Performance Evaluation of Algorithms using a Distributed Data Mining Frame Work based on Association Rule Mining

P.T.Kavitha

Research Scholar,
Sathyabama University,
Chennai, India
Email: [1] kavithapt@yahoo.com

Dr.T.Sasipraba

Dean,
Sathyabama University,
Chennai, India

***Abstract*— Numerous current data mining tasks can be implemented effectively only in a distributed data mining. Thus distributed data mining has achieved significant importance in the last decade. The proposed distributed data mining application framework, is a data mining tool. This framework aims at developing an efficient association rule mining tool to support effective decision making. Association Rule mining focuses on finding interesting patterns from huge amount of data available in the data warehouses. In order to build strong association rules, it depends on the extraction of association rules by Apriori algorithm, AprioriTID algorithm, AprioriHyprid algorithm, FP growth etc. The efficiency of the distributed data mining framework is determined based on the selection of the algorithm. The object oriented implementation has enabled the system to be platform independent. The use of self defined database format gives an upper hand for the system by operating efficiently without any need for third party database drivers. The mined results can be compared and graphically projected. Finally, some expectations for future work are presented where various modes of graphical representations can be included.**

Keywords-Association Rule Mining, Frequent Intemset, Frequent Pattern Mining, Distributed Data Mining

I. INTRODUCTION

Recent advances in computing, communications, and digital storage technologies, together with the development of high-throughput data-acquisition technologies, have made it possible to gather and store incredible volumes of data [16]. Complex distributed systems (computer systems, communication networks, and power systems, for example) are equipped with sensors and measurement devices that gather and store a variety of data for use in monitoring, controlling, and improving their operations [15].

These complex data distributed systems are to be mined to gain greater insight. Every business needs to manage and mine its data to gain better understanding about customer's preferences and behavior, products and services so that decision makers can make profitable decisions faster than their competitors. Data deluge and the lack of appropriate automated knowledge creation processes lead to a "knowledge gap". Data mining represents the most effective way to close the knowledge gap. It helps decision makers extract intelligence from a huge amount of data for better understanding and leverages the data warehouse and business data mart [17].

Most of the distributed databases contain transaction data. Within these transaction data, there is hidden affinities among products which tend to sell well together are found and this is called as market basket analysis or product affinity analysis. Association Rule Mining (ARM) has become one of the core data mining tasks and has attracted tremendous interest among data mining researchers when dealing with distributed data bases. ARM is an undirected or unsupervised data mining technique which works on variable length data, and produces clear and understandable results [14].

The computer systems have been moving from centralized massive computing devices supporting stationary applications, into client-server architecture that supports complex forms of distributed computing. Throughout this process restricted forms of code mobility have existed. A new type of evolution is now under

way that goes one step further, allowing complete mobility of applications among supporting platforms to form a large-scale, loosely-coupled distributed system and making parallel computing feasible.

Analysis of past transaction data can provide very valuable information on customer behaviour and business decisions [3]. The problem of extracting the knowledge is a difficult task for large datasets due to their static nature and geographical distribution datasets (same logical datasets are physically located faraway from each other)[1].

Due to these properties, algorithms that handle large datasets cannot assume or control the partitioned structure, the sizes, and the locations of the pieces of the datasets and must take account of the latencies and bandwidth required to move data among the places.

This paper uses Java platform to provide user interactive platform with all the components. The Distributed Clients or sites are used for processing data in a totally decentralized manner. We propose a JAVA based DDM framework, a totally decentralized framework for distributed data mining using association rules as the back bone of the system.

This system is completely platform independent including the database support. The use of client-server architecture enables us to perform distributed data mining. We define access rights to this framework by classifying users into groups. We can add or remove algorithms at any client side dynamically. The Benchmarking module evaluates performance between the algorithms. Thus the complete platform independency could be achieved using object oriented programming.

This paper is organized as follows: The definition, process and technique of data mining are introduced in section I, motivations and related work are explained in section II, the proposed distributed data mining framework using Java platform is given in section III and section IV explains the experiments conducted. Section V describes the conclusion and the future work.

