A compendium on Data Mining Algorithms and Future Comprehensive

Manalisha Hazarika, Mirzanur Rahman

Abstract*—Data mining is a powerful and new method of analyzing data and finding out new patterns from vast volume data .There is an enormous amount of data stored in databases and data warehouse due to enormous technological advancements in computing and Internet. In recent days multinational companies and large organizations have operations in many places in the world. Each place of operation may generate bulk volumes of data. Corporate decision makers require access from all such sources and take strategic decisions. The information and communication technology have highly used in the industry. One of the main challenges in database mining is developing fast and efficient algorithms that can handle large volumes of data as most of the mining algorithms perform computation over the entire databases ,often very large. Today's business environment, efficiency or speed is not the only key for competitiveness. Such tremendous amount of data, in the order of tera- to peta-bytes, has fundamentally changed science and engineering, transforming many disciplines from data-poor to increasingly data-rich, and calling for new, data-intensive methods to conduct research in science and engineering. This paper gives an outline of the existing data mining algorithms and give the future space of some algorithm.

Index Terms*—Data Mining, frequent item, load balance, parallel algorithm.

I. INTRODUCTION

The advent of information technology in various fields of human life has lead to the large volumes of data storage in various formats like records, documents, images, sound recordings, videos, scientific data, and many new data formats. The data collected from different applications require proper mechanism of extracting knowledge/information from large repositories for better decision making. Knowledge discovery in databases (KDD), often called data mining, aims at the discovery of useful information from large collections of data[1]. The literature available for data mining contains many definitions [2][3][4][5]. Some of them depend on the application and how data has been organized into a database whereas some of them depend on the discovery of new information from the facts in a database. Data mining is a process by which one can extract interesting and valuable information from large data using efficient techniques. Data Clustering, Data Classification, Detection of Outliers and Association Rule Mining are useful basic data mining a techniques depending upon the type of information sought from databases. Information and knowledge for managerial decision making in business.

Manuscript received July, 2013

Manalisha Hazarika, Information Technology, Gauhati University, Guwahati,India.

Mirzanur Rahman,,Information Technology, Gauhati University, Guwahati,India.

To generate information it requires massive collection of data.The data can be simple numerical figures and text documents, to more complex information such as spatial data, multimedia data, and hypertext documents. To take complete advantage of data;the data retrieval is simply not enough, it requires a tool for automatic summarization of data, extraction of the essence of information stored, and the discovery of patterns in raw data. With the enormous amount of data stored in file databse,it is increasingly important to develop powerful tool for analysis and interpretation of such data and for the extraction of interesting knowledge that could help in decission making. The only answer to all above is datamining.

The objective of this paper is to provide various data mining algorithms with reference association rule.Association rule mining is the most important technique in the field of data mining. It aims at extracting interesting correlation, frequent pattern, association or casual structure among set of item in the transaction database or other data repositories. Association rule mining is used in various areas for example Banking, department stores etc. This paper surveys the most recent existing association rule mining techniques using different algorithms and discuss future prospective. Only Association Rule Mining (ARM) is the major technique of data mining that finds correlations among items in a given data sets and establishes an association between two non overlapping sets of frequently occurring values in a database. It helps retailers in planning marketing strategies, catalog design and store layout by finding the association between the different items purchased by the customer. e.g. If retailer keeps bread with butter then the chances of sale will be increased because customer who buys bread is also interested in butter. Sequence pattern mining [6], plays an important role in analysis of shopping sequences, discovery of DNA sequences pattern, analysis of network access mode, and so on. Parallel mining algorithms have high-performance in massive data mining. In recent days Researchers have sought cost-effective improvements by building "parallel" computers-computers that perform multiple operations in a single step. The parallel algorithms guide these parallel computers to carry out this task. The scalable performance and lower cost of parallel platforms is reflected in the wide variety of applications. In this talk, we will present various applications of parallel algorithms, challenges associated in designing them.

The configuration of this paper is as follows. Section 2 describes literature review of ARM. Section 3, discusses various sequential ARM algorithms. Sections 4 discuss the challenges in exploiting parallelism to the ARM algorithms , Section 5 discuss proposed work and at last conclude.

II. ASSOCIATION RULE MINING BASIC CONCEPT

Given a set of transactions, where each literal (called items), association rules is an expression of the form X Y, where X and Y are set of items[1,3,6,8]. There are two important measures for association rules, support (σ) and confidence (c), can be defined as follows: Support is the ratio (in percent) of the records that contain X =>Y to the total number of records in the database. Confidence is the ratio (in percent) of the number of records that contain X =>Y to the number of records that contain X.

