Transactions D:
Computer Science & Engineering and
Electrical Engineering
Vol. 17, No. 1, pp. 1–10

SCIENTIA IRANICA

Design and Implementation of WSRF-Compliant Grid Services for Mining Fuzzy Association Rules

M. Deypir[1], G.H. Dastghaibyfard[1] and M.H. Sadreddini[1,*]

Abstract. *Data mining is a widely used approach for the transformation of large amounts of data to useful patterns and knowledge. Fuzzy association rules mining is a data mining technique which tries to find association rules without the effect of sharp boundary problems when data contains continuous and categorical attributes. Grid data mining is a new concept, which allows the data mining process to be deployed and used in a data grid environment where data and service resources are geographically distributed. In this paper, a grid service for mining fuzzy association rules is developed. The service is implemented based on recently proposed Data Mining Grid Architecture (DMGA) and uses the Web Service Resource Framework (WSRF). Experimental evaluations, after implementing and deploying the service, show the effectiveness and acceptable performance of the proposed grid service. Additionally, in this study, a new algorithm, namely FFDM, is developed to mine fuzzy association rules without raw data exchange, using the distributed storage of data grid environments. Empirical evaluation of FFDM reveals the scalability and efficiency of the proposed method, in addition to the advantages of minimum messaging and providing privacy of data.*

Keywords: *Fuzzy association rules mining; Grid computing; Data mining; Data grid.*

INTRODUCTION

Complex businesses and scientific applications require access to distributed resources (e.g. computers, databases, networks, sensors etc.). Grids have been designed to support applications that can benefit from high performance, distribution, collaboration, data sharing and the complex interaction of autonomous and geographically distributed resources. Grid computing is suitable for both computation and data intensive tasks. The grid is a distributed computing infrastructure that enables coordinated resource sharing within dynamic organizations consisting of individuals, institutions and resources. The main aim of grid computing is to give organizations and application developers the ability to create distributed computing environments that can utilize computing resources on demand. Grid computing can leverage the computing power of a large numbers of server computers, desktop PCs, clusters and other kinds of hardware. Therefore, it can help increase efficiency and reduce the cost of computing networks by decreasing data processing time, optimizing resources and distributing workloads, thereby, allowing users to achieve much faster results regarding large operations and at a lower cost.

Data and computational grids are two important categories of grid environments. A data grid is a dynamic logical namespace that enables the coordinated sharing of heterogeneous distributed storage resources and digital entities based on policies across domains. In a data grid environment, there is a single set of files that can be accessed without regard to the location and platform of the system on which the files are resident. Large amounts of scientific and business data are stored in data grids, in scales of tera and peta bytes. Data mining is an approach to extract hidden and probably useful knowledge from large databases and datasets. Grid data mining aims to apply data mining techniques to geographically distributed data in the heterogeneous environment of data grids. Data mining grids use grid technology to solve the problem of data mining scalability, and fit the problem into complex scenarios in which virtual organization are involved. Recently, the proposed Data Mining Grid Architecture (DMGA) [1] defines a set of services for data mining and usage

1. *Department of Computer Science and Engineering, School of Engineering, Shiraz University, Shiraz, Iran.*
*. *Corresponding author. E-mail: sadredin@shirazu.ac.ir*

Received 12 December 2008; received in revised form 19 September 2009; accepted 16 November 2009

patterns for their composition in a real scenario. Data mining has different functions and techniques including classification, clustering, association rules etc. In this paper, the main concentration is on the problem of mining association rules [2], when continuous and categorical attributes appear in data by means of a fuzzy concept in the grid environment. Both binary and fuzzy association rules [3] have a wide application area in the field of marketing, basket analysis and etc. The popular Apriori [4] algorithm is a base algorithm for mining traditional binary association rules. Both [1] and [5] present implementation of the Apriori algorithm in the grid environment. The main contribution of this paper is to the design and implementation of grid services based on DMGA architecture for mining fuzzy association rules in data grids. The remainder of the paper is as follows: The following section reviews previous work on data mining and grid based data mining. Subsequently, the problem of mining fuzzy association rules is revisited. Since the proposed grid service is based on Open Grid Service Architecture (OGSA) and WSRF concepts of Grid technology, these concepts are also reviewed. The reasons for selecting DMGA as a base framework for our proposed methods are also provided. Additionally, specifications of the fuzzy association rules mining grid service based on DMGA architecture are described. In this paper, the motivation for distributed mining of fuzzy association rules is noted and a method for mining fuzzy association rules in data grid environments, namely FFDM, is proposed. Implementation and deployment of grid services for mining fuzzy association rules, as well as FFDM, are discussed. In the Experimental Evaluation section, some performance evaluations of grid services and an empirical evaluation of the FFDM in term of scalability are presented. Finally, conclusions and open issues are presented.

