International Core Journal of Engineering Volume 7 Issue 5, 2021
ISSN: 2414-1895 DOI: 10.6919/ICJE.202105_7(5).0029

A Survey on the Classification Algorithms based on Big Data

Xudong Wei, Zhaofei Li, Zenan Wang, Yumei Chen, Xiaodong Tong, Sijing Deng, Wei Pu, Yuanbo Zhang, Xueshen Liu, Li He, Bin Liu

[1]Sichuan University OF Science & Engineering, Zigong 643000, China;

[2]Artificial Intelligence Key Laboratory of Sichuan Province, Zigong 643000. China.

Abstract

With the explosive growth of data, the use of big data technology machine learning classification algorithms to predict the results can improve the intelligent classification of data. It can provide data support for predicting classification in advance. Filter out the classification results to improve the efficiency of data processing and data realization. This article first introduces the development process of machine learning under big data, introduces the mainstream distributed processing framework spark, and then compares the advantages and disadvantages of classification algorithms under big data.

Keywords

Big Data; Spark; Machine Learning; Classification Algorithm.

1. Introduction

Machine learning is a branch of computer theory science formed by the combination of artificial intelligence computing, pattern recognition and other related basic disciplines. It is the foundation and core of artificial intelligence and is widely used in various fields such as voice, video, analysis and prediction. According to research, in most states, the larger the data volume, the higher the accuracy of the machine learning model. Therefore, machine learning is the main method of data processing, analysis, processing, and application under big data. The most important factor that affects its performance in this prediction system is the correct rate and efficiency of the selection of the classification algorithm. Improving key indicators requires a comprehensive discipline that combines multiple fields, involving many basic theories such as statistical methods, numerical analysis, mathematical statistics, and probability theory. The fundamental research core of this discipline is to explore how computers simulate and realize human learning behaviors. Through learning, computers can obtain new knowledge through information, and even possess innovative capabilities, reorganize and generate new knowledge structures, and optimize the performance of models. Research at home and abroad has shown that classification algorithms such as random forest, linear support vector machine Linear SVM, logistic regression algorithm logistics, decision tree algorithm decision tree have good performance.

2. General big data classification model

The process of the general big data classification model proposed in this paper is shown in Figure 1. The following will introduce mainstream big data analysis platforms and common classification algorithms.

2.1 Development of distributed processing platform spark

Spark is a large-scale distributed processing engine based on Hadoop mapreduce and a big data computing platform developed by the University of California, Berkeley. Compared with Hadoop, it has better performance in terms of speed, versatility, ease of use, and compatibility. Unlike

ISSN: 2414-1895 DOI: 10.6919/ICJE.202105_7(5).0029

MapReduce, Spark caches the output intermediate results in memory and directly participates in the next step of calculation. In MapReduce, local disks are frequently read and written. This makes Spark have an absolute advantage over MapReduce in terms of performance in dealing with iterative problems, and can better adapt to machine learning. The core of Spark, Spark Core, has a complete technology ecosystem BDAS. In addition to the basic Spark computing framework, it also has more advanced application sub-frames, mainly including Spark Streaming, structed Streaming two different response level stream processing frameworks, SparkSQL query analysis engine based on Dataframe, MLlib machine learning library and so on. It is precisely because of this feature that Spark MLlib is quite popular in the field of machine learning.