II. MOTIVATIONS AND RELATED WORK

The amount of data tends to grow up at an vast rate and the primary challenge is how to make the database a competitive business advantage by converting seemingly meaningless data into useful information.

By knowledge discovery in databases, interesting knowledge, regularities, or high-level information can be extracted from the relevant sets of data in databases and be investigated from different angles, and large databases thereby serve as rich and reliable sources for knowledge generation and verification. Mining information and knowledge from large database has been recognized by many researchers as a key research topic in database systems and machine learning. Companies in Many industries also take knowledge discovery as an important area with an opportunity of major revenue. The discovered knowledge can be applied to information management, query processing, decision making, process control, and many other applications.

Surveys of data mining for business applications in various domains have shown that business people cannot effectively take over and interpret the identified patterns for business use. This may result from several aspects of challenges besides the dynamic environment enclosing constraints.

1) There are often many patterns mined but they are not informative and transparent to business people who do not know which are truly interesting and operable for their businesses.

2) A large proportion of the identified patterns may be either commonsense or of no particular interest to business needs. Business people feel confused by why and how they should care about those findings.

3) Further, business people often do not know, and are also not informed, how to interpret them and what straightforward actions can be taken on them to support business decision-making and operation.

4) It has been increasingly recognized that traditional data mining is facing crucial problems in satisfying user preferences and business needs.[11]

5) FP-Growth is one of the most popular methods for ARM; it, however, exhibits poor spatial and temporal localities. We argue that it is possible to improve spatial and temporal localities of the mining task [18].

6) Data mining frameworks are normally designed more for power and flexibility than for simplicity [19].

7) Most of the data mining frameworks expect the users to possess a certain amount of expertise in order to find the right settings [19].

There are several data mining tool framework are available in the market. Five of the most highly acclaimed data mining tools are so compared on a fraud detection application, with descriptions of their distinctive strengths and weaknesses and the table 1 shows the comparison based on the algorithms implemented [20].

Table 1. Comparison of Data Mining Algorithms

Algorithm	IBM	ISL	SAS	TMC	Unica
Decision Trees	√	√	√	√	
Neural Networks	√	v	√	√	√
Regression		√	√		√
Radial Basis Functions	√				√
Nearest Neighbour			√	√	√
Nearest Mean					√
Kohonen self organizing maps		√	√		
Clustering	√	√			√
Association Rules	√	√			

Another research work shows the comparison of usage of algorithms (Table 2) in the mining frameworks [21].

Table 2. Comparison of Data Mining Tools

Algorithm	Clementine	Enterprise Miner	Intelligent Miner	Mineset	CART
Decision Trees	√	√	√	√	√
Linear/Statistical	√	√	√		
Multilayer Perceptron	√	√	√		
Nearest Neighbour					
Radial Basic Functions		√			
Bays				√	
K-Means	√	√	√		
Association Rules	√	√	√		
Kohnen	√				

Both the comparisons reveal that most of the data mining frameworks prefer other mining methods compared to Association Rule Mining. We tried to focus on the Association Rule Algorithms for building the data mining framework.

III. PROPOSED WORK

Here, we present the DDM framework for mining large distributed databases . It uses a server to perform the data mining processes using the clients' inputs.

Association Rule Mining

In data mining, association rule learning is a popular and well researched method for discovering interesting relations between variables in large databases. Based on the concept of strong rules, Agrawal et al. introduced association rules for discovering regularities between products in large scale transaction data recorded by point-

of-sale (POS) systems in supermarkets. Such information can be used as the basis for decisions about marketing activities such as, e.g., promotional pricing, product placements, Web usage mining, intrusion detection and bioinformatics etc.

Definition

Following the original definition of the problem in association rule mining is defined as: Let I={i1,i2,..,in} be a set of *n* binary attributes called *items*. Let D={t1,t2,…,tm} be a set of transactions called the *database*. Each transaction in *D* has a unique transaction ID and contains a subset of the items in *I*.

A *rule* is defined as an implication of the form X=>Y where

$$X, Y \subseteq I \text{ and } X \cap Y = \emptyset.$$

The sets of items (for short *itemsets*) *X* and *Y* are called *antecedent* (left-hand-side or LHS) and *consequent* (right-hand-side or RHS) of the rule respectively.