RELATED WORKS

Data mining algorithms and knowledge discovery processes are both computing and data intensive, therefore, the grid offers a suitable computing and data management infrastructure for supporting decentralized and parallel data analysis. The opportunity of utilizing grid-based data mining systems, algorithms and applications is increasing due to the need of users to analyze data that is distributed across geographically dispersed heterogeneous hosts. Grid-based data mining would allow corporate companies to distribute computing-intensive data analysis among large numbers of remote resources. At the same time, it can lead to new algorithms and techniques that would allow organizations to mine data where it is stored. This is in contrast to the practice of having to select data and transfer it into a centralized site for the mining process.

The problem of Association Rules Mining (ARM) was first introduced by Agrawal et al. [2] who proposed the well-known Apriori algorithm [4]. After introduction of the Apriori algorithm, a large number of improvements were proposed to enhance the efficiency of the problem. In order to cope with the problem of continuous variables and sharp boundary problems, Fuzzy Association Rules Mining (FARM) was introduced [3]. Mining fuzzy association rules is more general than the traditional association rules mining, because it can accept all types of attribute including binary, continuous and categorical, whereas the traditional association rules mining only accepts data having binary attributes. Since in data grids like EU DataGrid [6] and GridPP [7], a large amount of data is stored about scientific and business applications, including continuous attributes, it is more suitable to use fuzzy association rules mining to extract hidden rules. This study is in the context of the data mining grid and tries to utilize previous work in this field to use fuzzy association rules in grid environments.

In [5], an implementation of the Apriori algorithm in the grid environment is proposed. In this implementation, for the mining task, the Apriori algorithm is used on top of the Globus toolkit [8]. In [9], a grid based approach for enterprise-scale data mining is proposed, which is based on leveraging parallel database technology for data storage and on-demand computing services for parallelism in statistical computation. In that study, the algorithmic decomposition of data mining kernels between the data storage and compute grids is described, which make it possible to exploit the parallelism on the respective grids in a simple way, while minimizing data transfer between these grids.

The most well known effort for integrating the computational grid and data mining technique is the Knowledge Grid [10]. The Knowledge Grid offers global services based on the cooperation and combination of local services. The system architecture is more specialized for data mining tools that are compatible with a lower-level grid and, also, data grid services. Giannadakis et al. [11] propose an architecture called InfoGrid that focuses on data integration. This infrastructure includes a layer of Information Integration Services which enable heterogeneous information resources to be queried effectively. In [12], fuzzy association rules mining are used to discover patterns in performance monitoring data. These patterns are used to optimize the scheduling process.

In this paper, the design and implementation of a grid service for mining fuzzy association rules are described. This service is useful for large organizations, environments and enterprises which manage and analyze data that are geographically distributed in heterogeneous data repositories and warehouses. Fuzzy data mining is not only useful for user demand

application but is also needed to optimize the grid job scheduling strategy. The proposed grid service can be invoked by internal operation of the job scheduler inside the grid middleware system.

Another contribution of this paper is towards enhancing the performance of the mining process by performing the process in a distributed fashion using the processing power of data grid nodes. In this way, there is no need for raw data exchange, since data items are mined on nodes on which data are resident. Therefore, computational power, storage and the network resourced are utilized effectively in the data grid. In the other words, the privacy of data collections of local nodes is preserved and there is no need to move large amounts of data due to the limited bandwidth of the network and storage space.

FUZZY ASSOCIATION RULES MINING

Since the main contribution of this study is to provide the specification and implementation of a grid service for mining fuzzy association rules in both a centralized and distributed manner, this section provides a definition of the problem of mining fuzzy association rules. In traditional association rules mining, all attributes of the database are binary. A popular example of this type of database is the transactional database of a retail market in which each transaction is a set of items bought by a customer. Obviously, in other types of database, we may have continuous attributes, such as the age of people. In this case, it is possible to use a quantitative or fuzzy approach to mine association rules.

