A Survey On Distributed Data Mining Process Via Grid

Bagrudeen Bazeer Ahamed[1] and Shanmugasundaram Hariharan[2]

Assistant Professor[1], Associate Professor[2]
Department Of Information Technology
Pavendar Bharathidasan College of Engineering & Technology,
Tiruchirapalli -620009.
bazeerahamed@gmail.com, mailtos.hariharan@gmail.com

Abstract

Distributed data mining (DDM) techniques have become necessary for large and multi-scenario datasets requiring resources, which are heterogeneous and distributed in nature. In this paper, we focus our attention on distributed data mining approach via grid. We have discussed and analyzed a new framework based on grid environments to execute new distributed data mining approaches that best suits a distributed and heterogeneous datasets that are commercially available. The architecture and motivation for the design have also been presented in this paper. A detailed survey on distributed data mining technology was also carried out which could offer a better solution since they are designed to work in a grid environment by paying careful attention to the computing and communication resources.

Keywords: *Data mining; distributed databases; parallel databases warehousing; grids*

1. Introduction

In Internet era, the volume of data available for public usage is high. But the retrieved information catering the needs of an end user from such voluminous repositories remains to be a challenge. There exists variety of data types like flat files, text formats or any other multimedia modes[1]. Whereas the dataveillance and data-gathering tools described in the preceding section are used mainly to monitor and record activities of online users, other tools are used to exchange that data on the Internet[1]. This exchange of online personal information often involves the sale of personal data to third parties, which has resulted in commercial profits for certain online entrepreneurs, often without the knowledge and consent of individuals about whom the data is exchanged.

The paper is structured as follows. Section 1 provided some general information of current information retrieval tasks, while subsections in section 1 focus on distributed approaches. Section 2 details the existing developments in distributed data mining, while section 3 deals on GRID platforms and associated tool kits for GRID environment. Different types of algorithms for distributed data mining are analyzed in section 4, followed by the future focus on GRID in section 5. Finally section 6 provides conclusion future work.

1.1. Distributed Approach in a Database

Data mining has attracted a great deal of attention in the information industry and in society as a whole in recent years, due to wide availability of huge amounts of data and the imminent need for turning such data into useful information and knowledge. Data mining refers to extracting or “mining” knowledge from large amounts of data. Distributed data

mining (DDM) is data mining where the data and computation are spread over many independent sites. Each site has its own data source and data mining algorithms producing local models. From them global meaningful knowledge has to be derived. The Grid is a distributed computing infrastructure that enables coordinated resource sharing within dynamic organizations.

Association rules are most popularly used to show the relationships between data items with support and confidence measurements [1]. For example, the user can create an experiment that runs several schemes against a series of datasets and then analyze the results to determine if one of the schemes is statistically better than the other schemes [2]. Based on these concepts the study is being carried out in the subsequent headings. The goal of any distributed algorithm is given below (but not limited to) [5]:

- The character tics of local business processes
- The usability and reliability of data
- Average distribution of working load
- Memory expense
- The method of data design

Having discussed on some of the issues on distributed transactions in a database using data mining concepts, we move on to discuss the same in an data warehouse in the next sub section.

1.2. Distributed approach in a trusted Data Warehouse

Distributed data mining intends to get the global knowledge from the local data at distributed sites – tables. Data in data storage are distributed into different tables: fact table and dimension table. Data warehouse (DW) is a system that extracts, cleans, conforms, and delivers source data into a dimensional data store and then supports and implements querying and analysis for the purpose of decision making. A detailed description or comparison of OLAP and OLTP is available in [20].

The typical approach to deal transactions using distributed data in a data warehouse containing all voluminous data in variety of formats is quiet challenging and tricky [6]. This requires that the warehouse be trusted to maintain the privacy of all parties - since it knows the source of data, it learns site-specific information as well as global results. The overall architecture is shown in figure 1.