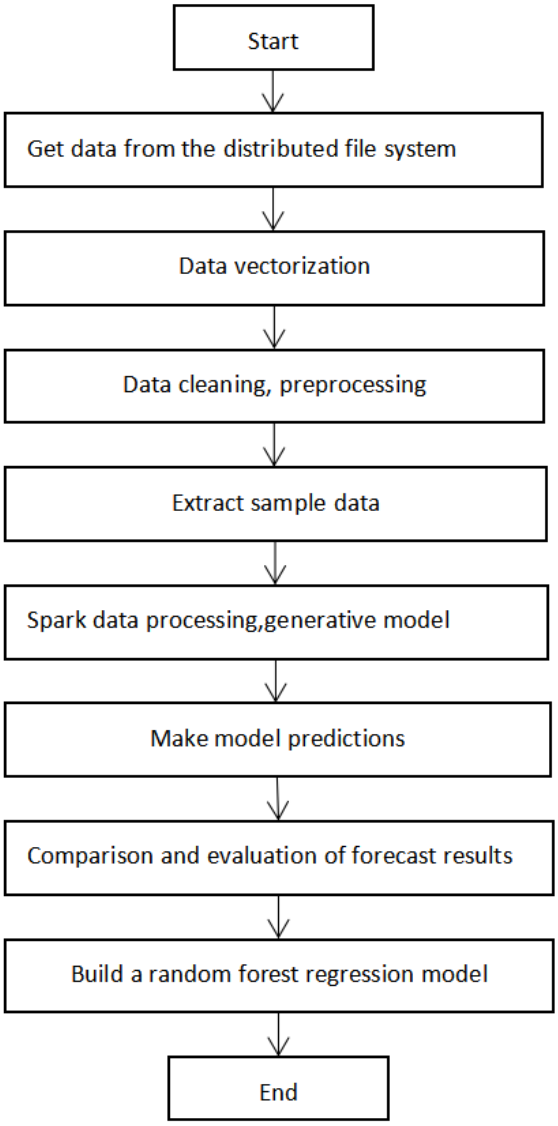

Figure 1. General big data classification model prediction flow chart

Spark RDD elastic distributed data set is a data abstraction of Spark core, proposed by spark0.0 version. Compared with ordinary data sets, RDD uses partitioned storage to facilitate parallel operations. Therefore, creating an RDD and operating on the RDD is the processing of data. Spark's high iteration efficiency is reflected in the RDD can be directly cached in memory, and the output result of the previous RDD can be directly used as the input of the next RDD in memory. Spark encapsulates the underlying processing and storage process. Users only need to call the upper-level interface to centrally process business logic and improve the efficiency of organizing code. RDD compile-time type safety. But whether it is IO operations or communication between clusters, it is

ISSN: 2414-1895 DOI: 10.6919/ICJE.202105_7(5).0029

necessary to serialize and deserialize structured data. This will frequently create and destroy objects, which will increase GC overhead.

Spark SQL is used to query the underlying massive data through SQL. The traditional shark components completely copy the logic code of Hive, making the maintenance and optimization of shark completely dependent on Hive. Hive is proposed on top of the optimization strategy of Mapreduce. Compared to the process-level parallel mapreduce, spark is thread-level parallel. Therefore, there is a thread safety problem in the compatibility of shark on spark. For compatibility, it is necessary to design and maintain a set of independent patches. Spark SQL only relies on HiveQL parsing and Hive metadata. Then, after HQL is parsed into an abstract syntax tree (AST), all subsequent processes are processed by Spark SQL. The SparkSQL calculation process and optimization are coordinated by Catalyst (functional relational query optimization framework). The implementation of Spark SQL solved the two problems of shark.

A distributed data set organized in DataFrame columns was proposed by spark1.3. It is a data abstraction created on the basis of Spark SQL. Spark reads data through metadata. Therefore, you only need to serialize the data during IO and read, ignoring its own structure. Spark can serialize data to off-heap, and directly manipulate data by manipulating off-heap memory. Schema and off-heap, DataFrame solves the shortcomings of RDD.

Later, after spark1.6, a distributed data set of Dataset, Encoder was proposed, which is structured data that has been serialized. His compile-time type is security check, performance is greatly improved, memory usage is also greatly reduced, GC overhead is greatly reduced, network data transmission is greatly reduced, and the use of scala and java programming is greatly reduced. The difference of the code.

2.2 Overview of classification algorithms

2.2.1 Overview of Decision Tree Algorithm

The tree structure of the decision tree is composed of root nodes, internal nodes and leaf nodes. Among them, the root node contains the complete set of samples to be classified, the internal node refers to the judgment on its attribute, and the leaf node represents a classification result. The essence of a decision tree is a tree composed of multiple judgment nodes. Finding the most important features that affect the target value and the way of dividing features is the core of building a decision tree. The algorithm flow of the decision tree is shown in the figure below.

Information gain is the basis for ID3 decision tree division, information gain = information entropy (front)-information entropy (back)

$$Ent(A) = -\sum_{k=1}^{n} p_k \log_2 p_k$$

When doing feature selection or data analysis, focus on features with high information gain

$$Gain(\mathrm{D}, \mathrm{a}) = Ent(D) - \sum_{i}^{k} p_i \times Ent(D \mid i)$$

The information gain rate is the basis for dividing the C4.5 decision tree, and the gain ratio metric is defined by the ratio of the gain metric Gain and the split metric Splitinformation

$$GainRatio(S_A, A) = \frac{Gain(S_A, A)}{SplitInfomation(S_A, A)}$$

$$SplitInfomation(S_A, A) = -\sum_{m \in M} \frac{|S_{Am}|}{|S_A|} \log \frac{|S_{Am}|}{|S_A|}$$

The Gini value Gini(D) is used to represent the probability that the category labels of two samples randomly replaced from the data set D are not consistent. The purity of the Gini value data set is negatively correlated.