For example, let us consider customer's purchase data shown in Table 1

Table 1: Consumer purchased Data

Trans ID (TID)	Item
1	egg, bread, milk
2	egg, milk
3	egg
4	egg, bread, milk
5	cheese

For example, if you were a owner of the supermarket, you would like to think of the layout of the store. In that case, the rules in Table 2 can be useful.

Table 2: Association Rules

Rule	Support	Confidence
egg, bread => milk	40%	100%
egg => bread	40%	50%

The problem of mining association rules can be decomposed into two sub problems:

1. Find all sets of items(itemset (IS)) whose support is greater than the user-specified minimum support(MS):large item set or frequent Item sets (FIS). For example, if MS is 40% then {egg, bread}(40%),{egg, bread, milk}(40%).
2. Use the frequent item set to generate the desired rules If ABCD, AB is FISs, then AB=>CD conf = ***support(ABCD)/support(AB***) and if conf>=minimum confidence(MC), then the rule holds For example, if MC is 75%, then {egg, bread} {milk}(100%) holds.

Notation used in this paper is summarized below k-item set---An item set having k items L_k --- Set of frequent k-item sets(those with minimum support). Each member of this set has two fields : i) item set and ii) support count.C_k Set of candidate k-item sets (potentially frequentcitem sets). Each member of this set has two fields : i) itemcset and ii) support count.

3. Algorithms for discovering large itemsets make multiple passes over the data. In the first pass, we count the support of individual items and determine which of them are large ,i.e. have minimum support. In each subsequent pass, we start with a seed set of itemsets found to be large in the previous pass. We use this seed set for generating new potentially large itemsets, called candidate itemsets, and count the actual support for these candidate itemsets during the pass over the data. At the end of the pass, we determine which of the candidate itemsets are actually large, and they become the seed for the next pass. This process continues until no new large itemsets are found. performance.

Parallel algorithm was necessary & implementable because of substantial improvements in multiprocessing systems & rise of multi-core processors. Performance of a computer is decided by Time required to perform basic operation (limited by clock cycles) & No. of these basic operations that can be performed concurrently. Eg: splitting up the job of checking all of the numbers from one to a hundred thousand to see which are primes could be done by assigning a subset of the numbers to each available processor, and then putting the list of positive results back together.While exploiting parallelism to sequential ARM algorithms the challenges occurs are efficient utilization of memory, minimization of communication among processors, load balancing among processors, efficient data and task decomposition and proper synchronization etc. which cannot be overlooked.

III. LITERATURE REVIEW OF PARALLEL ARM ALGORITHMS BASED ON SEQUENTIAL ARM ALGORITHMS

The sequential ARM algorithms developed are designed to find frequent itemsets and generate association rules. Here, we discuss few of them. These algorithms are classified on the basis of data layout storage used as shown in the following table.

Agrawal proposed GSP [7] which presented time constraints, sliding time window and user-defined taxonomies to decrease the number of sequences and to reduce the overhead. Shintani proposed three parallel tactics based on GSP:NPSPM,SPSPM and HPSPM [8], as the hash mechanism was used in HPSPM, it has the best performance, and is better than the first two algorithms. But they all need to scan the database for many times and to exchange remote database partition which result in greater communication overhead and I/O costs.

In order to address the above problems, Zaki presented pSPADE [9], which was based on a serial algorithm SPADE and a shared memory parallel structure, lattice theory was used to minimize I/O costs, due to the limited bandwidth of the shared memory parallel structure, the scalability may be inhibited at some point.

Wang in [10] made a comprehensive survey on parallel frequent pattern mining technology, he pointed out the efficiency, scalability of parallel technique in massive datamining and the transformation of the parallel mining platform from distributed systems to multi-core system;

Wu proposed EDMA [11] algorithm to mine association rules, it minimized the number of candidate sets, exchanged messages by local and global pruning and reduced the scan time by decreasing the size of average transactions and datasets.

The PartSpan algorithm was proposed in [12],data parallel and task parallel were used to divide and distribute the projection database, but it was lack of the necessary load-balancing mechanism.

So Zhou makes an improvement on the load imba-lance in parallel sequence mining algorithm DPA [13], a new approach was proposed to generate a balanced workload among processors and to reduce processor idle time.

On the basis of the tree projection, Valerie presented a new parallel algorithm based on distributed storage: STPF [14],it has good scalability by using breadth-first approach in the static load balancing mechanism.