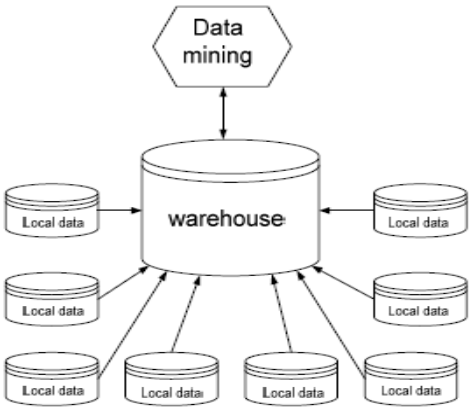

Figure 1. Build a data warehouse containing all the data

1.3. Parallel and distributed data mining

This section focus on the difference involved in parallel and distributed mining which is enormity and high dimensionality of datasets typically available as input to the problem of association rule discovery, makes it an ideal problem for solving multiple processors in parallel.[1] The primary reasons are the memory and CPU speed limitations faced by single processors. Thus it is critical to design efficient parallel algorithms to do the task. Another reason for parallel algorithm comes from the fact that many transactions databases are already available in parallel databases or they are distributed at multiple sites to begin with. The cost of bringing them all to one site or one computer for serial discovery of association rules can be prohibitively expensive.

For compute-intensive applications, parallelization is an obvious means for improving performance and achieving scalability. A variety of techniques may be used to distribute the workload involved in data mining over multiple processors. Four major classes of parallel implementations are distinguished. The classification tree in Figure 2 demonstrates this distinction. The first distinction made in this tree is between task parallel and data-parallel approaches.

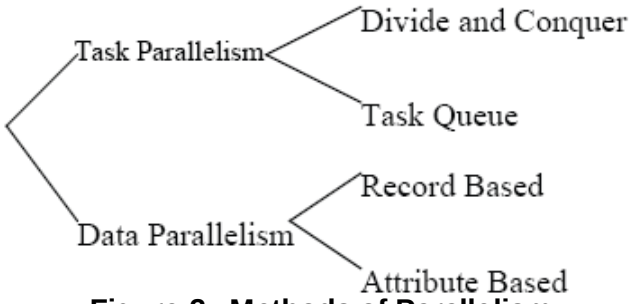

Figure 2: Methods of Parallelism

1.4. Distributed data mining

This Section deals with Distributed Data Mining Data process in which mining algorithms deal predominantly with simple data formats (typically flat files); there is an increasing amount of focus on mining complex and advanced data types such as object-oriented, spatial and temporal data. Another aspect of this growth and evolution of data mining systems is the move from stand-alone systems using centralized and local computational resources towards supporting increasing levels of distribution [2]. As data mining technology matures and moves from a theoretical domain to the practitioner's arena there is an emerging realization that distribution is very much a factor that needs to be accounted for.

Databases in today's information age are inherently distributed. Organizations that operate in global markets need to perform data mining on distributed data sources (homogeneous/ heterogeneous) and require cohesive and integrated knowledge from this data. Such organizational environments are characterized by a geographical separation of users from the data sources [3]. This inherent distribution of data sources and large volumes of data involved inevitably leads to exorbitant communications costs. Therefore, it is evident that traditional data mining model involving the co-location of users, data and computational resources is inadequate when dealing with distributed environments. The development of data mining along this dimension has lead to the emergence of distributed data mining. The need to

address specific issues associated with the application of data mining in distributed computing environments is the primary objective of distributed data mining. Broadly, data mining environments consist of users, data, hardware and the mining software (this includes both the mining algorithms and any other associated programs). Distributed data mining addresses the impact of distribution of users, software and computational resources on the data mining process. There is general consensus that distributed data mining is the process of mining data that has been partitioned into one or more physically/geographically distributed subsets [6].

The significant factors, which have led to the emergence of distributed data mining from centralized mining [4], are as follows:

- The need to mine distributed subsets of data, the integration of which is non-trivial and expensive.
- The performance and scalability bottle necks of data mining.
- Distributed data mining provides a framework for scalability, which allows the splitting up of larger datasets with high dimensionality into smaller subsets that require computational resources individually.