Let $T = \{t_1, t_2, \cdots, t_n\}$ be a relational dataset and t_j represents the jth record in T. Let $I = \{i_1, i_2, \cdots, i_m\}$ be the attribute set where i_j denotes a Boolean, categorical or quantitative attribute and $t_j[i_k]$ represents the value of the jth record in attribute i_k. The values of the attributes have to be partitioned into several fuzzy sets for mining fuzzy association rules.

Let V_1 and V_2 be two values of the record in a Boolean attribute, then, two values can be partitioned into fuzzy sets, V_1 and V_2:

$$V_1(x) = \begin{cases} 1, & x = V_1 \\ 0, & x = V_2 \end{cases}, \qquad V_2(x) = \begin{cases} 1, & x = V_2 \\ 0, & x = V_1 \end{cases}.$$

Categorical attributes with fewer values can be partitioned into several fuzzy sets with the same method. Each quantitative attribute is partitioned into several fuzzy sets. For each fuzzy set, a linguistic term is used. Figure 1 shows the defined fuzzy sets on the domain of the age attribute and their assigned linguistic terms. The number of fuzzy sets on the domain of each attribute can be determined by the FCM algorithm [13].

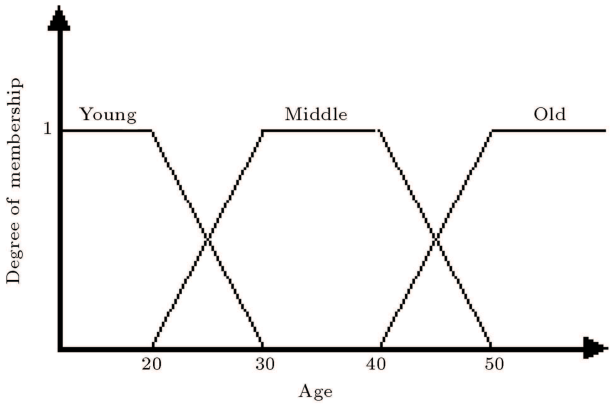

Figure 1. The definition of fuzzy sets and corresponding linguistic terms for age attribute.

In order to mine fuzzy association rules, a new database using the original database, T, is constructed. In this new database, attributes are fuzzy sets; values of the record in attributes are obtained as follows: Let f^k be a fuzzy set on attribute i_k. f^k is an attribute in the new database. The value of the jth record in attribute f^k is $f^k(t_j[i_k])$. $f^k(t_j[i_k])$ is the membership value of $t_j[i_k]$, with respect to fuzzy set f^k. In this new database, because attributes are fuzzy sets, these attributes are named fuzzy attribute sets. Let I still be the fuzzy attribute set, and $t_j(y_k)$ represent the value of the jth record in fuzzy attribute y_k, then, $t_j(y_k)$ falls in [0,1]. Let:

$$X = \{y_1, y_2, \cdots, y_p\} \subset I,$$

$$Y = \{y_{p+1}, y_{p+2}, \cdots, y_{p+q}\} \subset I,$$

$$X \cap Y = \emptyset.$$

An association rule is an implication of the form $X \Rightarrow Y$. Since attributes in X and Y are fuzzy attributes, $X \Rightarrow Y$ is called a fuzzy association rule. Since each fuzzy set is associated with a linguistic term, a fuzzy association rule describes an interesting relationship between two or more linguistic terms. Linguistic terms make generated rules more understandable. Support and confidence are two important measurements that are used with association rules. Their corresponding measurements in fuzzy association rules are called significance and certainty, which are defined as follows [3].

Definition 1

Let fuzzy attribute set $X = \{y_1, y_2, \cdots, y_p\} \subset I$, the fuzzy support of X, also called significance, be defined as follows:

$$\text{Significance } (X) = \frac{\sum_{j=1}^{n} \prod_{m=1}^{p} t_j(y_m)}{n}, \tag{1}$$

where $\sum_{j=1}^{n} \prod_{m=1}^{p} t_j(y_m)$ is called the absolute fuzzy support or absolute significance of X. Fuzzy attribute sets with at least a minimum fuzzy support are called frequent fuzzy attribute sets.

