classifier algorithm or a combination of decision trees. It combines the advantages of bagging and random feature selection [10]:

1. Bagging can estimate not only the importance of each feature but also generalization error;
2. The trees of random forest method are built using CART algorithm, which is compatible with the treatment of continuous attributes and discrete attributes;
3. Random forest method can effectively solve the problem of unbalanced classification;
4. Random forest method has excellent noise tolerance and high classification accuracy.

In the context of big data environment, random forest method is also characterized with following advantages:

1. From the perspective of the huge amount of big data, the random forest method can be competent;
2. The relatively simple decision trees generated by random forest method facilitates business analysts to interpret its meaning;
3. Random forest method is suitable for distributed and parallel environment, showing a good scalability;
4. The simple classifier created by decision trees can process data efficiently, which is applicable to the characteristics of rapid data refresh rate in the big data environment.

2.3 DISADVANTAGES OF RANDOM FOREST METHOD

Despite of the abovementioned advantages, random forest method is also facing new challenges in the mode of big data:

First, in the big data environment, the data update rate is very fast, so are the update rates of data characteristics and modes hidden in data. Decision trees based on training sets data will become out of date and less accurate in classifying data after a certain period of time. This requires algorithms in the big data environment to have data adaptability. Meanwhile, this ability should also be quickly reflected in the classifier, while the normal conduct of business or the stream-form passage of data through the classifier should not be affected.

Secondly, the decision tree, in fact, is a greedy algorithm that easily leads to instability and over fitting. Solving this problem has a great significance for improving the accuracy of decision tree.

Thirdly, the forest scale established by random forest has not been clearly defined; oversized scale may result in redundancy and thus reduce the efficiency and accuracy of classification.

3 Algorithm improvements in big data environment

3.1 ACCURACY AND PRUNING OF DECISION TREES

In order to meet the needs in the big data environment, improved algorithm should have the following characteristics:

1. It can quickly generate a classifier on a given data training set;
2. The resulting classifier can quickly classify new streaming data;
3. An algorithm should be of adaptability so as to respond to the changes in data modes and guarantee its accuracy;
4. It should limit the scale, namely, the number of trees, which will, on the one hand, ensure the efficiency of the algorithm, while on the other hand, will also guarantee its accuracy.

Before improving the algorithm, the accuracy A_t of a tree *t* from the random forest is first defined as follows:

Wherein, n_r is the time that the decision tree gives correct results, and *n* is the data amount processed by this tree. The accuracy indicates the ratio how often a certain tree gives correct results in a period of time.

For the classification problem, it is considered that the decision tree gives a correct result if the classification result given by decision tree *t* is consistent with the final result. For the regression problem, it is required to calculate the difference between the result x_i given by decision tree *t* and the final result, and their standard deviation will be taken as the accuracy rate of *t*:

According to the accuracy rate, we can measure the accuracy of a tree in period of time. The idea of algorithm improvement is to track the accuracy rate of each tree during the implementation process, regularly update the forest and eliminate those trees with lowest accuracy rate.

The improved random forest method is as follows:

1. Construct a decision tree group in accordance with standard random forest method.

2. Build a record sheet T_t for each decision tree t to record the generated results during execution.

3. After running for some time, the record sheets of all decision trees are scanned to pick out and delete those trees with lowest accuracy.

After accuracy screening, the number of trees in the forest will be reduced, thus realizing the pruning of all decision trees. However, excessive reduction in the number will also lead to reduced accuracy in the whole decision tree sets [11].

In order to maintain the decision trees to a certain number, these data sets should be tracked when pruned, thus generating new decision trees to maintain the quality of the entire forest.

3.2 SAMPLE SCREENING BASED ON MARGINS

In order to screen out more useful samples for decision trees from the data sets, we introduce the definition of margin.

Margin refers to the overall decision-making correctness rate of a random forest on a piece of given sample data *(x, y)*. It is calculated as follows:

$$margin(x,y) = av_k I(h_k(x) = y) - \max_{j \neq y} av_k I(h_k(x) = j) \quad ,$$

where $av_k(\ )$ is an averaging function and *I ( )* is a metric function. If the corrected results with respect to sample *(x, y)* can be obtained from most decision trees in the random forest, the *margin(x, y)*>0. The case of *margin(x, y)*<0 indicates that the sample is wrongly identified by most decision trees, suggesting that the algorithm draws a wrong conclusion on the sample.