Distributed Data Mining (DDM) is a branch of the field of data mining that offers a framework to mine distributed data paying careful attention to the distributed data and computing resources.[5] In the DDM literature, one of two assumptions is commonly adopted as to how data is distributed across sites: homogeneously and heterogeneously. Both viewpoints adopt the conceptual viewpoint that the data tables at each site are partitions of a single global table. In the homogeneous case, the global table is horizontally partitioned.

The tables at each site are subsets of the global table; they have exactly the same attributes. In the heterogeneous case the table is vertically partitioned, each site contains a collection of columns (sites do not have the same attributes). However, each tuple at each site is assumed to contain a unique identifier to facilitate matching. It is important to stress that the global table viewpoint is strictly conceptual. It is not necessarily assumed that such a table was physically realized and partitioned to form the tables at each site.

Applications in parallel and distributed data mining

The technology of parallel and distributed data mining can be applied on different real time applications. The major applications are listed below:

- Credit card fraudulent detection
- Intrusion detection
- Business analysis – prediction etc.
- Financial applications
- Astrological events
- Anomaly Detection

1.5. Grid computing as a technique for distributed scenario

Today amounts of data are collected and warehoused. Data sets are generated and stored at enormous speed in local databases, from remote sources or from the sky. At the same time, scientific simulations generating terabytes of data are performed in many laboratories. E-commerce and e-business applications store and manage huge databases about products, clients and transactions. Unfortunately, we are much better at storing data than extracting

knowledge from it. Large datasets are hard to understand and traditional techniques are infeasible for raw data.

Data mining helps scientists in hypothesis formation in biology, medicine, physics, and engineering. Companies use data mining techniques to provide better, customized services and support decision making. In all these different areas, massive data collections of terabyte and petabyte scale need to be used and analyzed. Moreover, in many cases datasets must be shared by large communities of users that pool their resources different sites belonging to a single company, or from a large number of laboratories, plants, or public organizations.

Grid computing has been proposed as a novel computational model, distinguished from conventional distributed computing by its focus on large-scale resource sharing, innovative applications, and, in some cases, high-performance orientation. Today grids can be used as effective infrastructures for distributed high-performance computing and data processing [1]. A grid is a geographically distributed computation infrastructure composed of a set of heterogeneous machines that users can access via a single interface. Grids therefore, provide common resource-access technology and operational services across widely distributed virtual organizations composed of institutions or individuals that share resources.

Although originally intended for advanced science and engineering applications, grid computing has emerged as a paradigm for coordinated resource sharing and problem solving in dynamic, multi-institutional virtual organizations in industry and business [2]. Grid applications include the following:

- Intensive simulations on remote supercomputers;
- Cooperative visualization of very large scientific data sets;
- distributed processing for computationally demanding data analysis;
- coupling of scientific instruments with remote computers and data archives.

In the last five years, toolkits and software environments for implementing grid applications have become available. These include Legion [3], Condor [4], and Unicore [5]. In particular, Foster and Kesselman's Globus Toolkit [6] is the most widely used middleware in scientific and data-intensive grid applications, and is becoming a de facto standard for implementing grid systems. The toolkit addresses security, information discovery, resource and data management, communication, fault detection, and portability issues. It does so through mechanisms, composed as bags of services that execute operations in grid applications. Today, Globus and the other grid tools are used in many projects worldwide. Although most of these projects are in scientific and technical computing, there is a growing number of grid projects in education, industry, and commerce. Together with the grid shift toward industry and business applications, a parallel shift toward the implementation of data grids has been registered.