ISSN: 2414-1895 DOI: 10.6919/ICJE.202105_7(5).0029

$$Gini(D) = \sum_{k=1}^{|y|} \sum_{k'=k} p_k \; p_{k'} = 1 - \sum_{k=1}^{|y|} p_k^2$$

The CART decision tree is divided based on the Gini index. The attribute with the smallest Gini index after division is used as the optimal sub-attribute.

$$Gini\_index(D,a) = \sum_{v=1}^{V} \frac{|D^v|}{|D|} Gini(D^v)$$

The decision tree is optimized through pruning, and there are two ways of pruning beforehand and pruning afterwards. Pre-pruning is to set a judgment basis to stop the growth of the tree during the construction of the decision tree. Post-pruning refers to the introduction of a test set after the decision tree is constructed to verify the classification and prediction results of the decision tree for the new input data. The generation of the decision tree is divided into three steps: data processing preparation, decision tree construction, and decision tree pruning, as shown in Figure 2:

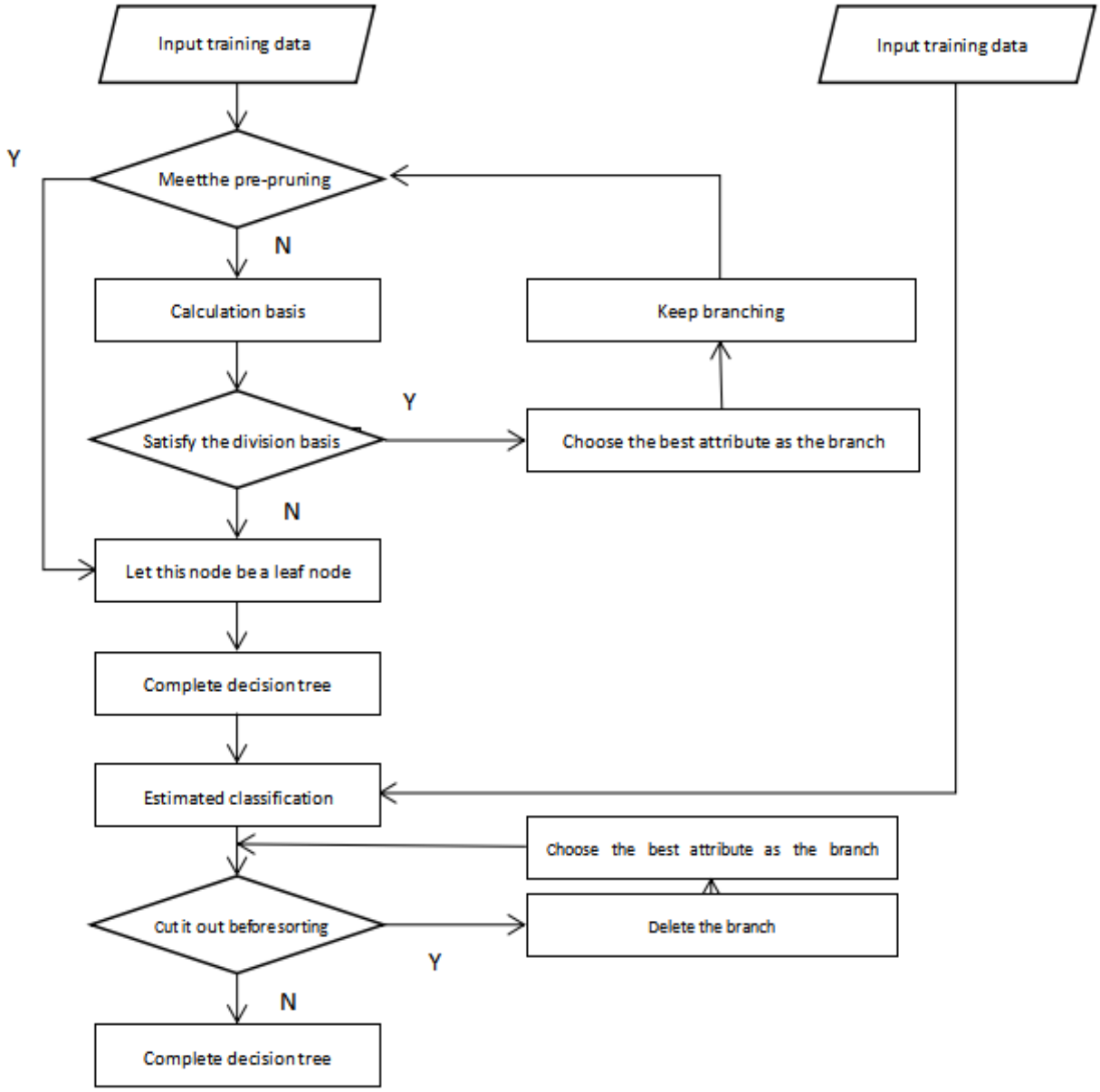

Figure 2. Flow chart of decision tree

ISSN: 2414-1895 DOI: 10.6919/ICJE.202105_7(5).0029

(1) First, the obtained data is divided into training data and test data, and then a node is generated to determine whether the node meets the pre-pruning conditions. If it is satisfied, it is set as a leaf node. If it is not satisfied, the best division method is selected according to the above division basis, and the branch is continued, and then the pre-pruning condition is judged.

(2) Repeat the steps to complete the construction of the decision tree, and estimate the correct rate of the decision tree classification by introducing the test set. It is judged whether the branch cut can improve the correct rate, and if possible, the node can be removed to continue to judge whether the tree node should be pruned and finally completed the construction of the decision tree.