Samples with *margin(x, y)* >0 illustrates that decision trees are able to gain right results. Since those decision trees in high similarity with the existing trees will not improve the accuracy of entire forest, such samples do not need to be processed again. Samples with *margin(x, y)* <0 will be recorded and used to form a new training data set *S'* so as to allow the newly generated decision trees to improve the accuracy of entire forest. Although the data set *S'* only account for a small part in the entire data set *S*, its data features are very different from other data.

3.3 GENERATION, SCREENING AND ADDITION OF NEW DECISION TREES

By applying the random forest method on the data set *S'*, a new decision tree set $\{h'(x,\theta_k),\ k=1,...\}$ is thus obtained. Since the data set *S'* represents only a small part of data from the entire data set, a certain proportion of decision trees should be screened from this set and added to the original decision tree set.

The number of decision trees to be screened can be determined based on the ratio between the data set *S'* and the entire data set *S*:

The screening methods may involve those as follows:

Methods 1, based on the accuracy sorting obtained by testing the data sets *S'*, N_{new} decision trees with maximum accuracy will be selected.

Methods 2, based on the accuracy sorting obtained by testing the entire data sets *S*, N_{new} trees with maximum accuracy will be selected.

Methods 3, calculating the ratio between margin mean and margin variance of each tree on the data set *S'*[12], which is taken as the importance measurement index for each tree, N_{new} trees with highest importance will be selected.

The improved random forest method has been shown below:

```
 Algorithm: newRandomforest(S)
        For i=1 to T do:
$T_{set}$ = bagging(S)
ChooseAttribute(M from N)
$h_i$ = buildTreeCART($T_{set}$, M)
addTree($h_i$, H)
endFor
Output (H)
endAlgorithm
runAlgorithmWithData((x, y), H)
If margin(x, y) <= 0
$S' = S' + (x, y)$
endIf
calculateAccurate(H)
deleteBadTrees(H)
$H'$ = newRandomforest($S'$)
H = H + chooseTrees($H'$)
        endRun
```

4 Data validation

4.1 TESTING DATA SET

Data sets used in the test are originated from real customer data of financial industry. The data amount accounts for 200,000 pieces with around 10,000 pieces of data from each quarter, which was sampled from a larger original 5-year data set.

The data set contains a target category and 16 feature attributes, which includes both the continuous numerical attributes and discrete attributes. Random forest kit in R language version was used in the test.

In order to verify the effectiveness of abovementioned improvements, we used 10000 pieces of data of the first quarter in the first-year as the initialized training set and use them to establish the initial random forest, where the number of trees equals 100.

4.2 THE INFLUENCE OF THE CHANGES IN DATA PATTERN ON ALGORITHM

Firstly, we validate when data changes with the time, whether its pattern will change and if such changes will affect the accuracy of the algorithm.

Through taking 10,000 pieces of data in the first quarter of the first year as initialized training set to establish random forest and using the established random forest to classify the data of each following quarter, it was clearly observed that the original random forest gradually became unadapted to new data and its accuracy also reduced slowly with the change of time:

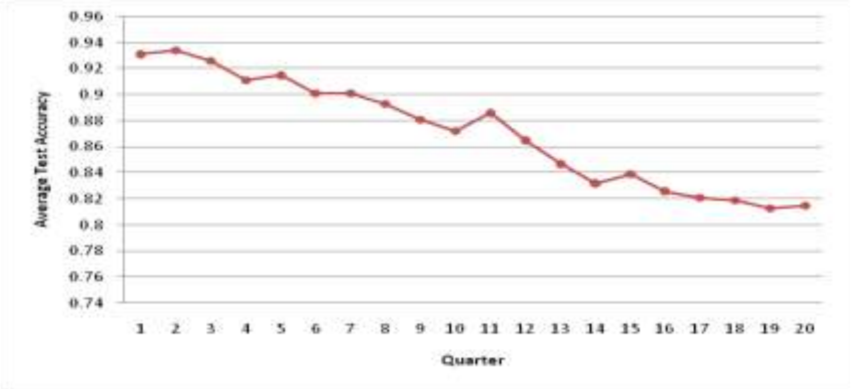

FIGURE 1 Accuracy of Random Forest Decreases with Time

After pruning, the accuracy of the original data set fluctuates with the changes in the pruning number, but the accuracy generally did not change significantly. In the process of decreasing the number of trees from 100 to 20, the changes in the algorithm accuracy has shown as below in Figure 2

