

# STRUCTURE OF THE DISTRIBUTED DATA MINING SYSTEM BASED ON MULTI-AGENT

P.DINESH

*Asst. Professor/ CSE Department  
Meenakshi Ramaswamy Engineering College  
Ariyalur, Tamilnadu, India – 621 804  
dinesh006@gmail.com*

## ABSTRACT

*Data mining means process of extracting hidden pattern, previous unknown knowledge and rules with potential value to decision from mass data in database. Association rule mining is a main researching area of data mining area, which is widely used in practice. Distributed database is commonly used. Distributed data mining is mining overall knowledge which is useful for management and decision from database distributed in geography. It can not only improve the mining efficiency, reduce the transmitting amount of network data, but is also good for security and privacy of data. Based on raised distributed data mining system, this these brings about a new distributed association rule mining algorithm DK-tree algorithm. DK-tree algorithm is based on the basic theory of twice knowledge combination, This algorithm just needs three times communication between the main and sub-site points, which greatly reduces the amount and times of communication, and improves the efficiency of selection. What's more, each sub-site point can fully use existing good centralized association rule mining algorithm to realize local association rule mining. The structure of distributed data mining application system, distributed association rule algorithms, which reduces the workload of algorithm analysis.*

**Key words:** data mining, distributed, multi-agent, Association rule, DK-tree algorithm

## 1. INTRODUCTION

Data Mining is a interdisciplinary study which integrates the results of the newest technologies such as Database technology, Artificial Intelligence Learning of the machine, Statistics, Knowledge Engineering, Object-Oriented Methods Information Retrieval, High-Performance Computing and Data Visualization. However, database technology, as a kind form of information storage and management, **OLTP (On-Line Transaction Processing)** as core application, it lacks of supporting mechanism of higher function such as decision-making, analysis, and forecasting, and so on.

**OLAP (Online Analytic Processing)** Decision Support and Classification, data mining and knowledge discovery emerge and show powerful vitality. Data mining and knowledge discovery help data handling technology into a higher stage, which can not only query the past data, but also can find the potential association decision and forecast future developing trend. Considering that the data is processed from a distributed database.

Data mining system often the needs data from database from different sites, which require the data mining system have the ability of distributed mining, and Mean while, we should design new distributed data mining algorithm according to the features of distributed data mining.

## 2. The Distributed Data Mining System (DDMS)

Specifically data mining can be regarded as a forecasting model or rule set got from one or more (distributed) data into the collections applying corresponding data mining algorithm. Here different strategies can be used mainly according to the data themselves, the distribution of the data, the software and hardware resources that can be used, and the required precision. Accordingly, the centralized distributed data mining systems have some differences in the following strategies [79, 80].

### (I)Data Strategy (DS)

The distributed data mining can be choose the final result of moving data, or moving middle result, or providing forecasting model, or moving data mining algorithm. We can use the distributed data mining system of Local Learning to establish models in each distributed places, and then carry these models to a centre region. We can also use the data mining system of Centralized Learning to carry the data to the centre region and then establish models. Besides, some data mining systems use Hybrid Learning, i.e. the strategy combining partial leaning and the centralized leaning process. Distributed data mining system should have good performance in Scalability.

## (2) Task Strategy (TS)

The distributed data mining system can choose to co-coordinately use one kind of data mining algorithm in several data stations, and can also choose to use different data mining algorithms independently in each data station. In the mode of independent Learning, each kind of data mining algorithm is respectively applied in each distributed data station; in the mode of Coordinated Learning, one (or more) data station use one kind of data mining algorithm to coordinate mining task in the Several data stations.

## (3) Model Strategy (MS)

There are many methods of combining the forecasting models established in different places. Among these methods, the simple and the most often used one is making use of voting, which is to combine the output of the models of each type. According to the majority voting. But the method of Knowledge Probing. Comprehensive model according to the input and output of all kinds of. The extensibility of the distributed data mining system is such a kind of ability of the system: when the number of the data sites is increasing, the performance of the system has no substantive and obvious declining. The effectiveness means to make use of the centralized system resources effectively and get the correct mining results.

The Portability refers to that a distributed data mining system should normally operate in the multi-environment with software and hardware equipments, and can combine multi-model with different expressions. Almost all the environment of most data mining systems will change. The Adaptively of the distributed data mining system refers to the ability how to evolve and adjust according to the changed environment.