2.2.2 Random forest overview

Random Forest is one of the most commonly used Bagging algorithms. Because of the application of a random process, multiple decision trees are generated differently. The tree model is obtained by combining elections to reduce variance and improve accuracy. The random characteristics of random forests and the principle of separate training of decision trees make the generation process parallel. The randomness of generating a random forest is reflected in the sampling of the original data from different training sets in each iteration. Each tree node is split by different feature subsets, and the training process of the random forest tree is the same as the training process of the individual decision tree. Random forest compares the prediction results of its various decision trees to select the best model. The choice of regression and classification is slightly different. The classification problem uses a voting system, each decision tree votes for a category, and the tree with the most votes is the final model. The prediction result of each tree in the regression problem is a real number, and the final prediction model is the average of all prediction results. Figure 3 shows the flow chart of the random forest:

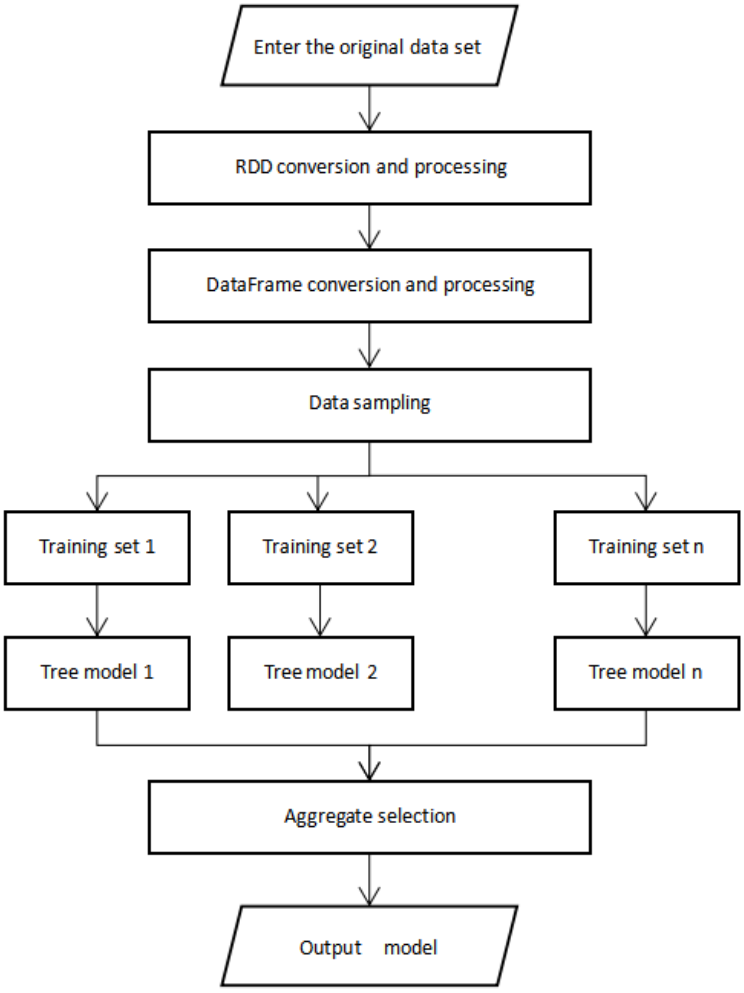

Figure 3. Flow chart of random forest

ISSN: 2414-1895 DOI: 10.6919/ICJE.202105_7(5).0029

(1) Generate an RDD data set from the data set in csv format through sparkContext's textFile. Perform format conversion and data preprocessing operations on liquor-related RDD data, use the createDataFrame() function in Context to convert liquor-related RDD data sets into DataFrame data sets, and perform data cleaning and data processing operations on the data sets.