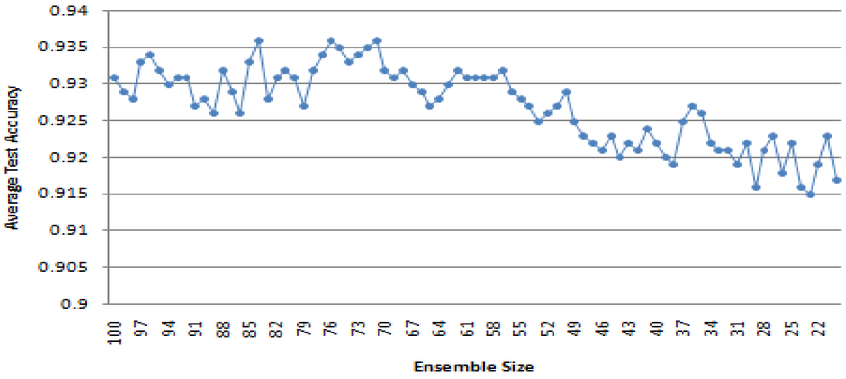

FIGURE 2 Effect of Random Forest Pruning on Accuracy

After pruning the random forests using this method, the accuracy fluctuation of different-sized decision tree sets also varied as shown in Figure 3.

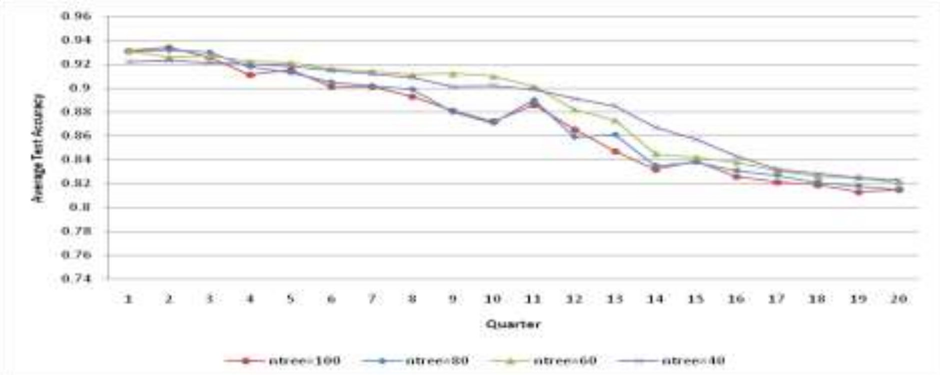

FIGURE 3 Effect of Different Pruning Levels on Accuracy

It can be seen that for different pruning levels, effect on algorithm is also slightly different with time. In general, pruning would increase the accuracy of the entire forest and decrease the influence of data changes; however, the effect was not prominent. The number of pruning was defined as 50, namely retaining the tree number to be 50; new decision trees generated from improved random forest method were then added to the original random forest. According to the three decision tree screening methods as described in section 2.3, the performance of improved random forest method on these data sets have been shown in Figure 4.