Definition 2

The significance of rule $X \Rightarrow Y$ is defined as follows:

$$\text{Significance} = \frac{\sum_{j=1}^{n} \prod_{m=1}^{p+q} t_j(y_m)}{n}. \quad (2)$$

Definition 3

The certainty or fuzzy confidence of $X \Rightarrow Y$ is defined as follows:

$$\text{Certainty} = \frac{\text{Significance}}{\text{Significance}(X)}. \quad (3)$$

To mine large attribute sets in databases, the well known Apriori algorithm can be adopted [3]. In an adopted version of the Apriori algorithm, not only is the appearance of each candidate attribute checked in each record, but the degree of the support of that candidate is also measured in each record.

OGSA AND WSRF

In order to achieve integration and interoperability in a standard grid environment, the fuzzy association rules mining grid service was designed and implemented using the emerging Web Services Resource Framework (WSRF) [14]. WSRF improves several aspects of web services to make them more adequate for grid applications. WSRF is a family of technical specification concerned with the creation, addressing, inspection and lifetime management of stateful resources. The framework codifies the relationship between web services and stateful resources in terms of implied resource patterns. A stateful resource that participates in the implied resource pattern is termed the WS-Resource. WSRF describes the WS-Resource definition and association with the description of a web service interface, and describes how to make the properties of a WS-Resource accessible through a web service interface.

Initial work on WSRF has been performed by the Globus Alliance [8] and IBM, with the goal of integrating previous work on the so-called Open Grid Service Architecture (OGSA) [15] with new service mechanisms and standards. The OGSA enables communication across geographically dispersed heterogeneous environments. The Globus Alliance recently released Globus Toolkit 4 (GT4) [8], which provides an open source implementation of the WSRF library, incorporating services implemented according to the WSRF specifications. The toolkit components that are most relevant to OGSA are the Grid Resource Allocation and Management (GRAM) protocol, the Meta Directory Service (MDS) and the Grid Security Infrastructure (GSI). These components provide the essential elements of a service-oriented architecture. Combining grid computing with web service technology, including both the physical resource (such as servers and storage devices) and logical resources (such as knowledge and algorithms), are known as stateful resources. This combination and its corresponding specification are known as a WSRF resource framework. We have developed the fuzzy association rules mining grid service by using the Java WSRF provided by a development release of Globus Toolkit 4. We use also the basic OGSA services to perform distributed fuzzy association rules mining in a data grid environment.

DATA MINING GRID SERVICE ARCHITECTURE

As mentioned above, our implementation of fuzzy association rules mining on the grid is based on DMGA [1]. Within this framework, the main functionalities of every stage are deployed by means of grid services. In DMGA, a data mining grid service includes three stages:

i) Pre-processing,
ii) Data mining stage,
iii) Post-processing.

All data mining services use both basic data and generic grid services. Data grid services are service oriented to data management in a grid. One of the best known data grid services is GridFTP [16,17]. The Reliable File Transfer protocol (RFT), Replica Location Service (RLS) and Data Access and Integration Service (DAIS) are other data services. RFT is used for data movement through data grids. RLS is a distributed registry that records locations of data copies. To access relational and XML databases, DAIS is used. Besides data grid services, data mining grid services also use generic and standard grid services. Generic services offer common functionalities in a grid environment. The Grid Resource Allocation and Management (GRAM) and Grid Security Infrastructure (GSI) are examples of generic grid services. The GRAM API allows programs to be started on remote resources, despite local heterogeneity. A layered architecture allows application specific resource brokers and co-allocators (e.g. DUROC) to be defined in terms of GRAM services. The Resource Specification Language (RSL) is used to communicate requirements to submit jobs for GRAM. The GSI provides security across the grid system.

The reasons for using DMGA in this study can be summarized as follows. Most importantly, DMGA uses existing data and generic grid services to perform data mining on the grid. In addition, DMGA introduces new services, which can effectively link with data mining tasks. These new services use the existing data grid services. Moreover, since in most data mining problems in the grid environment several services are involved, DMGA have the capability of a composition of services (i.e., creation of workflows). This ability to create workflows allows several services to be scheduled in an efficient and effective manner.

DMGA introduces the horizontal and vertical composition of the services to perform the whole of the data mining process on the grid. In a horizontal composition, different services available on the heterogeneous platform can be exploited to perform the data mining task. We propose FARM grid services using this type of composition. Another composition of services is the vertical composition. In this composition, the same replicated services are used to enhance the run time and speed up the mining task. We have used this composition to perform distributed fuzzy association rules mining in the grid, where the same replicated services access different data fragments.