Han proposed Par-CSP [15] algorithm for parallel sequence pattern mining,

dynamic load balancing and divide and conquer strategy were introduced to minimize the costs
and to obtain better speedup.
At the basic of Par-CSP, Niagara proposed an improved Par-ClosP [16] algorithm to solve the problem of parallel closed sequence mining. It introduced a new pruning method and pseudo projection technique to minimize the use of time and space.
FPM, IDD, CD, PEAR, PDM, DD, SPA, CandiDistri., HPA, PPAR, P-Cluster, MAFIA, SPRINT, SLIQ/R, SLIQ/D, ScalPrac implemented on dedicated homogeneous system and uses static load balancing technique based on the initial data decomposition for load assignment to the processors in the system. As a typical parallel database server has multiple users, and has transient loads, there is need of dynamic load balancing schemes.

Table 3: Comparisons of some Parallel algorithms

aAlgorithms	Data layout	Some Results
GSP [7]	Horizontal	Decrease the number of sequences and to reduce the overhead.
parallel tactics based on GSP: NPSPM 、SPSPM and HPSPM [8],	Horizontal	In greater communication overhead and I/O costs.
pSPADE [9]	Hrizontal	Shared memory parallel structure, lattice theory was used to minimize I/O costs.
EDMA [11]	Horizontal	Reduced the scan time
CCPD, PC, APM	Shared memory system	Homogeneous system,uses static load balancing
FPM, IDD, CD, PEAR, PDM,DD, SPA, Candi. Distri., HPA,PPAR, P-Cluster, MAFIA, SPRINT, SLIQ/R, SLIQ/D, ScalPrac	Distributed memory system	Homogeneous system,uses static load balancing
Par-Eclat, Par-Clique, Par-MaxEclat,Par-Max Clique	Hierarchical system	Homogeneous system,uses static load balancing
PartSpan	Projection database	Lack of the necessary load-balancing mechanism.
DPA		Balanced workload among processors and to reduce processor idle time.
STPF	Distributed storage	Stataic load balancing
Par-CSP	Closed sequence mining	Minimize the costs and to obtain better speedup. Dynamic load balancing

Dynamic load balancing is also crucial in a heterogeneous environment, which can be composed of meta-and super-clusters, with machines ranging from ordinary workstations to supercomputers A dynamic algorithm for heterogeneous system where there is no prior knowledge of the processors is in great demand. Kun-Ming Yu, Jiayi Zhou and Wei Chen Hsiao proposed a parallel and *distributed mining algorithm based on FP-tree structure, Load Balancing FP-Tree (LFP-tree) [17] that divides the itemset* for mining by evaluating the tree's width and depth and proposed a simple and trusty calculate formulation for loading degree. The experimental results show that LFP-tree can reduce the computation time and has less idle time compared with Parallel FP-Tree (PFP-tree) and has better speed-up ratio section that you want to designate with a than PFP-tree when number of processors grow [17]. But it has drawback of maintaining complex tree structure.

IV. CHALLENGES IN EXPLOITING PARALLELISM AND FACTORS AFFECTING ESTIMATION OF THE COMPLEXITY /COST OF PARALLEL ALGORITHMS

Since the sequential algorithms developed so far have many limitations and are not suitable for massive data sets, there is a great need of parallel algorithms for achieving high performance. To design such algorithms that handles massive data sets with large dimensions on different platforms with different configurations there are many challenges that need to be considered .

Table 4: Classification of parallel Algorithms

Algorithms	Method or data lay out used	Comparison with sequential ARMs
Apriori based algorithms	-Count Distribution -Data method follows a data-parallel strategy and statically partitions the database into horizontal partitions that are independently scanned for the local counts of all candidate itemsets on each process.	- Count Distribution method exhibits better performance and scalability than the other two methods. - The steps for the Count Distribution method aregeneralized for distributed memory multiprocessors. - Count Distribution algorithm,communicati on is minimized since only the counts are exchanged among the processes in each iteration
	-Data Distribution -The Data Distribution method attempts to utilize the aggregate main memory of parallel machines by partitioning both the database and the candidate itemsets.	
	-Count Distribution-The Candidate Distribution method also partitions candidate itemsets but selectively replicates instead of partition-and-exchangin g the database transactions, so that each process can proceed independently	