Support and Confidence

To select interesting rules from the set of all possible rules, constraints on various measures of significance and interest can be used. The best-known constraints are minimum thresholds on support and confidence.

The *support* supp(*X*) of an itemset *X* is defined as the proportion of transactions in the data set which contain the itemset. In the example database, the itemset {milk, bread, butter} has a support of 1 / 5 = 0.2 since it occurs in 20% of all transactions (1 out of 5 transactions).

The confidence of a rule is defined as conf(X=>Y)=supp (X U Y). For example, the rule {milk, bread}=>{butter} has a confidence of 0.2 / 0.4 = 0.5 in the database, which means that for 50% of the transactions containing milk and bread the rule is correct. Confidence can be interpreted as an estimate of the probability $P(Y \mid X)$, the probability of finding the RHS of the rule in transactions under the condition that these transactions also contain the LHS.

The *lift* of a rule is defined as

lift(X=>Y)= supp(X U Y)/ (supp(Y) * supp(X)) or the ratio of the observed support to that expected if X and Y were independent.

The rule {milk, bread} => {butter} has a lift of 0.2/(0.4*0.4)=1.25. The *conviction* of a rule is defined as conv(X=>Y)=(1-supp(Y))/(1-conf(X=>Y)).

The rule {milk, bread} => {butter} has a conviction of (1-0.4)/(1-0.5) = 1.2 and can be interpreted as the ratio of the expected frequency that X occurs without Y (that is to say, the frequency that the rule makes an incorrect prediction) if X and Y were independent divided by the observed frequency of incorrect predictions. In this example, the conviction value of 1.2 shows that the rule {milk, bread}=>{butter} would be incorrect 20% more often (1.2 times as often) if the association between X and Y was purely random chance.

The association rule extraction process is divided into the following two phases.

1) Discover the *frequent item sets*, i.e., the sets of items with a *support* greater than a given threshold.

2) Build the association rules using the previously obtained frequent item sets [12].

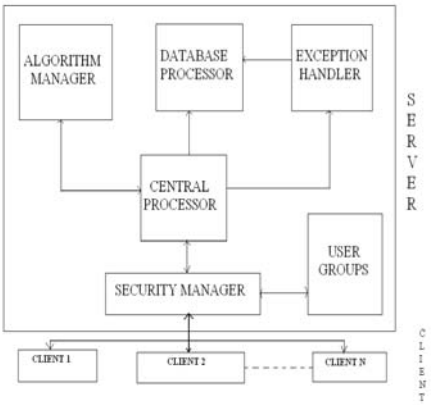
Most of the association rule mining algorithms follow the support-confidence framework, which first:

i) searches for frequent itemsets using the support threshold and then

ii) derives the rules from the set of frequent itemsets using the confidence threshold. While the second step is straightforward, the first step is very computationally expensive due to the huge search space of possible itemsets and the support counting operation that involves the processing of all transactions in the dataset [13].

A. Proposed system Design

Fig. 1. Proposed Data mining System Design



Server and Client

The server was implemented in Java for portability and because Java is a nice object oriented language with many standard libraries. Most of the server code was designed and written with reusability. This was actually a necessity for some classes since they would need to be used by people who want to add new functionality to the server by writing new mining algorithms or by creating and adding new databases.

The DDM server implementation is based on sockets. The server works in the traditional way, i. e. it accepts connections from clients and creates for each connection a new thread to handle the connection. Each thread will wait for requests from the corresponding client and when a request is received the thread will execute the request and send its result back to the client. When the client disconnects, the thread will terminate its execution.

The server performs all mining operations and even some operations like the sorting of the association rules such that a client application could be written with a minimal effort since it would just need to display the results sent by the server. For Association Rule Mining several algorithms are used. Apriori, AprioriTID, Apriori Hybrid, SETM, DIC etc.

The Client consists of a well defined user interface with user-friendliness.

B. Implementation of the Proposed system Design

Many components are created for implementing the proposed DDM frame work using association rule mining.