Data grids are designed to allow large data sets to be stored in repositories and moved with almost the same ease that small files can be moved. They represent an enhancement of computational grids, driven by the need to handle large data sets without repeated authentication, aiming to support the implementation of distributed data-intensive applications. Significant examples are the EU Data Grid [7], the Particle Physics Data Grid [8], the Japanese Grid Data Farm [9], and the Globus Data Grid [10] project. Data grid middleware is central for management of data movement and replication on grids. Furthermore, in many scientific and business areas it is necessary to use tools and environments for analysis, inference and discovery over the available data. Scientists and engineers can use those environments for implementing grid-based problem solving environments for doing "virtual" scientific experiments. Analysts can follow the same

approach in mining large volumes of data to support decision making. Therefore, the evolution of data grids is represented by knowledge grids offering high-level tools and models for the distributed mining and extraction of knowledge from data repositories available on the grid [11].

The development of such an infrastructure is the main goal of our research work, focused on the design and implementation of an environment for geographically distributed high-performance knowledge discovery applications called "KNOWLEDGE GRID"[5] (explained in detail in section 3).

- Parallel computing

Single systems with many processors work on same problem.

- Distributed computing

Many systems loosely coupled by a scheduler to work on related problems.

- Grid Computing (Meta Computing)

Many systems tightly coupled by software, perhaps geographically distributed, are made to work together on single problems or on related problems.

2. Existing Developments of Distributed Data Mining in Data Set

Data mining involves the use of sophisticated data analysis tools to discover previously unknown, valid patterns and relationships in large data sets (see Figure 3). Data mining consists of more than collecting and managing data; it also includes analysis and prediction [7]. This means data mining consists of the collection and management of data associated with analysis and prediction of future outcomes.

However, the situation arises where information may be located in different places, in different physical locations. Therefore, the goal is to effectively mine distributed data which is located in heterogeneous sites. Examples of this include biological information located in different databases, data which comes from the databases of two different firms, or analysis of data from different branches of a corporation, the combining of which would be an expensive and time-consuming process. Nowadays, the information overload means big problem, so data mining algorithms working on very large data sets take very long times on conventional computers to get results.

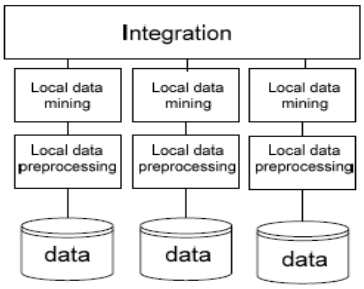

Figure 3. Data Mining Architecture

One approach to solve this problem is parallel computing – parallel data mining algorithms can offer an effective way to mine very large data sets. For Example by processing data from

car purchasing and by subsequent constructing new prices companies can run across decision - making problems on how to correctly apply constructing prices to sell utmost and to have the highest profit. Data from sales and the price structure of particular cars – the prices of different components and various discounts – are stored in diverse archives and because of the easier accessibility, also in a data warehouse. Parallel and distributed knowledge discovery is based on the use of networks for the mining of data in a distributed and parallel fashion. It is possible to manage and analyze data, which is geographically distributed in different data warehouses [4]. There are vertically distributed data structures, where the instances are represented by the couple attribute – value. Data in this set can contain errors or attribute values can be missing.

Even though there are two new parallel algorithms for mining association rules. The IDD (Intelligent Data distribution) algorithm effectively parallelizes the step of building hash tree and is, thus, scalable with respect to the increasing candidate set size. This algorithm also utilizes total main memory available more effectively than the CD (count distribution) algorithm. This is important if the I/O cost becomes dominant due to slow I/O system. The IDD algorithm improves over the DD (Data Distribution) algorithm which has high communication overhead and redundant work. These are some Data distribution techniques involved before evolution of grid services, when grid services applied in the DD (Data Distribution), Generic Integration of data distribution services are involved in the grid environment

3. Knowledge Grid

In this section the KNOWLEDGE GRID[5] in the parallel and distributed software architecture that integrates data mining techniques and grid technologies are studied . In the KNOWLEDGE GRID architecture data mining tools are integrated with generic and data grid mechanisms and services. Thus the KNOWLEDGE GRID can be exploited to perform data mining on very large data sets available over grids, to make scientific discoveries, improve industrial processes and organization models, and uncover business valuable information and data grid services. This approach benefits from "standard" Grid services that are more and more utilized and offers an open parallel and distributed knowledge discovery architecture that can be configured on top of grid middleware in a simple way.