## 2.1 Agent Introduction

The management information of agent and its relevant concepts is distributed artificial intelligence [81] its basic theory was brought about by John. McCarthy [82] in 1950s. And now it is widely used in areas such as process control, production-manufacturing, of the information management, intelligent database, data mining, and network management and e-business. Agent technology is a new algorithm model, which is highly intelligent, easy to construct distributed system and having strong reusability.

## 2.2 The characteristics and definition of Agent

The two characteristics of Agent is intelligent and acting ability. Intelligent means the ability to use reasoning, learning, and other skills to analyze and explain various information and knowledge

which it meets or receives. Generally speaking, Agent should have the following four basic characteristics (83):

**(1) Autonomy:** agent can be operated without the intervening of people or other agents. Also, agent can control its own behavior and inner situation; Environment.

**(2) Reactivity:** Agent can sense and understand its environment, time to the changes of environment.

**(3) Pro-activeness:** can not only respond to environment, it can also adopt behavior to face the objective through receiving some starting information.

**(4) Scalability:** Agent with sociability is very friendly. It has good social relationship diffuse skills. Agents can communicate with each other by agent language.

## 2.3. Multi-Agent System (MAS)

Multi-agent system (MAS) is a system composed of multiple interacting intelligent agents. Multi-agent systems can be used to solve problems which are difficult or impossible for an individual agent distributed data mining system. For example, Centralized data mining system middle-learning distributed data mining system— JAM [89] traditional data mining method, mining with multi-techniques; integrate results from each local site point, and achieve results in overall situation.

### 2.3.1 Distributed Data Mining System Based On Multi-Agent (DDMSBMA)

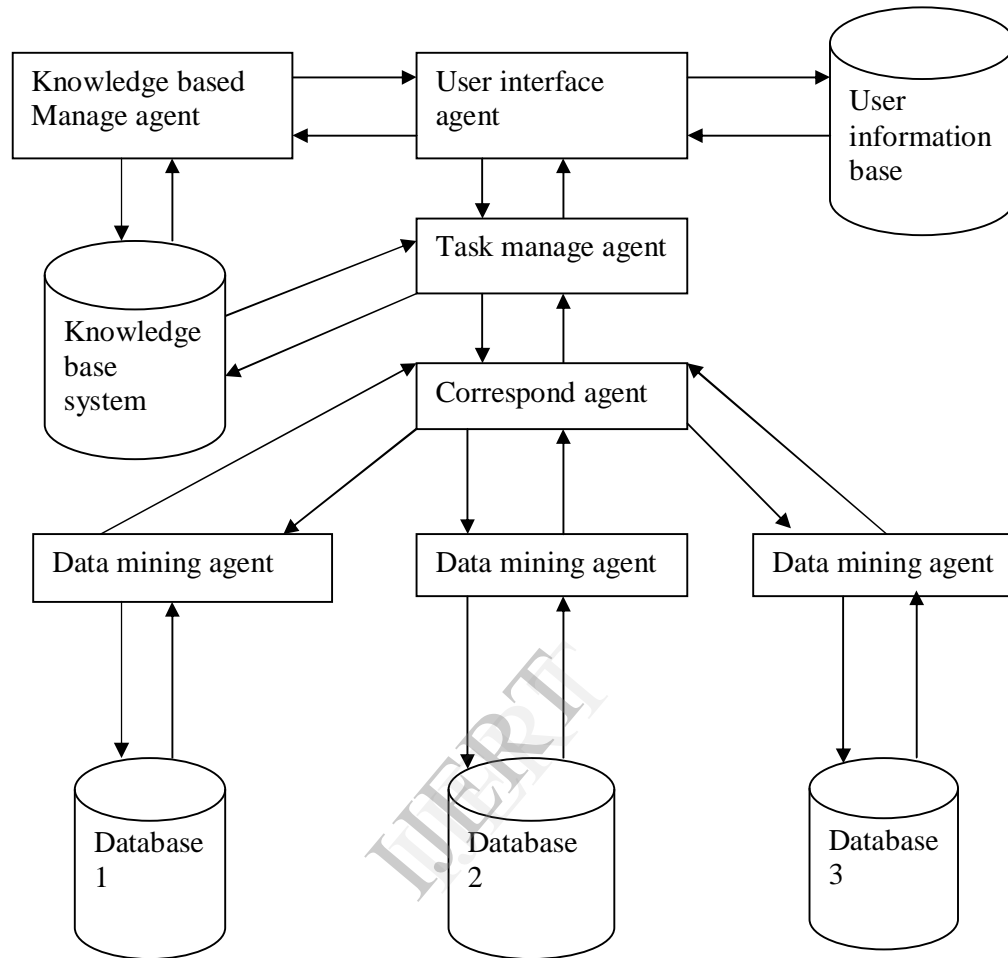
Distributed data mining system which is based on multi-agent. This system can not only mine local data information, but can also do distributed data mining in different data site point. It is composed of the Users' interface agent, users' information base, knowledge management agent, and task Management agent, the overall knowledge base, coordinating machine agent, and data mining agent.

