# Apriori Algorithm Optimization Study Based on MapReduce

Li Chunqing [1, a]

[1] Department of mathematics and computer science, Guangxi National Normal University, Chongzuo, Guangxi 532200,China

[a]lcq_96672@sina.com

**Keywords:** MapReduce，Apriori algorithm optimization, distributed, pruning

**Abstract.** To solve the deficiency of algorithm distributed association rules based on MapReduce, this paper introduces global pruning strategy to increase algorithm efficiency, adopts frequent matrix storage to reduce the consumption of internal storage, and puts forward MFMDAP of frequent matrix storage of MapReduce calculation model. Experiments show that the algorithm in the paper elevates the algorithm efficiency and saves the usage amount of internal storage, which is in favor of the calculation and storage of big granularity data. The effectiveness of algorithm has been approved in experiments.

## Introduction

Association rules algorithm mainly aims to mine the interrelationship among things, and the main idea is to get frequent item set of data items. Presently, it is widely used in all kinds of classification design and related sales and other fields; association rules have become a very important research direction in data mining[1].

Nowadays, in the aspect of data mining research classification, there are studies on Bayes method and Boosting method[2]. The research aspects include decision-making tree, neutral network, genetic algorithm, rough set method, fuzzy set method, and the classical statistical regression method is applied in KDD[3]; in the aspect of association knowledge, there are discussions on all kinds of algorithm optimization and the methods to generate rules; the combination of KDD and database is also under research. Association rules mining is a kind of algorithm and I/O concentrated task; faced with large amount of data association rules mining, the calculated amount is giant, and traditional serial algorithm can not be dealt with efficiently. Therefore, it is necessary to introduce mining algorithm with high performance so as to finish association rules mining task effectively. For this, R.Agrawal et al put forward Count Distribution，Data Distribution，Candidate Distribution and other algorithm[4]. However, algorithms of this kind are limited to a certain degree; for example, Count Distribution algorithm will produce plenty of communication traffic and candidate items; Data Distribution algorithm will lead to heavy traffic load and unoccupied processor; Candidate Distribution algorithm can easily result to problems of unbalanced load.

To improve and realize a kind of efficient parallel multidimensional association rules mining algorithm, to make parallel multidimensional association rules mining on magnificent multidimensional data, and to elevate mining efficiency and reduce I/O load of system, the main research content of this paper is to put forward the method to build multidimensional data with parallel structure based on MapReduce distributed computation model and Hadoop distributed architecture, after analyzing multidimensional data features in detail; at the same time, this paper puts forward and realizes a kind of efficient parallel multi-dimensional association rules mining algorithm for the typical application multi-dimensional data.

## MapReduce model and distributed structure

MapReduce algorithm model originates from functional programming language at the first place, takes examples by the characters of vector programming language, and is comprised of two

functions of Map and Reduce. However, in the distributed framework, the application of MapReduce is different from the use of original function. Through MapReduce distributed calculation model, developers can put emphasis on the logical processing of two stages of Map and Reduce, instead of focusing too much on the specific realization of distributed algorithm. In the MapReduce distributed calculation model, the data procession is divided into two stages: Map stage and Reduce stage. As is shown Figure 1:

(1) Map stage: The main task is mapping. The input data is decomposed into numerous small data sets, the data in which are processed by every node in the cluster and intermediate results are produced.

(2) Reduce stage: The main task is simplification and reduction. The intermediate results in every node of the cluster are merged according to certain business logic, and the reduced results are presented to the final user.