DBConfig

This component provides the data structures to store all the management information and also the methods for changing and retrieving this information. In order to make easier the user management, the server provides a system of groups and permissions. Since the DBConfig object contains all the state of the server regarding its users, groups, algorithms, and databases, it has to be serialized whenever its state is changed in order to not loose any information in the unfortunate event of a power failure. The DBConfig can also be serialized through the socket connection in order to provide a client application with a snapshot of the current status of the server. On the client side the application can use the methods provided by DBConfig in order to read its status.

Database Reader and Writer

The DBReader and DBWriter classes provide the functionality necessary to read and write datasets. Each row of the dataset is represented as a list of integers, the first integer giving the number of items in the transaction, and the next integers representing the items of the transaction. Error detection is provided by computing a CRC code for the contents of the dataset. The names of the columns and a description of the dataset are stored in the header of the dataset. The methods for reading or writing rows to the dataset return and respectively take as argument Itemset objects.

Algorithm manager

The algorithm manager deals with running algorithms over databases. It also has to deal with the loading of algorithms from disk and their execution since new algorithms can be added dynamically to the server. The strategy for loading algorithms is pretty simple, the algorithm manager first tries to instantiate an algorithm object normally, if this fails the algorithm manager will try to open a jar file with the same name as the algorithm, read all classes contained therein, and load the algorithm class. If the last step fails, that means that either the jar files did not exist or that the jar file was corrupted or that the algorithm was not implementing the interface

LargeItemsetsFinder.

The algorithm manager also takes care of the performance evaluation process.

AssociationsFinder interface and algorithms

The AssociationsFinder interface is the interface to be implemented by the algorithms that generate association rules. This interface is provided in the eventuality that someday we will want to be able to add other algorithms for this part of the mining process. . The interface provides a single method which receives as parameter a DBCacheReader that allows the algorithm to read the frequent itemsets from a cache. The results (the association rules found) are returned by the algorithm in a Vector. We have implemented the Apriori algorithm for generating association rules in file AprioriRules.java. In order to determine the large itemsets, we read the itemsets from the cache and we place them in a prefix tree (also called SET: Set Enumeration Tree, implemented in class SET.java) which allows us to find with reasonable performance the large itemsets and also to retrieve the support of each frequent itemset.

Synthetic data generator

The synthetic data generator is a reusable component that takes several parameters as Input and then generates a synthetic dataset, row by row.

ServerChild

The ServerChild implements a thread of the Server. It is fully responsible for the communication with a client application. The ServerChild stays in a loop waiting for client requests and when one is received it will process it and then send back the results. In case of a fatal error the ServerChild will close the socket connection and terminate its execution. This will not affect however the execution of the server itself.

Server

This is the server's main class, which opens a socket on which it listens for client requests for a connection and for each such request it creates a new thread running an instance of the ServerChild class.