Vertical Mining	There are four variations of parallel Eclat - ParEclat, ParMaxEclat, ParClique, and Par-Maxclique	-The Eclat-based algorithms scan the database only three times and significantly reduces the disk I/O cost.
	- MaxClique - All of them are similar in parallelization and only differ in the itemset clustering techniques and itemset lattice traversing strategies. - ParEclat and ParMaxEclat use prefix-based classes to cluster itemsets, and adopt bottom-up and hybrid search strategies respectively to traverse the itemset lattice. -ParClique and ParMaxClique use smaller clique-based itemset clusters, with bottom-up and hybrid lattice search, respectively.	
Pattern-Growth Method	A partitioning-based, divide and-conquer strategy is used to decompose the mining task into a set of smaller subtasks for mining confined patterns in the so-called conditional pattern bases. The conditional pattern base for each item is simply a small database of counted patterns that co-occur with the item. That small database is transformed into a conditional FP-tree that can be processed recursively.	-In parallel FP-growth, since all the transaction infor-mation is compacted in the FP-trees, no more data-base scan is needed once the trees are built. So the disk I/O is minimized by scanning the original database only twice. -The major comm-unication/synchronization overhead can minimized.

Different parallel ARM algorithms have been designed that considers one or more challenges based on communication, synchronization, load balancing some have focused on load balancing or memory utilization while other considered challenges like task decomposition etc.

1. How to obtain effective load balancing? Which type of load balancing static or dynamic to be used?
2. How to utilize memory efficiently? How to develop algorithm for different memory system?
3. Which data layout to be used (horizontal, vertical or hybrid)?
4. How to do efficient task decomposition? i.e. To develop algorithm that uses data or task parallelism.
5. How to minimize communication between processors?
6. How to minimize synchronization?

Different parallel ARM algorithms have been designed that considers one or more challenges .some have focused on load balancing or memory utilization while other considered challenges like task decomposition etc. Let us take above mentioned challenges one by one and discuss which of them are considered and how much covered by which parallel ARM algorithms.

Factors Affecting Estimation Of The Complexity/Cost Of Parallel Algorithms:

-Time

-memory(space)

-communication between different processors

This communication is achieved by message passing & shared memory.

A. *Load Balancing:*

For obtaining good load balancing the task should be mapped to all the process equally so that the execution time for all the process remains the same and none of them sits idle and wait for the task to be allocated to it. To achieve this following point has to be considered [18]:-

[1] The interaction time between processes should be minimized.

[2] The time for which processes sits idle and waiting for another task to be allocated to it should be minimized.

Achieving these two at the same time is quite difficult because minimization of interaction can be done only if all the related tasks are assigned to the same processes that lead to high workload imbalance. A task dependency graph can be used to determine which process is busy in execution and which waits for the others to complete its execution so that it will start its execution. The two mapping techniques that can be used by parallel algorithms are static and dynamic.

A. Static Vs. Dynamic Load Balancing

Static load balancing initially partitions work among the processors using a heuristic cost function; no subsequent data or computation movement is available to correct load imbalances resulting from algorithms' dynamic nature [20].In case of dynamic load balancing the check on the load of every processor in the system after a definite interval is done during execution and also to maintain equal load balance the data is moved from heavily loaded to least loaded processor. It costs for work and data movement. However, dynamic load balancing is advisable in the case where there is

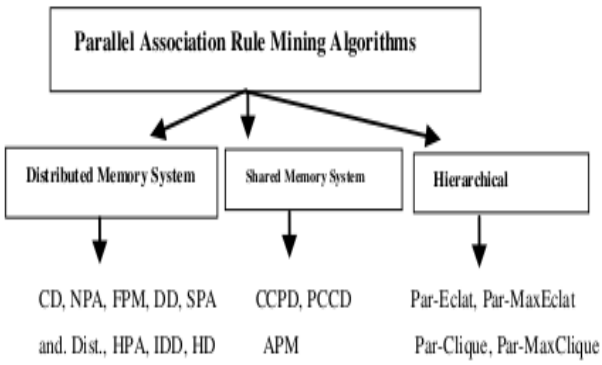

Figure 1: List of Parallel association rule mining algorithm developed so far for homogeneous system that uses static load balancing technique on different machines i.e. shared memory, distributed and hierarchical memory system [19]

a large load imbalance that changes with time adds extra computation for the processors that involved in data movement. So, the challenge here is to reduce the additional