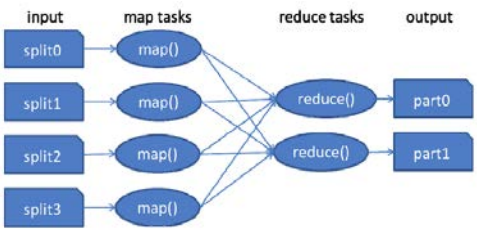



Figure 1: Map stage and Reduce stage of MapReduce

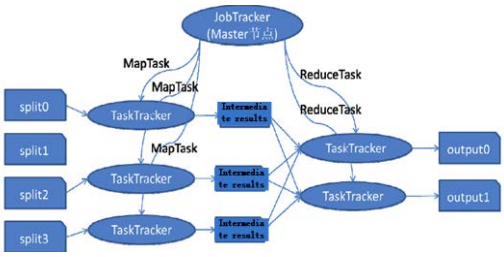



Figure 2: Distributed scheduling framework of calculation model

In the MapReduce distributed calculation model, MapReduce depends on the strong distributed scheduling strategy and fault-tolerant mechanism, providing a reliable distributed platform with strong expansibility. Figure 2 shows the distributed scheduling framework for MapReduce calculation model. In the MapReduce scheduling strategy, there are many unique characters to ensure the precision of distributed platform in the implementation process. For example, every node in the cluster can report the finished work and present state to Master node periodically; in this process, if the silent time of certain node exceeds preset time interval, Master node will mark the state of this node as dead, and distribute the data and tasks in this node to other nodes in order to resume.

## Apriori algorithm based on MapReduce

**Storage and calculation of distributed system:**The content in distributed system is rich, and there are professional operation system and program design language; certainly, distributed system also needs specific compiler and file system, and even distributed database system, and so on. However, hadoop is not the whole distributed system, whose submodule includes distributed storage system HDFS and database system HBase, and they are software system on the level of file system. As an open source project under Apache, Hadoop is composed of many projects, which is shown in Figure 3.

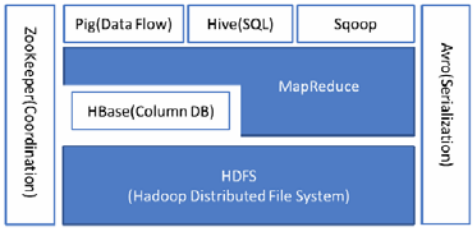



Figure 3: Hadoop subproject

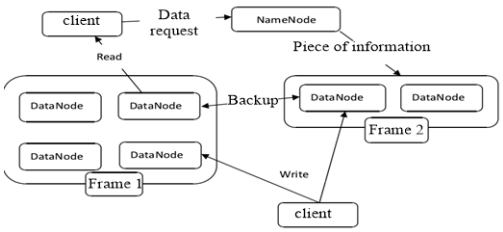



Figure 4: HDFS system structure

The core part of Hadoop is mapreduce and HDFS; mapreduce is responsible for the calculation of data; HDFS（Hadoop Distributed File System）provides file storage function for system. As is shown in Figure 4, HDFS adopts Master/Slaver structure model; HDFS cluster is constitutes of a NameNode and numerous DaTaNode nodes. NameNodes is responsible for managing DFS’s naming and visited operation; then, NameNode node can be seen as the main node in Master/Slaver

structure. Numerous DataNode nodes are mainly distributed on different nodes, which can upload data storage data and so on. Besides, customers can use HDFS to realize the storage mode of files conveniently.

**MapReduce association rules algorithm of global pruning:**Definition 1: DB is global database, DBi（i=1,2…n）is node block data, m(m<n) is the number of nodes, node is defined as n; local support degree is LUSpport=min_support*|DBi|; global support degree is set as min_support=2%, the support degree of item-sets is X. LUSpport>LUSpport defines X as local frequent item set, which will be global frequent item set if it is more than gobal support degree; the first threshold value of confidence coefficient is CONFIDENCE=65%，and if Y.itemConfidence>CONFIDENCE in the rule of X->Y, X->Y is strong association rules.

Definition 2: Strategy of frequent item set and rule production: first item set is produced by scanning object set, second item-set is produced by the first set since connection, third item-set are produced by first and second item-sets, and K item-set is produced successively, until the algorithm ends; association rules are produced by subset generation method.

Definition 3: Map and Reduce function parameter definition in the algorithm of MPAOR: suppose the form of Key/Value is the first step of MapReduce programming pattern, here is Key/Value definition of map function and reducer function: 1. in Map function: put in: Key：related to filename. Value: every line in the file; put out: Key: value is 0 (define in this way in order to put out to a reducer) Value: local frequent item set + space + occurrence number; put out: key : strong association rules. Value: confidence coefficient.