SERVICE SPECIFICATION OF FUZZY ASSOCIATION RULES MINING

In this study, among all data mining techniques, fuzzy association rules mining is chosen due to its wide application area and its ability to overcome sharp boundary problems. Additionally, other data mining techniques, such as the classification problem could be solved using this technique [18]. Figure 2 shows the proposed FARM grid service based on Data Mining Grid Service Architecture. In the FARM grid service, the preprocessing step is preparing data for the mining process. In this step, input data is filtered and cleaned. The second step is the main step whereby fuzzy attribute sets are extracted and fuzzy rules are generated. Finally, in the post processing step, the results are transformed to a readable and understandable format for the user.

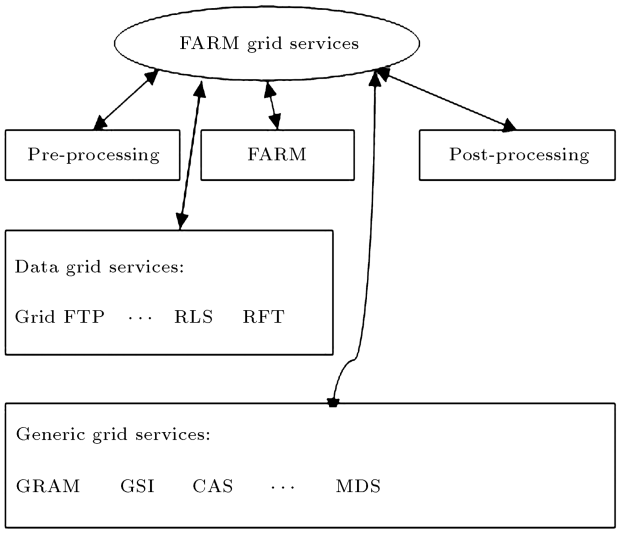

Figure 2. Design of the fuzzy association rules mining grid service based on DMGA architecture.

The data mining grid service uses generic services provided by gird software to submit and execute the job. The FARM service uses GridFTP as a data grid service to submit the input data from client to server and send back the result for the user of fuzzy rules. GridFTP is a high performance data transfer protocol. It is selected because it uses a parallel data stream to exchange data and, hence, it has good performance. Since the FARM grid service is WSRF Compliant, its main algorithm is a WS-Resource, which uses other WS-Resources like data collections and algorithms. The service has three resources, namely minSign, minCert and filename, which save the current state for an invocation to another invocation about minimum significance, minimum certainty and input data, respectively. These resources can be used in the later service invocations.

DISTRIBUTED MINING OF FUZZY ASSOCIATION RULES

Where data are geographically distributed, integrating data from various sites is almost infeasible due to the large size of data, limited bandwidth and privacy concerns. Cheung et al. [19] proposed FDM to mine all association rules in a distributed setting. The FDM aggregates result is produced by a local site to get globally frequent itemsets. In this study, a new version of FDM, namely Fuzzy FDM (FFDM), is developed to mine fuzzy association rules in data grids. Since existing data grids, i.e. GridPP, include dispersed scientific data, this method can mine continuous data efficiently in terms of communication and run time. In the other words, FFDM, like FDM, exchanges only information about frequent itemsets and, thus, the privacy of data is provided since participants of a data grid including scientists or institutes do not tend to exchange their raw data.

To perform distributed fuzzy association rules mining in a data grid, each site uses pre-processing and post-processing grid services as described above. When data are prepared in local sites, deployed FFDM services are started. Similar to FARM, minimum significance and minimum certainty are determined by the user who launches the distributed data mining task across the grid. In each round of FFDM, on every node of the grid environment, steps shown in Algorihtm 1 are executed.

1. Candidate generation
2. Determine local frequent attribute sets
3. For all local frequent attribute sets, determine polling sites
4. For all my candidates send the polling request to other nodes
5. Respond to polling requests of other nodes
6. Determine my globally frequent attribute sets
7. Exchange my result with other sites

Algorithm 1. Main part of FFDM executed on each site.