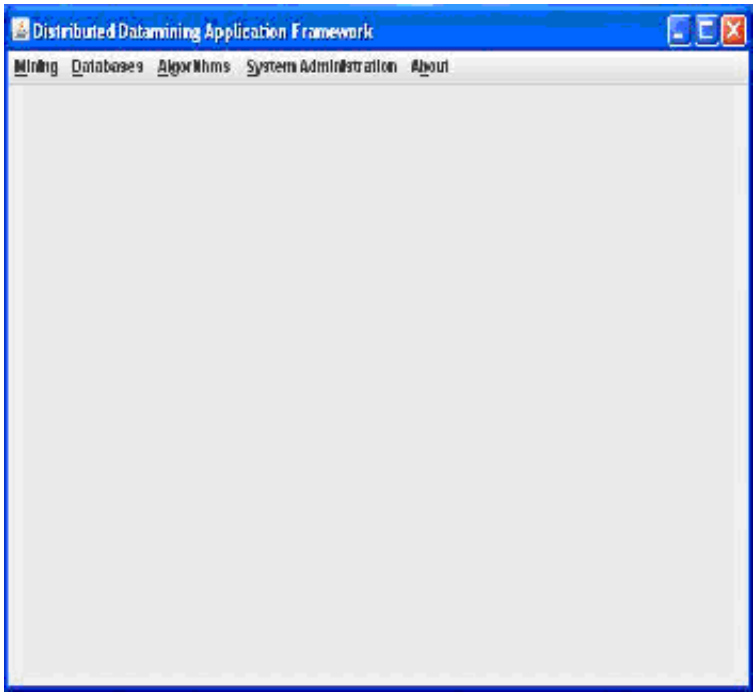
Server communication protocol

The general strategy is that the client sends a request in the form of a String containing a special identifier of the request, followed by a Vector containing additional different Objects, representing the parameters of the request. The server performs the requested action and, if successful, it sends back a message consisting in a String "OK", followed by a Vector containing different Objects representing the results that must be communicated to the Client. Or the server can send back a String "ERROR", followed by a Vector containing as its first element a String explaining the cause of the failure. The Server might send back a String "WARNING", followed by a Vector containing a String describing the reason for the warning.

IV. EXPERIMENTAL RESULTS

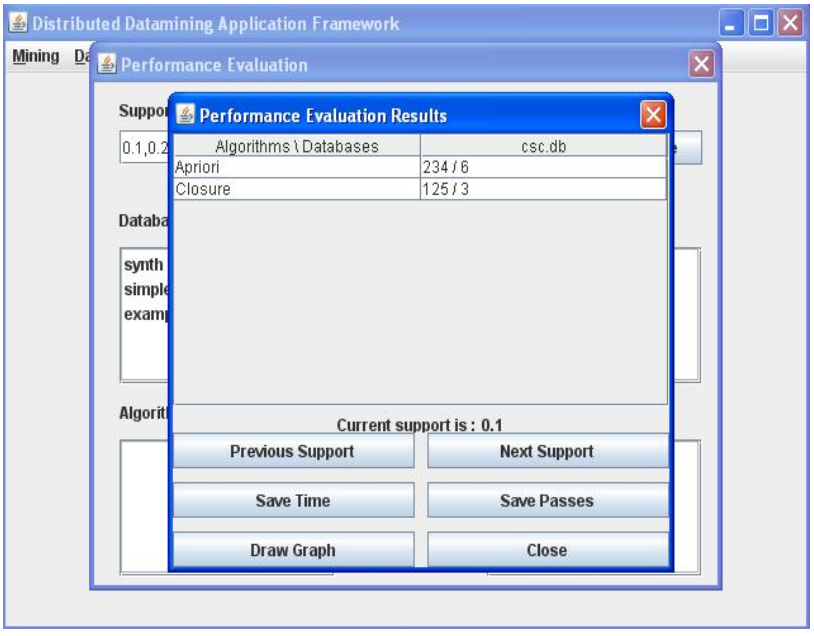
In this section we present the performance of the proposed ddm framework model on different data bases and different algorithms. The mining model has been implemented in Java, Pentium IV processor has been used with a speed of 1.86 GHZ and the paradox (.db) format is used for the database setting.

Fig. 2. Distributed Data Mining Framework



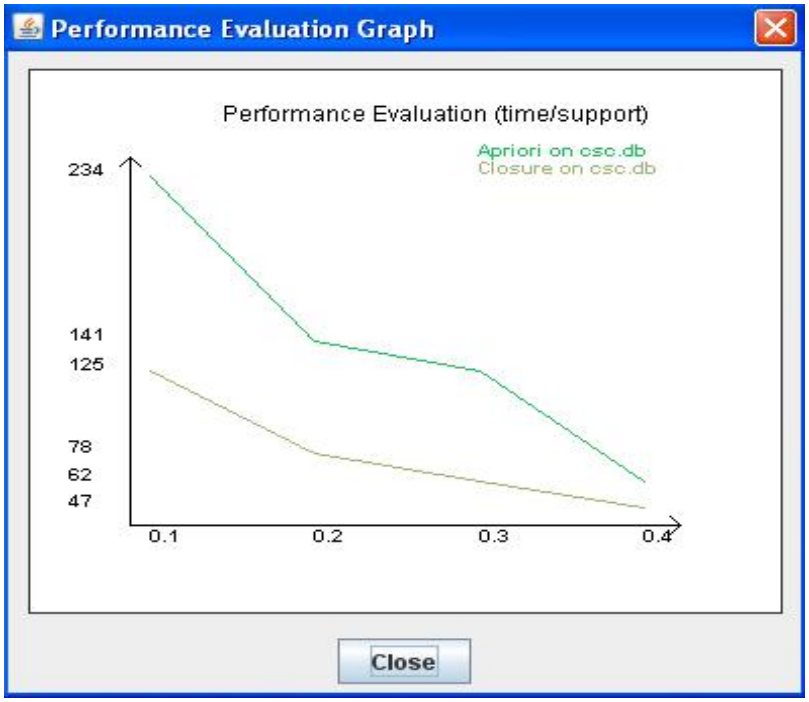
The input value for mininmum support and confidence should be the digital value between 0 and 1. When we click mine the system will do the client side input value checking. The Refresh button is to refresh the contents because multiple users may use the system at the same time. Database and algorithm resource can be updated at any time.