The KNOWLEDGE GRID architecture uses basic grid mechanisms to build specific knowledge discovery services on top of grid toolkits and services. These services can be developed in different ways using the available grid environments. The current implementation is based on the Globus Toolkit [14]. Like Globus, the KNOWLEDGE GRID offers global services based on the cooperation and combination of local services. We designed the KNOWLEDGE GRID architecture so that more specialized data mining tools are compatible with lower-level grid mechanisms.

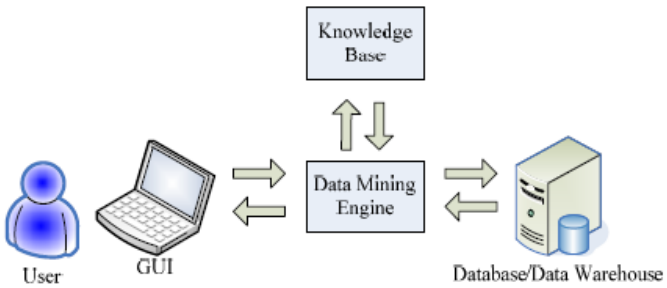

Figure 4. Architecture of Data mining System

- *The Knowledge Directory Service (KDS)* extends the basic globus monitoring and discovery service and manages metadata describing data and tools used in the *KNOWLEDGE GRID*. These include:
- Repositories of data to be mined (data sources).
- Tools and algorithms used to extract, filter and manipulate data; tools to mine data and visualize and store mining results.
- Distributed *execution plans*. An execution plan is an abstract description of a distributed data mining application, that is a graph describing the interaction and data flow between data sources, data mining tools, visualization tools, and result storage facilities.
- Knowledge obtained as result of the mining process, i.e., learned models and discovered patterns.

3.1 Globus Toolkit Services

The main services offered by Globus Toolkit 2 are the following:

- *Grid security infrastructure (GSI)*. Enables secure authentication and communication over an open network providing a number of services, including mutual authentication and single sign-on run-anywhere authentication, with support for local control over access rights and mapping from global to local user identities [15]. GSI is based on public key encryption, X.509 certificates, and the secure sockets layer (SSL) communication protocol.
- *Monitoring and discovery service (MDS)*. Provides a framework for publishing and accessing information about grid resources [16] by using the lightweight directory access protocol (LDAP) as a uniform interface to such information. MDS provides two types of directory services: the grid resource information service (GRIS) and the grid index information service (GIIS). A GRIS can answer queries about the resources of a particular grid node; examples of information provided include host identity (e.g., operating systems and versions), as well as more dynamic information such as current CPU load and memory availability. A GIIS combines the information provided by a set of GRIS services managed by an organization, giving a coherent system image that can be explored or searched by grid applications.
- *Globus resource allocation manager (GRAM)*. Provides facilities for resource allocation and process creation, monitoring, and management. GRAM simplifies the use of remote systems by providing a single standard interface for requesting and using remote system resources for the execution of jobs. The most common use of GRAM is remote job submission and control, to support distributed computing applications.
- *Dynamically-updated resource online co-allocator (DUROC)*. Manages multirequests of resources, delivers requests to different GRAMs and provides time-barrier mechanisms among jobs [11]. In Globus, a GRAM provides an interface to submit jobs on a particular set of physical resources, whereas the DUROC is used to coordinate transactions with independent GRAMs.
- *Heartbeat monitors (HBM)*. Provides a mechanism for monitoring the state of processes. The HBM is designed to detect and report the failure of processes that have identified them to the HBM. It allows simultaneous monitoring of both

Globus system processes and application processes associated with user computations [4]. The HBM also provides notification of process status exception events, so that recovery actions can be taken.