### 2.3.2 Structure of Distributed Data Mining System Based On Multi-Agent

Structure of the DDMS need to pro-active and autonomous based on .Agents act dynamically based on the conditions and interacts with each other. The complex domains knowledge is applied on distributed web sites. The knowledge of agents in some application are auctioned .among the website In a multi-agent system, this knowledge is usually collective .This collective intelligence of multi agent system must be developed by the distributed domain knowledge and distributed data, A sensor network that Distributed Data Mining (DDMS) algorithms and Multiple Agent systems (MAS) are (Buhalis, 2003): improved capacity management and operations efficiency; Usually, Data Mining systems are built using client-server of the architecture, with different

distribution on the two components of the items below.

sample the database, and then mine the sampling data to increase mining efficiency.



**Figure: 1 Structure of Distributed Data Mining System Based On Multi Agent**

Data pre-handling is an important step in the process of data mining (knowledge Discovery), especially when mining data that contain yawp or are incomplete, even Conflicting, it needs data pre-handling even more in order to increase the quality of the data mining objects, and finally achieve the aim of increasing the quality of the mode knowledge obtained by data mining.

In the process of mining, it coordinates the information transferring among the data mining g nets. When it needs to make the mining task known to some data mining agent, it should first test whether this mining agent is busy or not. If this mining agent is not busy, then make the mining task known immediately; or else, wait until the present mining task is over, then make the new mining task known Transferring the metadata, and providing the overall sharing. Sampling is first to

### 2.3.2 Module Function:

Each function of module system is composed of multi-agent. The agents Coordinate with each other to realize functions of the system. Therefore, this system is the distributed data mining system based on multi-agent. In the following, we will introduce the main functions of each module:

#### 2.3.2.1 Users' Interface Agent

It is used to realize the alternation between users and the computers. The communication between users and the system is finished through the users' interface agent. The users need not to communicate with other agents.

#### 2.3.2.2 Task Management Agent

Task management agent is used to establish, manage, start-up and execute data mining task. Making use of task management agent can pack the handling steps needed by the data mining task

including data picking-up manner, data pre-handling manner, data dispersing method, data mining algorithm, and so on, into a data mining task. Task management agent can manage this mining task, and through task management agent, start-up the mining task, and record the state information of mining execution. This is in favor of the establishment, management, repeat execution of mining task, and so on. Specifically speaking, task management agent has the following functions:

Determining which focus on kind of algorithm will be used (in this system, the mining algorithm applied in data mining sub-agent is not the only. We set many kinds of effective mining algorithm suitable for different conditions in it), and making the task known to the corresponding agent. Result of mining every time for the future comprehensive analysis of the mining knowledge manner.

#### 2.3.2.3 Correspond Agent

After receiving the agent mining task given by task management agent, according to the specific situation, it makes the task known to the corresponding data mining agent; after mining, it receives the mining results transmitted from mining agent, and then transmits them to the task management agent;

In the process of mining, it coordinates the information transferring among the data mining agents. When it needs to make the mining task known to some data mining agent, it should first test whether this mining agent is busy or not. If this Mining agents is not busy, and then makes the mining task known immediately; or else, wait until the present mining task is over, and then make the new mining task known.