As shown in Algorithm 1, each site generates candidate attributes based on the previous iteration, then, it scans the local database to determine their significance. Polling sites are determined to assign local frequent attribute sets to the nodes, equally. In this way, each site is responsible for an equal part of all local frequent attribute sets. In steps 3 through 5 of Algorithm 1, the site gathers significant information of its attribute sets and responds to other site requests. In step 6, the site determines which of its attribute sets are frequent and notifies other sites with its found result. In this way, at the end of each round, every site has all the frequent attribute sets of all sites. The process continues until no candidate can be generated. The number of rounds depends on the maximal size of frequent itemsets. At the end of the algorithm, every site has a complete collection of frequent attribute sets. Thus, each site can generate fuzzy association rules, using minimum certainty and a complete set of fuzzy attribute sets. The final step is to represent generated rules using a post-processing grid service.

Since, at the beginning and end of the mining task, all sites request pre-processing and post-processing grid services, we can replicate these services across the data grid environment to enhance response time.

IMPLEMENTATION AND SERVICE DEPLOYMENT

The implementation of the grid data mining service is based on the Globus Toolkit 4. The Globus toolkit is a community-based, open architecture, open source set of services and software libraries that support gird and grid applications [8]. The toolkit addresses issues of security, information discovery, resource management, data management, communication and portability. The Globus toolkit is used in major grid projects worldwide. It is based on the OGSA in which a grid provides an extensible set of services that virtual organizations can aggregate in various ways [20]. In the Globus Toolkit, a grid Service is based on the web Service but with some improvements introduced by WSRF specifications: stateful and potentially transient services, data services, lifetime management, notification, service groups and portType (interface) extension.

In the implementation, a service for mining fuzzy association rules is written in java and is WSRF-compliant. The implemented service containing class files, WSDL and WSDD etc. is archived using the globus build service, which is a script using ANT to generate GAR files. A GAR file is a grid archive file containing all required files to deploy a service. After generating a GAR file, it is deployed using the globus-deploy-gar command provided in the Globus Toolkit.

For implementing FFDM, communication between nodes is needed. We have used the Message Passing Interface extended with a grid service (MPICH-G2) to implement the FFDM algorithm. MPICH-G2 is a grid-enabled message passing interface which has primitives to exchange data between grid nodes. We use these primitives to communicate candidate attribute sets between nodes participating to execute the FFDM algorithm. This library also uses the services provided by the Globus Toolkit for authentication, authorization, resource allocation, executable staging and I/O, as well as for process creation for monitoring and control. The MPICH-G2 uses Grid Resource Allocation and Management (GRAM) to co-schedule subtasks across multiple computers. GRAM provides a web service interface for initiating, monitoring and managing the execution of arbitrary computations on remote computers. MPICH-G2 automatically converts data in messages sent between machines of different architectures, and supports multiprotocol communication by automatically selecting TCP for inter-machine message passing.

EXPERIMENTAL EVALUATION

In order to evaluate the FARM grid service, we setup the Globus Toolkit 4 on 3 nodes, deploy the fuzzy association rules mining grid service on one of the nodes and perform some experimental evaluations. The site which provides the fuzzy association rules mining grid service is also a GridFTP server that receives input data and sends back the result. If the dataset is available at the site which provides the service, the service mines the fuzzy association rules and sends the result to the client. The client also can use a third party transfer to move data residing in a GridFTP server to the fuzzy data mining server. The time for completing the service is measured where it is invoked by a remote client. This time includes:

(i) Server time: Time spent by the server for performing the service (running the data mining algorithm).

(ii) Client time: Total time for obtaining the results

in the client interface. This time includes the server time, the time for connecting to the server and the time for transferring the results.

(iii) GridFTP time: Time spent sending input files and transferring the results.

The overhead of the invocation of the grid service is the addition of the service connection duration and the GridFTP time. In the experimental evaluation, an Intel 3.01 GHz, a processor having 512 MB of main memory is used to run the service. Obviously, in a real grid environment, it is possible to have this grid service running on different nodes having different specifications. Therefore, depending on the technical characteristics of the nodes in which the data mining process is performed, its performance is different. This feature makes it possible to decide, according to different criteria, which data mining gird service is to be used. This task could be carried out by a broker service. In this case, if we have several equal or different data mining grid services at several locations, we can use a trading protocol for deciding, at run time, which one is the best fit for the client requirements.