- *GridFTP*. Implements a high-performance, secure data transfer mechanism based on an extension of the FTP protocol that allows parallel data transfer, partial file transfer, and third-party (server-to-server) data transfer, using GSI for authentication [5]. This allows grid applications to have ubiquitous, high-performance access to data in a way that is compatible with the most popular file transfer protocol in use today.
- *Replica catalog and replica management*. Provide facilities for managing data replicas, i.e., multiple copies of data stored in different systems to improve access across geographically-distributed grids. The replica catalog provides mappings between logical names for files and one or more copies of the files on physical storage systems; it is accessible via an associated library and a command-line tool. The replica management combines the replica catalog (for keeping track of replicated files) and Grid FTP (for moving data) to manage data replication.

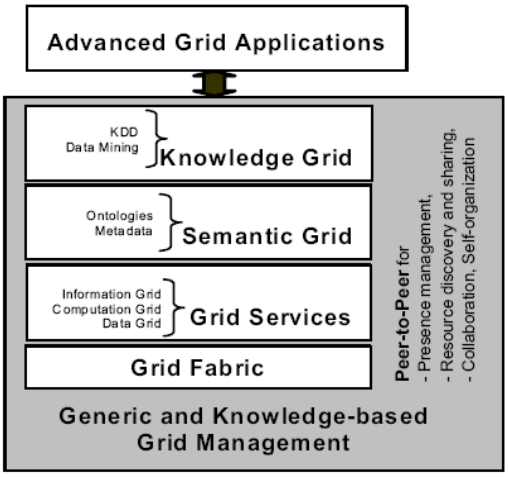

Figure 5. Building Knowledge grid services

3.2 DDM Using Grid Architecture

A possible infrastructure for a virtual organization, implemented using Grid technologies, is presented in Figure.vi. The company has a central branch and several local branches (LB) [8]. Each branch is composed of a number of Grid nodes (GN) interconnected in a Grid infrastructure. In the case study, the data mining task is the discovery of the association rules in the local branch databases, and the implementation of the Grid infrastructure is based on the Globus toolkit [5]. In the OGSA context, the association rules discovery task is exposed in the form of Grid services.

The mining service has several components specific to a Grid service: service data access, service data element, and service implementation. The association rules discovery service is interacting with the rest of the grid services: service registry, service creation, authorization, notification, manageability and concurrency. There are two types of grid services they are

Apriori and Predictive Apriori Algorithms in which the Apriori Grid Service must comply with OGSA[5] rules, constraints, standard interfaces and behavior.

4. Comparison of Predictive Apriori and Apriori Algorithms

Apriori algorithms that are most dominantly investigated are apriori and predictive apriori algorithms [20] which were run using weka toolkit on a centralized database. Sujni Paul and Sumithra (2010) have concluded the following:

- Pred. Apriori mines a higher quality set of rules
- Pred. Apriori needs fewer rules
- But pred. Apriori is slower than Apriori

The service data access contains a standard interface and a discovery service, for registering information about Grid service instances with registry services. The client application calls the standard method Find Service Data, which retrieves service information from individual Apriori Grid Service instances [7]. The service data access defines a standard interface and semantics for dynamic service creation of the Apriori Grid Service, located at the Service Data Element level.

Since Apriori Algorithm is used in the open grid service environment it compares with similar set of rules which is fast retrieve data when comparing with Predictive Apriori mines [12]. In the survey of car industries the testimony it is to be denoted as to survey easier to sell the product as service as well as in high profitable manner.

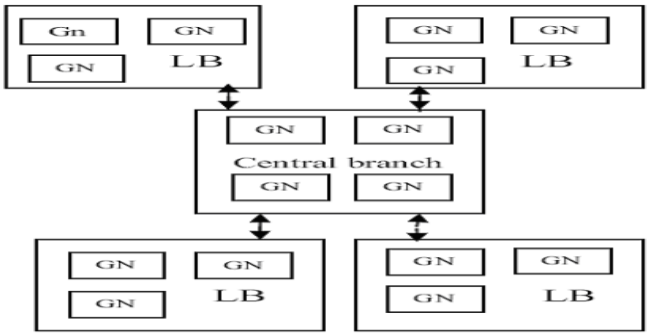

Figure 6. Virtual Organization infrastructure using grid technologies

In the case study, from the central branch the association rules discovery task has to be launched, mining the data from central and local branches. It is a distributed framework for mining associative rules in geographic distributed databases. The framework implements VO structure and it uses data mining methods and techniques and different technologies [17] like Grid Java and relational databases. The client module allows the user to send the parameters and the commands to the rest of the components. Configuration parameters of the framework and of the Apriori algorithm [20], then call the mining methods.