#### 2.3.2.4 Data Mining Agent (DMA)

DMA is the core of the system, which can realize the data mining analysis of the local database. It mainly includes three functions: data picking-up, data pre-handling, and data mining. They can be expressed by different sub-agents. Data picking-up and data pre-handling are data preparing process. Data preparing is a very important link in the data mining process.

Data pre-handling includes data cleaning, data integrating and data dispersing. Data cleaning is to handle the prepermission in data and clean dirty data. Data integrating is to combine and handle the data of multi-data source to solve Semantic fuzziness and conform into coherent data storing.

#### 2.3.2.5 Knowledge Management Agent

It provides the managing function of the data mining result knowledge, stores mining result into the overall knowledge base, and displays mining results, and provides the interface that the experts

estimate the mining results and according to the expert analyzing result, deletes some useless rules. Estimating and testing the results are the indispensable part in the whole process, including the mode explaining discovery.

#### 2.3.2.6 Users' Information Base

There are two kinds of information base storing in the users; information base; one issuers' managing information that is used to enrol, maintain and manage the users' information, and authenticate the users' logging, and setup the purview and PRI. The other is the information about users' interest and hobbies that is regarded as the reasoning rules used for the alternation between the users' interface and users.

#### 2.3.2.7 The overall Knowledge Base

The results of the data mining knowledge base not only can be provided to the users through person-computer interface, but also can be stored into the overall knowledge base for the future further analysis. Because the model diversity of the mining, the representation form of the knowledge will be different, having no uniform forms. Therefore, it can setup a table for each kind of mining algorithm to store the knowledge got through this algorithm.

#### 2.3.3 Work Process System

In this system, data mining agent DMA is responsible to store-extract data and mine higher-level users' information from data. DMA works in the parallel form. The coordinator is used to communicate and share information between DMA. Coordinator collaborative agent provides information to users, and feedbacks the users' information to the agent.

The basic work principle of the system is as follows:

The users (who have passed the identification test of users' mining agent) give out the mining requirements;

The task managing agent accepts mining requirements, and packages the mining requirements according to the scheduled format and then transmits it to the coordinator.

The coordinator analyzes the mining requirements and fixes the involved DMA.

DDMA mines automatically the corresponding information according to the mining requirements; The coordinator collects the corresponding information from each DMA, and then analyzes it comprehensively, and gets the final result information.

Task managing agent submits the result information to the users through the users' interface agents.

### 3. Association Rules

The association rule mining is an important research topic in the field of data mining, and the important content of KDD data research[52] It is put forward in the environment of supermarket data with the motive of discovering how each kind of the mode of users' buying behavior through association rule algorithmic 1993, Agrawal, etc. first put forward the problem of the association rule of the items in mining users' trade databaset53] From then on, many researchers have done research on the mining problem of association rules. Their work includes the optimization of the original algorithms, such as introducing random sampling, Distributed, parallel ideas, and so on, to increase the efficiency of the algorithm of31mining rule and promote the application of the association rule. At present, the mode identification, and so on. With the deep of the research and application promotion of the association rule mining algorithm.

#### 3.1 Basic conception and problem describing

Set  $I = \{i_1, i_2, \dots, i_m\}$  are item set, in which the element is item, marked  $D$  as the aggregate of (transaction)  $T$ , here the transaction  $T$  is the item set,

Every transaction has its unique identity, such as transaction number, marked TID.

Set  $X$  is an aggregate of item in  $I$ , if  $X \subseteq T$ , then transaction  $T$  included the  $X$ .

A association rule is a containing formula like  $X \rightarrow Y$ , here  $X \subseteq I$ ,  $Y \cap X = \emptyset$ , and  $X \cup Y = C_p$ .

The support of the rule  $X \rightarrow Y$  in transaction data base  $D$  is the ratio of  $X$  and  $Y$  data and all the transaction data, it is called  $\text{support}(X \rightarrow Y)$ :

$\text{Support}(X \rightarrow Y) = \frac{| \{ T \mid X \cup Y \subseteq T, T \in D \} |}{| D |}$  (2.1)

A association rule is a containing formula like  $X \rightarrow Y$ , here  $X \subseteq I$ ,  $Y \cap X = \emptyset$ , and  $X \cup Y = C_p$ . The support of the rule  $X \rightarrow Y$  in transaction data base  $D$  is the ratio of  $X$ .