B. Distributed Vs. Shared Memory System

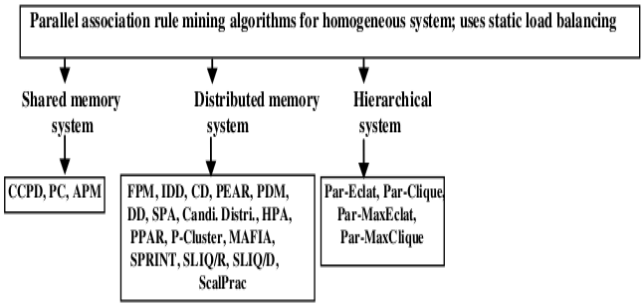

Figure 2: Categorization of Parallel association rule mining algorithm according to the shared, distributed & hierarchical

In distributed memory each processor has a private memory whereas in shared memory all processors access common memory. Parallel programs are easy to implement on such a system but a common bus's finite bandwidth can limit systems [19], the scalability which is eliminated in a distributed-memory, by having message-passing architecture. But it involves complex programming. Since synchronization is implicit in message passing, so the challenge in distributed memory system is communication optimization. The main challenge for obtaining good performance on distributed memory system is finding a good data decomposition among the nodes and minimizing communication [19] where as in shared memory system, is to achieve good data locality.

To overcome these constraints of shared memory system and distributed memory system a hybrid of both can be used but the challenge is how to achieve that. So, we conclude that a parallel ARM algorithm is required to be designed and implemented for hybrid memory system for better memory utilization.

C. Data Vs. Task Parallelism

Task and data parallelism plays important role for exploiting algorithm parallelism. In case of data parallelism the database is partitioned among P processors-logically partitioned for Shared Memory systems, physically for Distributed Memory systems whereas in Task parallelism the processors perform different computations independently, such as counting a disjoint set of candidates, but have or need access to the entire database [19]. e.g. Count distribution, Non-Partitioned Apriori (NPA), Fast Parallel Mining algorithms are build on Distributed Memory system and uses data partitioning. Data Distribution, Candidate Distribution, Simply-Partitioned Apriori, Hash-Partitioned Apriori, Intelligent Data Distribution and Hybrid Distribution algorithms are built on Distributed Memory systems but uses task partitioning [19].

D. Decomposition Techniques

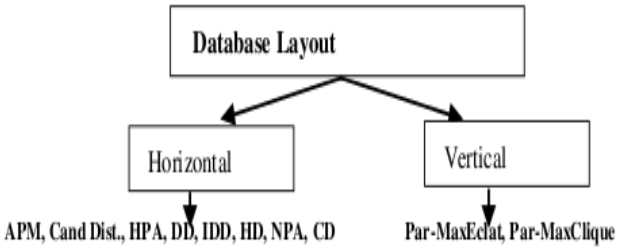

Figure 3: Categorization of parallel ARM algorithms on the basis of data layout [19]

Another challenge involves the selection of which decomposition technique i.e. recursive, data, exploratory and speculative that lead to the best parallel algorithm for a given problem that provides the concurrency. Most ARM algorithms assume a horizontal data layout where each transaction along with the attribute values for that transaction is stored as a unit in a row. In vertical data layout, each attribute is associated with a list of all transactions containing the item and the corresponding attribute value in that transaction.

V. CONCLUSION AND FUTURE WORK

In this way we have studied the versatile use of parallel algorithms in finding frequent items. The parallel algorithms can solve the problems in various domains and give us efficient throughput. Lots of parallel ARM algorithms are developed so far but some are strong in some points while others are strong in some others Also most of the algorithms are design-ed for homogeneous system with static load balancing which is far from reality. Algorithm for heterogeneous system with dynamic load balancing is required to develop with high per-formance. So, there is a great need of the algorithm with the minimum constraints discussed above.

Now a days parallelism is become very popular in\ass-ociation rule mining algorithms for big data in petabytes era for good computation, fast algorithm and some efficiency. So we are decided to apply this concept on a association rule mining algorithm. We have taken the help of all the above studied algorithms and proposed one algorithm which will try find out the discovery of frequent itemset using eclat algorithm in less time complexity, less synchronization, less load balance and high throughput. In this paper, we discuss large-scale datamining approaches large-scale data set. Based on our observations, we proposed an idea of utilizing. In the future, we plan to implement and explore our new framework by integrating with different data mining toolkits and evaluating with different data sets, and then compare it with existing large scale data mining approaches.

REFERENCES

1. Heikki, Mannila. 1996. Data mining: machine learning, statistics, and databases, IEEE
2. R. Agrawal, T. Imienski and A. Swamy, Database Mining : A Performance Perspective, IEEE Tran. On Knowledge and Data Engg., December,1991.