It can be seen that the subsets of frequent item sets in the local node Si must be frequent item set, and the supersets of infrequent item sets are definitely not frequent item sets; global frequent item-sets X must be local frequent item-sets on certain node Si(1<i<n), and all of the subsets of X are considered as local frequent item-sets. Suppose the local frequent item-set on Si is Li，(i=1,2…n), all of the nodes on Li are merged and set L is gained; and L is surely the superset of global frequent item-sets. Algorithm realization process is added into global pruning process, and at the same time improved algorithm also applies distributed processing strategy. The specific realization process is shown in Figure 5:

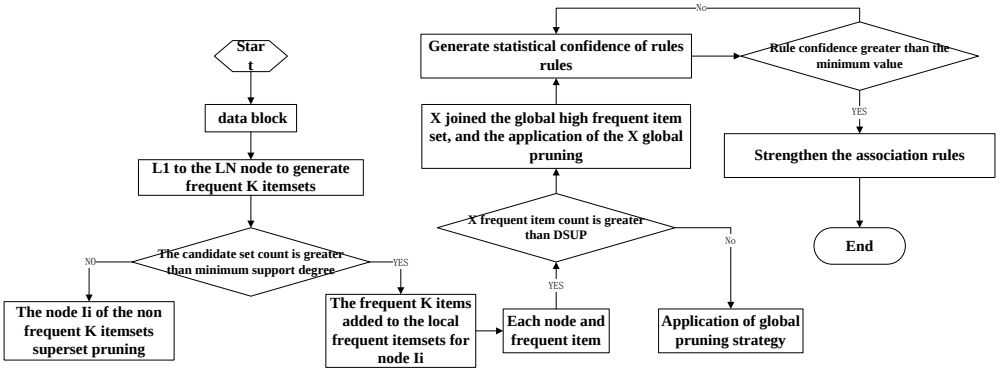



Figure 5: Algorithm processing chart

**MapReduce association rules algorithm of frequent matrix:**To make data matrix mining, matrix should be built first. The principle to that every line corresponds to on shopping basket; at the traversal of one shopping basket, if the corresponding object of the current element in this line appears, this element is set as 1, otherwise as 0, and in this way, there will not be the same shopping baskets in the matrix; the appearing frequency of certain object can use weight matrix, and the corresponding object appears again, the corresponding line of weight value in this line only needs to add 1. In this way, the unnecessary waste of internal storage can be reduced.

Definition 1: Suppose I={i1,i2,…,in} is the set of all the items. Every object is presented with Ti=(i1，i2,…,in) (1≤i≤n). Every object has to have its own mark.

Definition 2: Object set DB corresponds to R, whose definition is R=(DB1,DB2,…,DBn).

Definition 3: First set and first set matrix’s definitions: first set is defined as S1={Ik | Ik ε I}; and first set matrix R1 is defined as R1={S11,…S1n}T.

Definition 5: Second set and second set matrix’ s definitions: second set is defined as S2={Ii,Ij | Ii,Ij}; and first set matrix R1 is defined as R2={S21,…S2n}T. In the same way, K set and K set matrix can be gained.

The MFMDAP algorithm procedure is as following:

Data fragmentation process of input data

To control data fragmentation is to split object data on a level. MapReduce bank divides D into n data blocks with similar scales (this process is realized through InputFormat and data blocks are divided into InputSplit); they are sent to m nodes, and Map function is operated to execute tasks.

Local data piece transforms into matrix

Input data are sent to different computational nodes, and transformed to matrix; the data scanning for the first time produces first frequent set and local frequent matrix at the same time.

Production of local frequent set

Second frequent set is produced according to the formula in definition 3: DBij=DBi^DB, and at the same time, the support degree of frequent sets can be counted according to definition 3.

Generation of frequent k set: according to the formula of definition 4: DB12…k=DB1^DB2^… ^DBk and the generation of multi-sets also prune useless items of candidate frequent item set. At the node of every DataNode, local frequent matrix set generates.

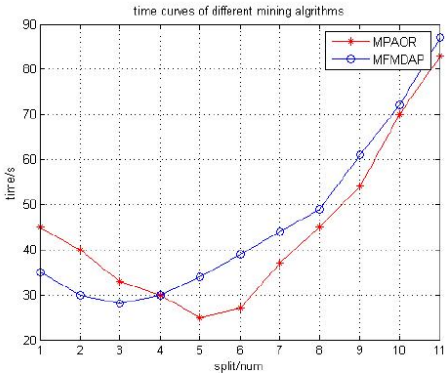
Local frequent matrix changes into local frequent sets

According to transformation rules, frequent matrix calculated at every node is transformed into local frequent set.

5. Generation association rules (1) all of the non-void proper subsets of random frequent sets are found. (2) As for the calculation confidence of every rule, confidence coefficient can be calculated through grouping, and the acquisition formula of confidence coefficient: itemConfidence=CountAll/countReason, which is the first rule found if it is greater than the minimum CONFIDENCE, and it goes on until all the association rules generate.