#### 3.2 Association Rule Mining Algorithms

The mining of the association rule has made remarkable achievements, and already put forward many good association rule mining algorithms. According to the application environment we classify them into three types:

One is the association rule mining algorithm used in centralized database system [60]. This kind of algorithm includes AIS, Apriori, AprioriTid, Apriori Hybrid [65] put forward by Agrawal, etc. and DHP [66] put forward by Park, etc., and the dividing algorithm PARTITION [67] put forward by Sava sere, etc., and the sampling [68] algorithm put forward by Tovionen, etc., and some updating

algorithms of the association rule such as FUP, IUA and NEWIUA, and so on. Among them, the basic idea of Apriori algorithm is scanning database repeatedly.

The length of scanning in the  $k$ th time is the big itemset  $L_k$ ; when scanning in the  $k+1$ th time, on the basis of  $k$  itemset in  $L_k$ , the candidate collection  $C_{k+1}$  will be produced; DHP algorithm uses Hashing technology to improve the Producing process of the standby itemset  $C_k$ ; the algorithm PARTITION is to divide the database, reducing the I/O times in the mining process; the algorithm.

The second one is the algorithms solving the problems of the association rule mining in the parallel environment: the CD (Count Distribution) put forward by Agrawal, etc., DD (Data Distribution), CAD(Candidate Distribution) [69] and PDM(efficient Parallel Data Mining for association rules ) [73] put forward by Park, etc., and soon.

All these algorithms are based on the algorithm Apriori. The precondition is that the processor contains special memory and disk and there is no region that can be shared in structure. The processor is joined by communication network, and the information transmitting is used for communicating; and data are allotted evenly To the special disk of each processor.

The algorithm CAD combines CD and DD algorithm. When generating one item collection, we use CD algorithm or DD algorithm. But when generating the following  $k$  ( $k > 1$ ) item collection, the algorithm allots the frequent item collection  $U_{k-1}$  and also e-allots the transaction database; when generating  $C_k$ , it should separate it from other processor, leaving them to be handled in the next pruning time. Although CAD algorithm avoids large quantity information transmitting, its efficiency is not so ideal due to the e-allocation of the transaction database.

The third one is the algorithm solving the problems of the association rule mining in the distributed environment, such as DMA [71], FDM, etc. [72,78]. The design of algorithm DMA is based on the principle that if the item collection  $X$  in  $DB$  is big Item collection, then it must also be big item collection in some  $DB$ ". The algorithm uses local pruning technology to generate the candidate big item collection which is smaller than that of the algorithm CD. When each sites exchange supporting number, the algorithm uses polling site technology to make the communication cost of each tern collection  $X$  degrade to  $O(n)$  from  $O(n^2)$  of the algorithm CD, and  $n$  is the number of the site. Although the algorithm DMA overcomes some weakness of the algorithm CD, it needs the supporting number of all the other sites when

generating K frequent big itemset, having more synchronization times with other sites. The algorithms FDM and DMA are almost the same. The difference is only that FDM adds the overall Pruning technology. Unnecessary communication and data transmitting as much as possible.

**4. DK-Tree Algorithm**

Distributed data mining system based on multi-agent. The realization of a better distributed mining system will depend on a high-effective algorithm. The association rule mining is a kind of data mining algorithm distributed association rule mining algorithm whose communication cost is lower—DK-tree algorithm.

**4.1 The Basic Principle of DK -Tree**

DK -tree algorithm has following steps:

- (1) each site point adopts local-mining, and then gets local rules set  $R(i | i=1, 2, \dots, n)$ , in which local-mining use Apriori algorithm to realize association mining.
- (2) Local rules set  $R(i)$  resulted from each site point is sent to the main controlling site point as results. The main site point builds an overall rules knowledge database, which is used to collect all rules sent by sub-site points, and reflect them onto an association tree. Then an association rules tree—DK-tree is generated.
- (3)DK-tree includes all association rules database. And final association rules will be mined in this overall database.
- (4)Compare rules mining from site point 2 with rules in the tree. If it is same with a certain rule in the tree, the new branch with not be created. Then the Record this rule's appearing times as 1, and record it into next rule. If not, then create a new branch, and record appearing times as 1;
- (5)With the given value N— the least appearing times rule, scan through the rule tree formed by overall rules base. And then compare the appearing times of each rule recorded in rule tree, delete branch whose times is smaller than value N, and delete correspondent rule from rule knowledge base.
- (6)Because what we want is overall association rule, apparently this rule should exist commonly in all sub-site points database. If some rule is generated just from certain site points, them it is definitely not overall rule. Therefore, the selection of rule times value N will influence directly this algorithm's speed and rate of convergence.
- (7) After deleting all branches smaller than value N, scan all sub-site points' database again, and obtain information of left branch rules in each sub-site point, such as supporting rate, confidence coefficient, and event number of supports.