3. [3]M-S Chen, J Han and P. S. Yu, Data Mining : An Overview from a Database Perspective, IEEE Tran. On Knowledge and Data Engg., December, 1996.
4. A.Y. Zomya, T.E. Ghazawi and O. Frieder, Parallel and Distributed Computing for Data Mining, IEEE Concurrency, Oct./Nov. 1999
5. Jong Soo Park, Ming-Syan Chenand Philip S. Yu. An effective hash-based algorithm for mining association rules. In Proceedings of 1995
6. R. Agrawal, and R. Srikant, "Mining sequence patterns," proceedings of the 11th International Conference on Data Engineering. Taipei, 1995, pp3-14.
7. R. Agrawal , and R. Srikant, "Mining sequence patterns: Generalizations and Performance improvements," proceedings of the 11th International Conference on Extending Database Technology. Heidelberg, Springer-Verlag, 1996, pp13-20.
8. Jiawei Han, Micheline Kamber, Simon Fraser University, A book on "Data Mining: Concepts and Techniques", Academic Press, Morgan Kaufmann Publishers, 2001, pp. 227-240.
9. Zaki, "Parallel sequence mining on share-memory machines", Journal of Parallel and Distributed Computing. vol. 61, pp401-426, 2001.
10. M. J. Zaki, S. Parthasarathy and W. Li., "Parallel data mining for association rules on shared memory multi- processors". In Supercomputing 96, Pittsburg, PA, November 1996, pp. 17-22.
11. W. Jian, and L. Xingming, "An efficient association rule mining algorithm in distributed database," the first International Workshop on Knowledge Discovery and Data Mining. 2008, pp108-113.
12. Q. Shaojie, T. Changjie, D. Shucheng, Z. Mingfang, P. Jing, L.Hongjun, and K. Yungchang,, "PartSpan: Parallel Sequence Mining of Trajectory patterns," the fourth International Conference on Fuzzy Systems and Knowledge Discovery. 2008, pp363-367.
13. Y. Kunming, Z. Jiayi, H. Tzungpei, and Z. Jialing, "A load-balanced distributed parallel mining algorithm," Expert Systems with Applications. vol.37, pp2459-2464, 2009.
14. V. Guralnik, N. Garg, and G. Karypis, "Parallel tree projection algorithm for sequence Mining," proceed ings of the 7th International Euro-Par Conference Manchester on Parallel Processing. London, UK: Springer-Verlag , 2001, pp310-320.
15. H. Jiawei, C. Shengnan, and P.David, "Parallel mining of Closed sequence patterns," proceedings of the 11th ACM SIGKDD International Conference on Knowledge Discovery in data mining. USA:New York,2005, pp562-567.
16. F. Niagara, "A Parallel Mining Algorithm for Closed sequence Patterns," proceedings of the 21st International Conference on Advanced Information
17. Kun-Ming Yu, Jiayi Zhou and Wei Chen Hsiao, "Load Balancing Approach Parallel Algorithm for Frequent Pattern Mining", V. Malyshkin (Ed.): PaCT 2007. © Springer-Verlag Berlin Heidelberg 2007.LNCS 4671, pp. 623–631. Networking and Applications Workshops.. 2007, pp392-395.
18. A book "Introduction to Parallel computing", by A. Grama, A. Gupta, G. Karypis and V. Kumar, second edition, published by Pearson Education, pp. 95, 110-112, 115-120.
19. International Journal of Advancements in Technology http://ijict.org/ ISSN 0976-4860 Exploiting Parallelism in Association Rule Mining Algorithms Rakhi Garg, P. K. Mishra Department of Computer Science, Banaras Hindu University, Varanasi, Uttar Pradesh-221005, India
20. Parallel Data Mining Algorithms for Association Rules and Clustering Jianwei Li Northwestern University Ying Liu DTKE Center and Grad. Univ. of CAS Wei-keng Liao Northwestern University Alok Choudhary Northwestern University

AUTHORS PROFILE

Manalisha Hazarika , She received the bachelor's degree from Gauhati University in 2009. . She did her post-graduation from Gauhati University, 2011.She did her master of Technology in Information Technology from Gauhati University in2013

Mirzanur Rahaman He received the bachelor's degree in Jorhat Engineering collegeat 2006. He did his post-graduation from Tezpur
University, India in Information Technolgy 2008. he is working as an Assistant Professor in Department of Information Techonlogy in Gauhati University from January 2010.