Fig. 3. Performance Evaluation of Algorithms and Databases



The algorithms for association rule mining are added and the performance evaluation is done using this proposed ddm framework. Here we have taken the Apriori and Closure algorithm for evaluation. Several Hybrid algorithms can also be added for more effectiveness.

Fig. 4. Performance Evaluation of Apriori and Closure Algorithms



V. CONCLUSION AND FUTURE WORK

Data mining has a large family composed of various algorithms, and the scope is still expanding, because researchers devote to improve the efficiency and accuracy of existed algorithms, new approaches increase with time. The more complex the application is, the larger the gap comes into existence between application and users. We study data mining related applications to draw the concepts and characters, and then propose a selection model to match these business requirements to data mining categories to connect complex data mining concepts like association rule mining with business problems and assists users to choose the best data mining solution.

The number of users, groups, databases, and algorithms existing in the system will affect the performance of the administrative operations since we use simple structures to manage this information and their access and modification performance is usually linear. So, in future we will concentrate on the performance factors while working with large number of algorithms, databases and clients. Decision making would be easier by increasing the modes graphical representations such as bar graphs, pie charts etc. By adding additional data mining models like clustering, classification etc., can be converted into a full-fledged data mining framework for mining real large databases. And we are now concentrating to apply this distributed data mining framework for spatial data mining.

REFERENCES

[1] Dr (Mrs).Sujni Paul,"AN OPTIMIZED DISTRIBUTED ASSOCIATION RULE MINING ALGORITHM IN PARALLEL AND DISTRIBUTED DATA MINING WITH XML DATA FOR IMPROVED RESPONSE TIME", International Journal of Computer Science and Information Technology, Volume 2, Number 2, April 2010.

[2] Jiangping Chen, Wuhan Hubei, "AN ALGORITHM ABOUT ASSOCIATION RULE MINING BASED ON SPATIAL AUTOCORRELATION", The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences. Vol.XXXVII. Part B6b. Beijing 2008.

[3] Josenildo C. da Silva2, Chris Giannella1, Ruchita Bhargava3, Hillol Kargupta1;4, and Matthias Klusch, "2Distributed Data Mining and Agents", IEEE conference,2005.

[4] Jeawei Han, Micheline Kamber, "Data mining concepts and techniques", Second Edition.

[5] S. Nirmal Chander, P. Ram Prasath, V. Santhosh Kumar, "Enhancing the Relevance of Semantic Web information Retrieval Results using Extention Theory", TISC, 2010.

[6] Vuda Sreenivasa Rao, Dr. S Vidyavathi, "DISTRIBUTED DATA MINING AND MINING MULTI-AGENT DATA", (IJCSE) International Journal on Computer Science and Engineering, Vol. 02, No. 04, 2010, 1237-1244

[7] S.VEERAMALAI, A.KANNAN, "An Intelligent Association Rule Mining Model for Multidimensional Data Representation and Modeling", International Journal of Engineering Science and Technology Vol. 2(9), 2010, 4388-4395

[8] Vuda Sreenivasa Rao, S Vidyavathi, G.Ramaswamy, " DISTRIBUTED DATA MINING AND AGENT MINING INTERACTION AND INTEGRATION: A NOVEL APPROACH", IJRRAS 4 (4) , September 2010