**4.2 The Basic Concept and Theory**

Distributed database system S that is composed of n sites  $s_1, s_2, s_3 \dots s_n$ . DB is the distributed database of S.

The database in the station  $s_1$  is  $DB_1$ .  $DB = DB_1 \cup DB_2 \cup \dots \cup DB_n$ . D and  $D_i$  respectively stand for the size of the database in DB and  $D_i$ ,  $D = D_1 + D_2 + \dots + D_n$ . DB is called overall database and  $DB_i$  is called local database.

- 1) Each site point adopts local-mining, and then gets local rules set  $D(i | i=1, 2, \dots, n)$ , in which local-mining use Apriori algorithm to realize association mining.
- 2) The distributed database  $DB = \{DB_1, DB_2, DB_n\}$ ; The threshold of the minimum supporting degree min-sup; the threshold of the minimum confidence
- (3) For all sites do
  - {
  - (4) Each sub-point adopts Aprior algorithm local-mining to generate local rule set  $D(i)$ ;
  - (5) Site point  $i$  sends each rule's attribute information generated from rule set  $R(i)$  and  $D(i)$  to the center;
  - }
  - (6) Establishing rule knowledge base D; storing the entire rule set  $D(i)$  transferred from each site in D; For  $i = 1$  to  $n$  do
  - (7) if  $n(i) < N$  then delete this branch
  - (8) Else scanning the database again, getting the information of each sub-sites in the left branch rule in the DK -tree; through the information calculating the supporting degree DK -tree and the believing degree confi.
  - {
  - (9) If  $sup_i < min-sup$  then delete this branch
  - (10) Else
  - (11) If  $conf_i < min-conf$  then delete this branch
  - (12) Else this rule is the overall rule, output
  - (13) End
  - }
  - }
  - (14) End

**4.3 Example of DK-Tree Algorithm**

In order to make readers understand more deeply, we explain a simple example with DK -tree algorithm specifically.

We give the mining rules of three sub-sites storing in the rule database D. We use DK -tree algorithm to these rules to find the overall rule.

Suppose the transaction number that each site transfers is: site 1: 2500; site 2:2750; site 3: 3000)

Rule of station 1	
Num	Rule
1	a=5 b=3 c=2 d=1
2	b=5 c=3 d=4
3	C=4 d=5 a=2 b=3
4	b=2 a=5 d=1 c=4

Rule of station 2	
Num	Rule
1	b=2 c=4 a=3
2	a=5 b=3 c=2 d=3
3	b=5 c=3 d=4
4	c=4 d=5 a=2 b=3

Rule of station 3	
Num	Rule
1	b=2 a=5 d=1 c=3
2	a=5 b=3 c=2 d=1
3	b=5 c=3 d=4
4	d=2 c=4 b=5

Rule{a=5 b=3 c=2 }=>d=1 information				
station	Supi	confi	Transaction support number	Transaction number the former piece
Station1	0.16	0.8	4250	5312
Station2	0.43	0.5	825	1833
Station3	0.15	0.16	4500	6712

4.1Table: The rule of the sub-site in rule knowledge base D.

First, we construct the rule of the site 1 in rule knowledge base D into the following DK -tree according to the constructing method mentioned above.

Establish the root node of the "null" of the tree, and each rule constitutes a branch according to the order that the former piece - > the latter piece. The latter one is the leaf node of the branch, and record the appearing times of this rule at the place of leaf node.

First, create a root node of tree, and mark it with "null";

D start scanning rules knowledge base D, D is an overall rules base which is formed by collections from each sub-site point.