The client finds and creates the Apriori Grid Service and sends it the parameters for: locating and connecting to the remote transaction databases, locating the transaction tables or calling the pre-processing methods, Apriori algorithm minimum support and confidence, locating the knowledge base with the partial rules generated [15]. In the next step, the Apriori Grid Service instances perform the association rules discovery task independently on the remote databases, located on the Central Branch or on the Local Branches of the VO [4].

The user/client application receives notifications when an association mining service completes its job and the results can be explored from the knowledge base. The user can explore the partial association rules generated or can apply incremental methods to combine the partial rules into more general rules [20]. The Apriori Grid Service implements an original Apriori library for mining the associative rules. The library is implemented in Java for portability and compatibility with the implementation [1] of the Globus 3 Toolkit Core services [4] and, in this phase, it contains the serial version of the algorithm.

5. Future of GRID

To face the growing complexity of Grids and the overwhelming amount of data to be managed, main requirements of future Grids will be:

- Knowledge discovery and knowledge management functionalities, for both user's needs (intelligent exploration of data, etc.) and system management;
- Semantic modeling of user's tasks/needs, Grid services, data sources, computing devices (from ambient sensors to high-performance computers), to offer high level services and dynamic services finding and composition;
- Pervasive and ubiquitous computing, through environment/context awareness and adaptation;
- Advanced forms of collaboration, through dynamic formation of virtual organizations;
- Self-configuration, autonomic management, dynamic resource discovery and fault tolerance.

6. Conclusion & Future Work

With the rapid development of next-generation network technologies, distributed data mining has been recognized as one of the most important information technologies for automating the process of analyzing and interpreting large volumes of data in modern knowledge industries and high-tech sectors such as science, engineering and medicine. Currently, no coherent framework exists for developing and deploying data mining applications on the network. There are a number of data mining trends is in terms of technologies and methodologies which are currently being developed and researched. These trends include methods for analyzing more complex forms of data, as well as specific techniques and methods. The trends identified include distributed data mining, hypertext/hypermedia mining, and ubiquitous data mining, as well as multimedia, spatial, and time series/sequential data mining. The future scope of this work is that parallel distribution of knowledge extraction could be made on a distributed grid environment to produce a more optimized result.

References

[1] M. Z. Ashra_, D. Taniar, and K. A. Smith, "A Data Mining Architecture for Distributed Environments", IICS 2002, Vol 2, pages 27-38, 2002.

[2] Albert Y. Zomaya, Tarek El-Ghazawi, Ophir Frieder, "Parallel and Distributed Computing for Data Mining", IEEE Concurrency, 1999.

[3] The Grid Security Infrastructure. The Globus Project. [Online]. Available: http://www.globus.org/security.(last accessed on 23.2.2011).

[4] The Globus Resource Specification Language RSL v1.0. The Globus Project. (available Online at: http://www.globus.org/gram/rsl_spec1. html)

[5] M.Cannataro and D. Talia, "KNOWLEDGE GRID Architecture for Distributed Knowledge Discovery", CACM, Vol. 46, No. 1, pp. 89-93, 2003.

[6] Zakim J, Pan Y, "Introduction: recent developments in parallel and distributed data mining," Journal of Distributed Parallel Database, Vol. 11, pp. 123-127, 2002.

[7] Matthew Eric Otey, Srinivasan Parthasarathy, " Parallel and Distributed Methods for Incremental Frequent Item set Mining," IEEE Transaction on Systems, Man and Cybernetics- Part B, Cybernetics, Vol. 34, No. 6, Dec. 2004.