[9] Vuda Sreenivasarao, Rallabandi Srinivasu, Prof. G.Ramaswamy, Nagamalleswara Rao Dasari, Dr. S Vidyavathi, "The Research of Distributed Data Mining Knowledge Discovery Based on Extension Sets", International Journal of Computer Applications (0975 – 8887), Volume 8– No.2, October 2010

[10] Yoones Asgharzadeh Sekhavat, Mohammad Fathian, Mohammad Reza Gholamian, Somayeh Alizadeh, " Mining important association rules based on the RFMD technique", Int. J. Data Analysis Techniques and Strategies, Vol. 2, No. 1, 2010.

[11] Longbing Cao, Senior Member, IEEE, Yanchang Zhao, Member, IEEE, Huaifeng Zhang, Member, IEEE, Dan Luo, Chengqi Zhang, Senior Member, IEEE, and E.K. Park," Flexible Frameworks for Actionable Knowledge Discovery", IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 22, NO. 9, SEPTEMBER 2010

[12] Nicolas Marin, Carlos Molina, José M. Serrano, and M. Amparo Vila, "A Complexity Guided Algorithm for Association Rule Extraction on Fuzzy DataCubes", IEEE TRANSACTIONS ON FUZZY SYSTEMS, VOL. 16, NO. 3, JUNE 2008

[13] Tien Dung Do, Siu Cheung Hui, A. C. M. Fong, Senior Member, IEEE, and Bernard Fong, Senior Member, IEEE "Associative Classification With Artificial Immune System", IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION, VOL. 13, NO. 2, APRIL 2009

[14] Dr (Mrs).Sujni Paul, "AN OPTIMIZED DISTRIBUTED ASSOCIATION RULE MINING ALGORITHM IN PARALLEL AND DISTRIBUTED DATA MINING WITH XML DATA FOR IMPROVED RESPONSE TIME", International Journal of Computer Science and Information Technology, Volume 2, Number 2, April 2010

[15] G.J. Pottie and W.J. Kaiser, "Wireless Integrated Network Sensors," Comm. ACM, vol. 43, no. 5, 2000, pp. 51–58

[16] Anup Kumar and Mehmed Kantardzic University of Louisville Samuel Madden Massachusetts Institute of Technology "Distributed Data Mining Framework and Implementations", IEEE INTERNET COMPUTING Published by the IEEE Computer Society JULY • AUGUST 2006

[17] S.N.Sivanandam, S.Sumathi, " Data Ming: concepts, tasks and techniques", Thomas Business Information India Pvt.Ltd, ISBN: 81-315-0009-8, India 2006

[18] Muhaimenul Adnan and Reda Alhajj, "A Bounded and Adaptive Memory-Based Approach to Mine Frequent Patterns From Very Large Databases", IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART B: CYBERNETICS, VOL. 41, NO. 1, FEBRUARY 2011

[19] Cristobal Romeo, Sebastian Ventura, " Educational Data Mining: A Review of the state of the Art", IEEE Transactions on systems, man and cybernetics- Part C: Applications and Reviews, Vol 40, No.6, Nov 2010

[20] Dean W. Abbott, I. Philip Matkovsky, John F. Elder IV, Ph.D., "An Evaluation of High-end Data Mining Tools for Fraud Detection" A research for supported by the Defense Finance Accounting Service Contract N00 244-96-D-8055 under the direction of Lt. Cdr. Todd Friedlander, in a project initiated by Col. E. Hutchison.

[21] John F. Elder IV & Dean W. Abbott, "A Comparison of Leading Data Mining Tools", Fourth International Conference on Knowledge Discovery & Data Mining, 1998, New York

AUTHORS PROFILE

P.T.KAVITHA

Pursuing her PhD in Sathyabama University, Chennai,India. She is working as a Lecturer in SRR Engineering College, Chennai. She has presented papers in distributed data mining in national and International Conferences.

Dr.T.SASIPRABA

Working as the Dean of Sathyabama University, She has published several papers in International and National Journals. Presently guiding number of PhD Scholars. She has obtained her Doctorate from Sathyabama University.