It creates a branch in tree with each rule mined from site point 1 according to the order of first component ( P ) — >consequent ( B ) .

(1)The rule's consequent is the leaf node of branch, and record this rule's appearing times as 1 at the leaf node

(2) Here, we regulate  $N=2\text{minsup}=0.1/\text{minconf}=0.7$  then

{a=5 b=3 c=2}=>d=1 %41,  $n=2=N$ , keep this branch;

{a=5 b=3 c=2}=>d=3% 4\*,  $n=1<N$ , delete this branch;

{b=5 c=3}=> d=4 %,  $n=3>N$ , keep this branch;

{c=4 d=5 a=2}=> b=3 %,  $n=3>N$ , keep this branch;

{b=2 c=4}=> a=3%,  $n = 1 <N$ , delete this branch;

{b=2 a=5 d=1}=> c=4%  $n=1<N$ , delete this branch;

{b=2a=5d=1}=> c=3 %,  $n=1<N$ , delete this branch;

{a=3c=6 d=5}=> b=4%,  $n=1<N$ , delete this branch;

{b=1a=3d=2}=> c=1%,  $n = 1 <N$ , delete this branch;

{d=2 c=4}=>b=5%,  $n=1<N$ , delete this branch;

Scanning the database of these three stations again to find the information of the left branch in the three sub-sites, the findings are as follows are station and station information rule are represented DK tree algorithm.

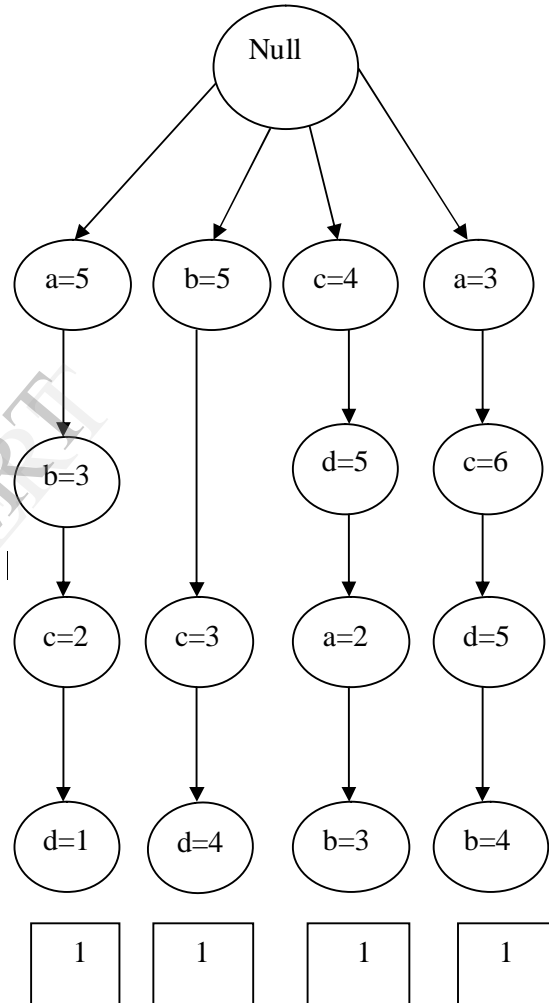


Figure 2: DK -tree formed by rule of the site 1

(3) Then, begin to read each rule from the site 2. if it is the same as the existing rules in the tree, then add one to the appearing times of this rule; or else, Establish another branch. According to this method read all rules.

Distributed association rule mining algorithm based On multi-agent—DK-tree algorithm. We can see that it greatly reduces the network communication cost to use such algorithm to do distribute association rule mining.

It can calculate the supporting degree and the believing degree of each rule and get high-effective and reliable mining results that users need only by mapping the mining results of the sub-sites to a association tree and through the basic information of each rule.

In the whole in the mining process, its only needs communication two times in sub-stations and the main station, transferring is less(only the mining results of the sub-sites needs transferring), and the r requirement to the network bandwidth is low; the mining efficiency is high; the Security and privacy of data have been guaranteed. As a distributed association rule mining algorithm, DK-tree has good practicability.