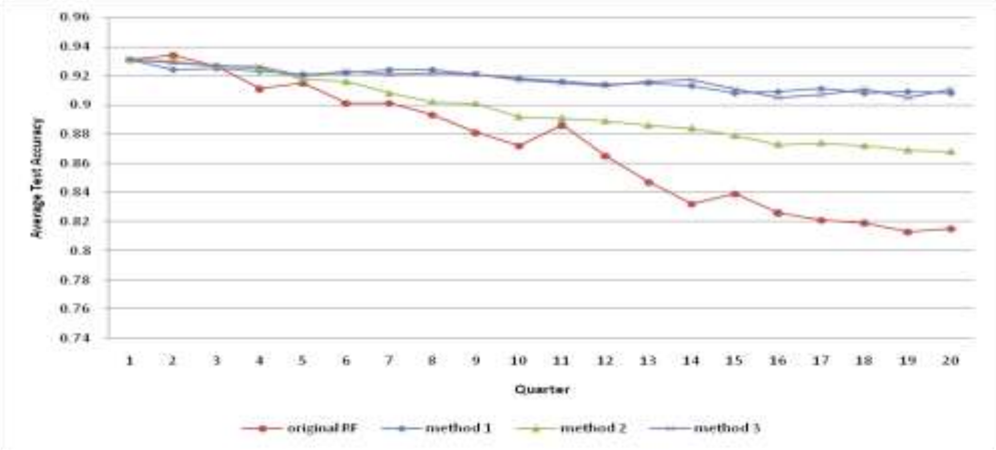

FIGURE 4 Accuracy Comparison on the Improved Random Forest Method

It can be found from above chart that the adaptability of improved random forest algorithm to the changes of data pattern is far better; the whole forest also updates slowly with time so as to maintain a high accuracy rate on new data. Among those methods for screening new decision trees, both the screening method by using the accuracy rate of new sample data set S' and the screening method by using the ratio between each tree's margin mean and margin variance can gain satisfactory results.

5 Conclusions

Based on the original random forest method, this article proposed a new improved model for the algorithm. After improvement, it operates well in the today's big data mode. Especially, its data patterns will also change gradually with time, which allows it to give a better play in the big data scenarios.

In the regard of process, the improved method needs not to scan the processed data again. Instead, it only requires to record the processing results of each decision tree when using classifier for processing data and to record the wrongly processed sample data when generating the final results. It has better practicability and feasibility due to this variation as well as its low required storage and computation costs.

The improved method also has a good performance in verification by using real financial industry data. However, it should be noted that it requires in-depth exploration on other aspects. For example, whether can it be adopted to other types of data sets? Whether can the pruning decision functions be improved? What is the appropriate proportion of the new decision trees? All of these questions need to be further explored in subsequent studies.

References

[1] Han J, Kamber M. Data Mining: Concepts and Techniques[M]. San Francisco, CA: Morgan Kaufmann Publishers, 2001

[2] Liu B, Ma Y, Wong C K 2001 Classification using association rules: weakness and enhancements. In Vipin Kumar, et al. Data Mining for Scientific Applications

[3] Bernardo J M, Smith A F M 2001 Bayesian Theory. Measurement Science and Technology **12** 211

[4] Liu Hongyan, Chen Jian, Chen Guoqing et al 2002. Review of Classification Algorithms in Data Mining *Journal of Tsinghua University (Science and Technology)* **42**(6) 727-30

[5] Li Xiujuan, Tian Chuan, Feng Xin, et al 2010 Research on Classification Technology in Data Mining *Modern Electronics Technique* **33**(20) 86-8

[6] Li Xuechan 2008 Research on Classification Calculation Way of a Great Amount of Data According to the Database Sampling *Computer Science* **35**(6) 299-cover 3

[7] Shafer J, Arawal R, Mehta M 1996 SPRINT: a scable parallel classifier for data mining *Proceedings of the 22th International Conference on Very Large Data Bases* 544-55

[8] Breiman L 2001 Random Forests *Machine Learning* **45**(1) 5-32

[9] Zhang H P, Wang M H 2009 Search for the smallest random forest, *Stat. Interface* **l2** 381-8

[10] Robnik-Sikonja M 2004 Improving Random Forests *Proceedings of the 15th European Conference on Machine Learning* 359-70

[11] Leistner C, Saffari A, Santner J, et al 2009 Semi-supervised random forests *IEEE 12th International Conference on Computer Vision* 506-13

[12] Shen Chunhua and Li Hanxi 2010 Boosting through Optimization of margin Distributions *IEEE Transactions on Neural Networks* **21**(4) 659-75

Author

Liu Yingchun , 1980.12, Wucheng County, Shandong Province,P.R.China

Current Position, Grades: The PHD of School of Economics and Management, BeIHang University, China.
University studies: received her B.sc.in Computer Science and Technology form the Northwestern Polytechnic University of Xian in China. She received her M.Sc. From the BeiHang University in China.
Scientific interest: Data Mining algorithm, research and application of big data and so on.
Publications: more than 4 papers published in various journals
Experience: She has completed four scientific research projects.