Application research of data mining algorithm in big data environment

Zenghui Zhao
School of Intelligent Science and Information, Xi'an Peihua University, Shaanxi 710000, China

Corresponding author and e-mail: Zenghui Zhao, 296883619@qq.com

Abstract. The Hadoop platform forms a complete large-scale ecological distribution system, including HDFS, MapReduce, HBase and other subsystems. This paper analyzes the parallel processing of Hadoop platform and applies it in the field of data mining algorithms. In order to obtain better algorithm efficiency, a K-Modes clustering algorithm based on big data platform is proposed. It uses cluster mode to replace the central node. The mining process uses naive Bayes to improve mining efficiency. The experimental results show that it has better adaptability, saves time and improves the efficiency of the algorithm.

1. Hadoop platform
Hadoop platform is a distributed processing cluster platform composed of common commercial servers, which can process massive data resources stored in parallel. Since the open source of Apache Hadoop system, at the beginning, there were only HDFS, MapReduce, HBase and other subsystems. At present, Hadoop platform has formed a complete large-scale ecological distribution system.

MapReduce consists of two phases: Map and Reduce, through which parallel computation and stored procedures are realized. In addition, MapReduce framework provides a universal interface for parallel programming, which allows users to invoke the interface according to their needs and quickly complete the data program processing of big data parallelization. MapReduce realizes the data processing through the input method of key value data, and realizes the data processing, and automatically completes the partition data and scheduling data management. In the process of parallel program computing storage, MapReduce completes processing scheduling and allocation of computing resources by analyzing and processing the parallel framework. In addition, it classifies the input/output data and performs relevant degree scheduling to monitor the degree state. In this way, it can realize the collection, sorting and synchronization of computing nodes.

2. Research on data mining algorithm

2.1. Study on clustering data mining algorithm

2.1.1. Clustering algorithm
K-modes clustering algorithm is a kind of unsupervised learning method, very adaptive. The principle of the algorithm is as follows: assume that the sample is, and the number of attributes in the data set is, and then the difference value of the sample can be obtained:
$d(X,Y) = \sum_{j=1}^{m} \delta(x_j, y_j)$

$$\delta(x_j, y_j) = \begin{cases} 
0, & (x_j = y_j) \\
1, & (x_j \neq y_j) 
\end{cases}$$

$Q$ is represented as a cluster center matrix, $Q = [q_{1,1}, q_{1,2}, \ldots, q_{1,m}]$, Then $W$ sample matrix can be divided:

$$w_{i,l} = \begin{cases} 
0, & i \in l \\
1, & i \notin l 
\end{cases}$$

The objective function of k-modes clustering algorithm is:

$$P(W, Q) = \sum_{l=1}^{k} \sum_{i=1}^{n} \sum_{j=1}^{m} w_{i,l} \delta(x_{i,j}, q_{l,j})$$

Where, $k$ represents the number of clusters, $n$ represents the number of samples, and $m$ represents the number of attributes.

The basic principle of k-modes clustering algorithm is to first divide the samples into highly similar class clusters, but the similarity between different classes of clusters is very low. Assumes that the number of samples is $N$, the algorithm from the first $N$ samples freely choose $K$ samples, as a cluster center, then, in turn, to calculate the distance between each sample and cluster center, maximum sampling the similarity of samples, to be classified as group clustering, clustering cluster formation, after iteration, until after all kinds of the center of the cluster stability can be concluded that $K$ clusters. The basic flow is shown in table 1.

**Table 1. Basic flow chart of k-modes algorithm**

| Input:       | data source to be clustered, cluster central set defined for the first time |
|-------------|--------------------------------------------------------------------------------|
| Output:     | clustering results                                                              |
| Map() function | First, define the Map node. Divide the first round of clustering center into the initial Map nodes with DistributedCache algorithm; |
|             | for each i(sample) in m(The data set) Solve the difference value between i and the cluster center, and mark the smallest difference value as i |
|             | In formula (2), the target value is accumulated and calculated |
|             | for each h(attribute) in i(sample) Count the mode of each attribute and write it to the mode matrix |
|             | Output mode matrix and final objective function. |
| Reduce() function | Store all the results from the map () function |
|             | The value of each element in the same position in the cumulative mode matrix is formed into a new matrix, and the value of the objective function is calculated twice. |
|             | Select the value according to the principle of minimum difference value and update the cluster center; |
|             | Returns the target function value and enters it into the Main() function |
| Main() function | If the result of two loops is the same, the algorithm terminates. |
|             | If the target values of the two loops are different, continue with the Map() function |
2.1.2. Algorithm instance