(2) Train the DataFrame to generate multiple decision trees. In this paper, the Bagging algorithm is used to sample the data again. In Random Forest, in addition to randomly selecting samples (besides the basic operation of Bagging) before training each weak classifier, in order to make each tree not so similar, a subset of features must be randomly selected from the feature set for training . If several features and labels have a strong correlation, then these features will be selected in all tree clocks. The random forest algorithm selects only i from n features for training, avoiding the occurrence of overfitting.

(3) Finally, multiple decision trees are integrated, and the results of each decision tree are counted by voting, and the one with the most votes will be output as the final model.

2.2.3 SVM support vector machine

For a large amount of data processing and analysis, in addition to the corresponding analysis of the data In addition to this category, statistical work should also be carried out on this basis. This algorithm is a type of supervision Learning method, based on the VC dimension theory and the least structural risk in the statistical theory Principle-based, combining limited sample information with model complexity and learning ability Find the optimal processing path in time to obtain the best generalization ability. Support Vector Machines The method is the data algorithm proposed in recent years. The main ideas include the following two layers

Surface: One is to analyze the linearly separable state, and the linearly indivisible state With the help of non-linear mapping algorithm, the conversion between samples is realized, that is, input from low-dimensional The space linearly inseparable becomes a high-dimensional feature space to achieve linearly separable; the second is Based on the structural risk minimization theory, create the optimal score in the feature space Cut the plane to achieve global optimization.

3. Concluding remarks

Big data is available. Big data processing has become a research hotspot today. Combining scientific statistical methods and adapted machine learning algorithms can make data processing efficient and correct, enhance the efficiency of data processing, reduce the difficulty of data processing, and enhance the model's performance. Correct rate. In addition, in order to cope with the growth of big data and data processing requirements, research should be conducted on the basis of traditional machine learning algorithms, and algorithms suitable for various models should be proposed to improve data processing and mining capabilities.

References

[1] Han Chengcheng, Zeng Sitao, Lin Qiang, Cao Yongchun, Man Zhengxing. Summary of stream data classification algorithms based on decision trees[J]. Journal of Northwest University for Nationalities (Natural Science Edition), 2020, 41(02): 20-30 .

[2] Ou Huajie. Overview of machine learning algorithms in the context of big data[J]. China Information Technology, 2019(04):50-51.

[3] Liu Xia, Ou Zhipeng, Chen Yinan, Li Yuanhui, Chen Lei. Route passenger flow prediction simulation based on three models[J]. Journal of Henan Education Institute (Natural Science Edition), 2019, 28(01): 37-40+46.

[4] Fan Xinxin, Chen Xiuguo, Yang Ya, Yan Hongmei, Wang Jianbin. Communication power state evaluation system based on SVM in cloud environment[J]. Journal of Anhui Electrical Engineering Vocational and Technical College, 2019, 24(01): 114-117.

[5] Cao Hui, Yang Lijian. A three-dimensional contour reconstruction algorithm for magnetic flux leakage signal defects based on bias estimation [J]. Nondestructive Testing, 2019, 41(02): 33-39.

ISSN: 2414-1895 DOI: 10.6919/ICJE.202105_7(5).0029

[6] Chen Guangkai, Chen Shuhong, Pan Wei, Chen Chen, Jiao Runhai. Electricity tracking algorithm based on random forest rolling prediction[J]. Wisdom Electric Power, 2018, 46(12): 45-49+104.

[7] Zhou Jianfeng. A method for feature selection and prediction of popular Weibo based on FA-SVM[J]. Computer Applications and Software, 2018, 35(12):107-111.

[8] Li Xing, Li Tao. Design and implementation of a recommendation system based on Spark[J]. Computer Technology and Development, 2018, 28(10): 194-198.

[9] Xiao Yao, Bi Junfang, Han Yi, Dong Qiwen. Research on click-through rate prediction in online advertising[J]. Journal of East China Normal University (Natural Science Edition), 2017(05): 80-86+100.

[10] Jin Xin, Yan Longchuan, Liu Jun, Zhang Shulin. Decision tree-based automatic fault diagnosis and analysis method of enterprise information system[J]. Telecommunications Science, 2017,33(03):163-167.

[11] Jia Bin, Ma Yan, Zhao Xiang. DDoS attack traffic distributed detection model based on combined classifier [J]. Journal of Huazhong University of Science and Technology (Natural Science Edition), 2016, 44(S1): 1-5+10.