[8] Antonio Congiusta, Domenico Talia, Paolo Trunfio, "Service-oriented middleware for distributed data mining on the grid," Journal of Parallel and Distributed Computing, Vol. 68, pp. 3–15, 2008.

[9] A. Grama, A. Gupta, and V. Kumar, "Isoefficiency: Measuring the Scalability of Parallel Algorithms and Architectures," IEEE Parallel and Distributed Technology, vol. 1, no. 3, pp. 12–21, Aug. 1993.

[10] C.H. Papadimitriou and K. Steiglitz, Combinatorial Optimization: Algorithms and Complexity. Englewood Cliffs, NJ: Prentice-Hall, 1982.

[11] T. Shintani and M. Kitsuregawa, ªHash Based Parallel Algorithms for Mining Association Rules, Proc. Conf. Paralellel and Distributed Information Systems, 1996.

[12] The Monitoring and Discovery Service. The Globus Project. [Online].Available: http://www.globus.org/mds (last accessed 21.05.2010).

[13] D. Talia, "The Open Grid Services Architecture: Where the Grid Meets the Web", IEEE Internet Computing, Vol. 6, No. 6, pp. 67-71, 2002.

[14] D. Talia, P. Trunfio, "Toward a Synergy between P2P and Grids", IEEE Internet Computing, vol. 7, no. 4, pp. 94- 96, 2003.

[15] OGSA-WebDB. http://www.gtrc.aist.go.jp/dbgrid/ogsawebdb/ (last accessed 14.02.2010.

[16] Michael D. Beynon, Tahsin Kurc, Alan Sussman, and Joel Saltz. Optimizing execution of component-based applications using group instances. In Proceedings of the Conference on Cluster Computing and the Grid (CCGRID), pp. 56–63, May 2001.

[17] Jiawei Han, Micheline Kamber, "Data Mining Concepts and Techniques", Morgan Kaufmann Publishers, 2002.

[18] Cristian Aflori, Mitica Craus," Grid implementation of the Apriori algorithm", Science direct, Oct 2006.

[19] Aparna S Varde, Makiko Takakshi, Elke A Rundensteiner, Mathew Oward, "Apriori algorithm and game-of-life for predictive analysis in materials science", International Journal of knowledge based and intelligent engineering systems, 2004.

[20] Sujni Paul and Mrs. R. Sumithra, "Using distributed apriori association rule and classical apriori mining algorithms for grid based knowledge discovery", In Proceedings of International conference on Computing, Communication and Networking Technologies, Coimbatore, India, Oct 2010.

[21] J.M. Conroy, "Parallel Nested Dissection," Parallel Computing, vol. 16, pp 139–156, 1990.

Authors

B.Bazeer Ahamed received a Bachelor of Technology in Anna University, Chennai and Master in Computer Science in Anna University of Technology, Tiruchirapalli. He has four years of teaching experiences. At present he is working as Assistant Professor in IT department, Pavendar Bharathidasan College of Engineering and Technology, Trichirappalli-9. He has published more than three research papers in the international journals and international conferences. His research interests include Data Mining, Software Engineering and distributed computing. His career plan is to continue the research in the Data Mining and Data Warehousing.

Dr. S. Hariharan received his undergraduate and masters degree in Engineering in the field of Computer Science from Madurai Kammaraj University and Anna University, Chennai respectively. Then he received his Ph.D in Computer Science Engineering in the year 2010 from Anna University, Chennai, India. He has acquired 7 years of experience in handling U.G & PG programmes in various academic affiliations. He started his career at Crescent Engineering College as Lecturer in the year 2004. Currently he is working as Associate Professor in Department of Information Technology at Pavendar Barathidasan College of Engineering and Technology, Trichy, India. His research interests include Computer Networks, Wireless Communications, Information Retrieval, Data mining, Web Mining, and Text Analysis. He has published 35 papers in referred journals and conferences. He also serves as editor, associate editor and reviewer for several international journals and conferences.