**Performance analysis**

Performance test aims to make more accurate evaluation on the performance of parallel Apriori algorithm at the same time of imposing pressure on it. There are 6 nodes in the groups adopted in this experiment, and 2/3 of the data set is chosen in the experiment; the number of MPAriori and MPAOP algorithm nodes are set as 4, with IP addresses are 222.27.254.166；222.27.254.25；222.27.254.152；and 222.27.254.90 respectively; in this experiment, suport=0.1;. As is shown in Figure 6, it is discovered that in the experiment, when the block number is 3 or 4, the efficiency of MFMDAP algorithm is good, and when it is 4 and 5, the efficiency of MFMDAP algorithm is relatively good. From this, it can be seen that when the data is smaller, the efficiency of MFMDAP algorithm is better, but when the data on single node is relatively large, the execution time of MFMDAP algorithm is relatively short. This also proves that FMDAP algorithm is more suitable for data with large granularity. As is shown in Figure 7, serial Apriori algorithm is dominant when the data size is small, but with the increase of data size, the advantage of MPAOP becomes more distinct. This also testifies that when the data size is small, distributed communication and data distribution will influence algorithm efficiency; however, with the increase of data size, the advantage of distributed algorithm MPAOP is more highlighted.



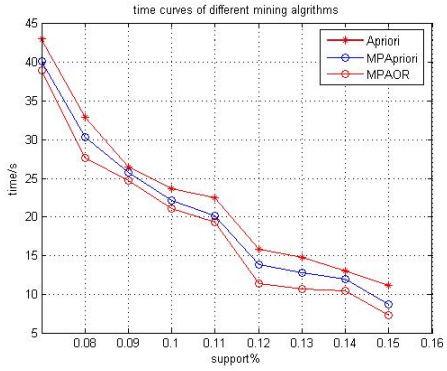




Figure 6: Algorithm operation timetable under different data blocks Figure 7: Chart of algorithm operation efficiency with different support degrees

## Conclusion

The core part of association rules algorithm is the generation of frequent items; Apriori algorithm has simple process and outstanding solving performance, which has gained wide recognition and application in data mining field. In the aspect of basic Apriori algorithm optimization, this paper makes deep study on pruning, frequent item storage mechanism and compute mode, and makes improvement and experiment experimental demonstration on aspects of algorithm solving performance and efficiency and so on; through analysis and comparison of experiment results, it is proved that improving algorithm can elevate the executive efficiency of algorithm. Under different data segmentation, MPAOR algorithm is suitable for dealing with data of small size on every node after segmentation, but MFMDAP algorithm is suitable for processing of big data with small blocks.

(The 2014 annual Guangxi University of science and technology research project (YB2014417) "research and Implementation on Algorithm of association rules based on Hadoop")

## References

[1] Shen Guoqiang. A highly efficient multi-dimensional association rules mining algorithm with multi-levels [J]. Computer engineering and applications, 2006，22(1):11-13.

[2] Zhang Feng. Study on multiple-valued association rules mining algorithm [D]. Xi’an: Xi’an University of Technology, 2010.

[3] Fayyad U，Piatetsky Shapiro G， Smyth P.The KDD Process for Extracting Useful Knowledge From Volumes of Data[J].Communications of the ACM，1996，39(11):27-34.

[4] R.Agrawal, J.Shafer. Parallel Mining of Association Rules:Design,Implementation,and Experience[J]. Technical Report, 1996..

[5] Wang Liwei. Summary of data mining research status [J]. Library and information, 2008(5) :1-11.

[6] Wang Yue. Study on distributed association rules mining methods [D]. Chongqing: Chongqing University, 2003.

[7] Chen Lei. Distributed data mining study based on cloud computing framework [J]. Chengdu University of Information Technology Journal, 2010,25(6) : 577-579.

[8] Li Lingjuan. Study on association rules mining algorithm in cloud computing environment [J]. Computer technology and development, 2011，21(2) : 43-50.

[9] Rong Xiang. Frequent item set mining methods based on MapReduce[J]. Xi’an Institute of Posts and Telecommunications Journal, 2011，16(4) : 37-39.

[10] Zhu Anzhu, Study on Apriori algorithm improvement and transplant based on Hadoop [D]. Wuhan, Huazhong University of Science and Technology, 2012.