Clustering algorithm k-modes application practice, to analyze the sample analysis of ants looking for food, the ant is through its tentacles and touch feet to detect the distance between food and its own, so through the ant touch feet and tentacles for the attribute analysis of ant colony food search and food distance relationship, the data is shown in table 2.

| The serial number | Antenna length (mm) | distance (mm) | Touch the foot length (mm) | distance (mm) |
|-------------------|---------------------|---------------|---------------------------|---------------|
| 1                 | 10                  | 30            | 3                         | 50            |
| 2                 | 13                  | 50            | 6                         | 70            |
| 3                 | 12                  | 60            | 8                         | 90            |
| 4                 | 15                  | 80            | 3                         | 100           |
| 5                 | 21                  | 100           | 5                         | 120           |
| 6                 | 22                  | 120           | 10                        | 150           |

Then divide it into 2 groups, that is, k=2.

Firstly, the objects in the table are grouped in the form of 3 groups and allocated to Mapper nodes. The group number corresponds to the node number of Mapper, and then the Cost variable Cost=-1 is calculated through Reduce function.

Then select any data from the group to obtain the clustering center, respectively:

- CostA= (length of tentacles =10, distance =30, length of contact feet =3, distance =50)
- CostB= (antenna length =21, distance =100, contact length =3, distance =100)

According to the algorithm flow in table 1, the above data is calculated and processed, and the final result is output, so that the clustering center can be obtained. After the above iteration, it can be concluded that the final node center is:

- CostANew= (length of antenna =12, distance =60, length of touch foot =8, distance =90)
- CostBNew= (length of tentacles =15, distance =80, length of contact feet =4, distance =100)

2.2. Classification algorithm based on Hadoop platform

The main process of naive bayesian classification algorithm of Hadoop platform is as follows: the user will send a request job to the Hadoop platform, and the Hadoop platform will randomly assign a job number after receiving the job. The background job server will then process the job. After analyzing the job structure information, it will be assigned to the task server of relevant job nodes for processing. All training sets and intermediate processing results in the process of job processing are stored in the HDFS file set. After processing the training set data, the classifier will return relevant result values to the corresponding subsequent task nodes. When all JobTracker has finished, the algorithm is finished.

2.2.1. Naive bayesian classification algorithm

The basic theory of naive bayes classification algorithm is bayes' theorem, and the mathematical model of bayes' theorem is as follows:

\[
P(A | B) = \frac{P(B | A)P(A)}{P(B)}
\]

In the above equation, the probability of P (A) only when A occurs is also called prior probability. The posterior probability is P (A|B), representing the probability of A occurring under the condition of B; Similarly, P (B|A) is the probability of B occurring under A condition.

When two events A and B are independent of each other, P(AB)=P(B|A)P(A).

Training and testing are important steps in the process of classification algorithm. First of all, after the prior probability is calculated, the posterior probability is obtained by using the bayesian formula. In the classification of object attributes, the sample classifier should be used for probability calculation, so as to obtain the category of the maximum probability. The specific process is as follows:
If we define \( C_1, C_2, \ldots, C_n \), with \( n \) classes, thus the data set sample \( X \) to be classified can be obtained:

\[
P(C_j \mid X) \geq P(C_j \mid X), \quad 1 \leq j \leq n, \quad j \neq i
\]

The naive bayesian classification algorithm directly assigns the unclassified sample \( X \) mentioned above to the class with the highest posterior probability.

When \( P(X \mid Y_i)P(Y_i) \) reaches the maximum value, the conditional probability reaches the maximum value, and the calculation formula is as follows:

\[
P(X \mid Y_i) = \prod_{k=1}^{n} p(x_k \mid Y_i)
\]

In the formula, \( P(X) \) can be omitted as a fixed value. \( P(Y) \) is obtained by calculating the ratio of the training record. The attribute set \( X = \{x_1, x_2, \ldots, x_n\} \) contains \( n \) attributes.