## CONCLUSION

In this paper distributed data mining comes into being as time requires. The homogeneous and the diversity of data are one of the different problems in distributed data mining. It also addresses the issue of handling of dynamically generated data sets. Since this approach does not require huge amount of data transfer from remote to central site, the network resources are used optimally. The paper gives the brief description of the role played by agents in DDM that will result in an increase in the research work of Distributed Association Rule Mining. Distributed association rule algorithm based on multi-agent, and analyzes, compares this result based an algorithm.

Multi agents are used in the DDM for decision for Developing algorithms and architectures that will be work on real data sets for Distributed Association Rule Mining. Future algorithms and methods should also consider the development of adaptive, fault-tolerant and easily extendable systems in the area of agent-based distributed association rule mining. DK-tree algorithm is used mining in each sub-site point. Each sub-site point can do off-line mining, which will improve the efficiency of data mining.

In the future work, we will discuss about other distributed mining algorithms like classification, and soon.

## REFERENCES

- [1] R. Agrawal and R. Srikant, "Fast Algorithms for Mining Association Rules in Large Databases," Proc. 20th Int'l Conf. Very Large Data Bases, pp. 487-499, 2004.
- [2] N. Panda, P. K. Sahu and Trilok Nath Pandey, "Improving the Performance of Distributed Data Mining with Multi-Agent System," *IJCSI*, vol. 9, no. 2, No. 3, March 2012.
- [3] O. Folorunso, A. S. Sodiya, G.O. Ogunleye A.O. Ogunde, "A Review of Some Issues and Challenges in Current Agent Based Distributed Association Rule Mining," *Medwell Journals*, vol. 10, no. 2, pp. 84-95, 2011.
- [4] E. George Dharma Prakash Raj and A. S. Raja,

"Mobile Agent Based Distributed Association Rule Mining: A Sur-vey," *IJRRCS*, vol. 3, no. 5, October 2012.

- [5] J. Dasilva, C. Giannella, R. Bhargava, H. Kargupta, and M.Klusch. Distributed data mining and agents. *Engineering Applications of Artificial Intelligence*, 18(7):791-807, October 2005.
- [6] C. Giannella, R. Bhargava, and H. Kargupta. Multi-agent Systems and Distributed Data Mining. *Lecture Notes in Computer Science*, pages 1-15, 2004.
- [7] M.J. Zaki. Parallel and Distributed Association Mining: A Survey. *IEEE Concurrency*, vol. 7, no. 4, pages 14-25
- [8] M. Wooldridge. An Introduction to Multi Agent Systems. John Wiley & Sons Ltd. 2005.
- [9] Kargupta H. and Chan P. (editors). *Advances in Distributed and Parallel Knowledge Discovery*. AAAI press, Menlo Park, CA, 2000
- [10] Cho, V. & Wüthrich, B. (2004). Distributed Mining of Classification Rules. *Knowledge and Information Systems*, 4, 1-30
- [11] Cannataro, M. and Talia, D. (2005). The Knowledge Grid. *Communications of the ACM*, 46(1), 89-93.
- [12] S. Datta, K. Bhaduri, C. Giannella, R. Wolff, and H. Kargupta. Distributed data mining in peer-to-peer networks. *Internet Computing, IEEE*, 10(4):18-26, 2006.
- [13] T. Marwala and E. Hurwitz. Multi-Agent modeling using intelligent agents in a game of Lerp. eprint arXiv: 0706.0280, 2007.
- [14] M. Deshpande, M. Kuramochi, and G. Karypis. Automated approaches for classifying structures. Proceedings of Workshop on Data Mining in Bioinformatics (BioKDD), pages 11-18, 2002.
- [15] V. S. Ananthanarayana, D. K. Subramanian, and M. IM. Murty. Scalable, Distributed and Dynamic Mining of Association Rules. In Proceedings of HIPC'00, pages 559-566, Bangalore, India, 2000.
- [16] V. Gorodetsky, O. Karsaev, and V. Samoilov. Infrastructural Issues for Agent-Based Distributed Learning. In *Proceedings of the 2006 IEEE/WIC/ACM international conference on Web Intelligence and Intelligent Agent Technology*, pages 3-6. IEEE Computer Society Washington, DC, USA, 2006.
- [17] L. Cao, C. Luo, and C. Zhang. Agent-Mining Interaction: An Emerging Area. *Lecture Notes in Computer Science*, 4476:60, 2007.