The posterior probability calculation formula is as follows.

\[
P(Y_i \mid X) = \frac{P(Y_i) \prod_{k=1}^{n} p(x_k \mid Y_i)}{P(X)}
\]

2.2.2. Test results

Experimental data were derived from the data set in UCI: Pokerhand and Skin Segmentation were shown in Table 3.

**Table 3.** Original data table

| The serial number | The data set       | Sample size | dimension | Number of categories |
|-------------------|--------------------|-------------|-----------|---------------------|
| 1                 | Pokerhand          | 1010884     | 10        | 11                  |
| 2                 | Skin Segmentation  | 1246183     | 5         | 3                   |

Part of the data in the data set is randomly taken as the test set, and the average value is taken as the result output after running the result for several times. The number of test sets was gradually increased, and the time required for the execution of naive bayesian algorithm was analyzed, as shown in Table 5, as the speedup ratio (parallelization processing time on Hadoop platform/stand-alone processing time - 1). The analysis of Table 4 shows that large-scale data on Hadoop platform has classification efficiency.

**Table 4.** Experimental results schedule

| The serial number | Number of test sets (10,000) | Speed up than |
|-------------------|-------------------------------|---------------|
| 1                 | 1                             | 0.46          |
| 2                 | 5                             | 0.72          |
| 3                 | 10                            | 0.81          |
| 4                 | 25                            | 1.21          |
| 5                 | 50                            | 1.42          |
| 6                 | 75                            | 1.61          |
| 7                 | 90                            | 1.82          |
| 8                 | 100                           | 1.89          |

3. Conclusions

Hadoop platform forms a complete large-scale ecological distribution system, including HDFS, MapReduce, HBase and other subsystems. The algorithm proposed in this paper can effectively improve the efficiency of the algorithm by replacing the clustering center with cluster mode, and it has better self-adaptability. The effective implementation of classification mining based on naive bayes algorithm can effectively save the algorithm time.
Acknowledgement
Fund Program: University-level Scientific Research Program of Xi’an Peihua University in 2019, “Research on the construction of industry-university-research platform based on artificial intelligence” (No. PHKT19003).

References
[1] Li shuang, Chen ruirui, Lin nan. Hadoop framework k-means clustering algorithm for big data mining [J]. Computer engineering and design, 2018, 39(12):3734-3738.
[2] Chen e x. application research of data mining algorithm based on Hadoop [J]. Journal of bohai university (natural science edition), 2018, 39(03):274-280.
[3] Chen Juan. Algorithm analysis and application based on Hadoop Internet of things data mining [J]. Computer times, 2018(06):29-31+34.
[4] Wang 100. Study on clustering k-means algorithm based on Hadoop platform [J]. Computer and telecommunications, 2018(04):18-20.
[5] Hao yanni, tian weili. Application of data mining algorithm based on Hadoop in wine information data analysis system [J]. Computer application, 2017, 37(S1):72-74+79.
[6] Hong bo, lv yanxia, huang lei. Research and implementation of data mining algorithm based on Hadoop framework in big data environment [J]. Electronic design engineering, 2017, 25(07):41-44.
[7] Liu baolong, su jin. Double MapReduce improved Canopy-Kmeans algorithm [J]. Journal of xi’an university of technology, 2016, 36(09):730-737.
[8] Wang hao, shi wei, li huan. Research on meteorological data mining based on bayesian network under Hadoop [J]. Electronic devices, 2016, 39(04):841-846.
[9] Bu renzi. Study on parallelization of dbik-means clustering algorithm based on Hadoop platform [J]. Journal of xinyu university, 2016, 21(04):115-118.
[10] Li yanghui. Study on k-means clustering algorithm based on MapReduce [D]. Xi’an university of technology, 2016.
[11] Wang lei, zhang yongjian, jia jipeng, niu xiaoguang, nie changlong. Data mining method of public building energy consumption based on Hadoop [J]. Computer system application, 2016, 25(03):34-42.
[12] Ma bin, Yin lifeng. A parallel naive bayesian network public opinion classification algorithm based on Hadoop platform [J]. Modern book information technology, 2015(02):78-84.