To evaluate the performance of the grid fuzzy association rules mining service, a large data set is needed. A program is written which generates datasets having different sizes, using the input parameters of the user including the number of records and fields. For this experiment, a data set containing 50000 records, having three continuous fields, is generated.

Experimental evaluations have been performed several times. The results are shown in Figure 3. The figure show the run time of the grid service including server time, client time and GridFTP time. The X axis shows the experiment number and the Y axis shows the duration for client time, server time and GridFTP time. All experiments have been performed at the normal workload of the server. The average client time of all tests is 24950 ms. The overhead in Figure 3 includes the difference between client and server time, i.e. service connection and GridFTP time. As shown in the figure, we have a small overhead. If the data is replicated in the server in the following invocation of the service for the same data input, the GridFTP time for sending input data is omitted from the overhead. The saved results on the server are sent to the client, if a repeated request including the same data input, the same minimum significance and certainty is issued by the client. In this case, the server and GridFTP time for sending data input is omitted and client time includes only the service connection time and GridFTP time for sending the results. Since implementation of the fuzzy association rules mining grid service has the advantage of using the service in the whole grid environment including different virtual organizations; this overhead is very small in comparison to its advantages. On the other hand, this service can be used by other generic grid services, like GRAM, for job scheduling, which can enhance the performance of the whole grid system.

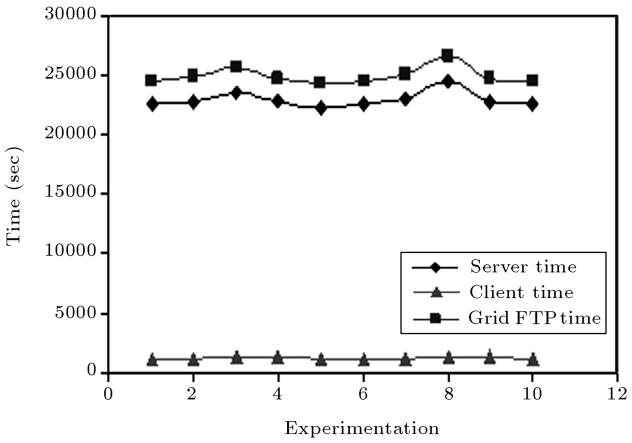

Figure 3. Performance evaluation results of fuzzy association rules mining grid service.

Both grid technology and distributed mining algorithms reduce computational time and increase the speed of the application. The experimental results show that a distributed version of the fuzzy association rules mining algorithm outperforms an ordinary centralized algorithm. Its performance is scalable in terms of database size and number of nodes. In FFDM mining, fuzzy association rules do not need the GridFTP time of the FARM grid service to integrate data in a single site. The FFDAM uses the distributed storage of the data grid. The FFDM method is implemented using the Visual C++ 6 and MPICH-G2 library. MPICH-G2, a grid-enabled message passing interface, has the same API of MPICH, but is used to execute distributed and parallel methods on the grid environment. Thus, the program written using MPICH can be executed on the grid environment using MPICH-G2. To empirically evaluate the implemented distributed data mining approach, the abalone dataset from the UCI machine learning repository [21] is used. This dataset contains both categorical and numerical attributes. To evaluate the performance of the FFDM, using a large dataset, the abalone dataset is copied 32 times and a large dataset, including 133,664 transactions, is generated using the copies. We perform experiments using different numbers of computers on the grid environment using this dataset. Specifications of these computers are the same as the first experiment. Since the number of participants including scientists and institutes, involved in scientific data grids is increasing, the proposed FFDM must be scalable to handle very large data collections that are geographically distributed. On the other hand, an important positive characteristic of any distributed

method is scalability. Thus, the scale-up of the method is evaluated in this experiment where the result reveals how the algorithm handles large datasets when more computers are available in a data grid. In the experiment, as a computer is inserted in the grid, a copy of the above generated dataset is resided on the new node. In this way, as the number of computers is increased, the size of the problem is also increased. To submit the distributed job to the Globus, an RSL is needed to describe the requirement for the GRAM service. The GRAM uses this specification to execute the job. Figure 4 shows some information presented in the RSL we have written, to perform the experiment when 6 computers are used.

This RSL describes the job for the mpirun command of MPICH-G2 for our experimentation. As you can see in Figure 4, the RSL describes the job using some attribute-value pairs as follows. The PBS is used as a local scheduler that is running on a Server.eng4.shirazu.ac.ir computer. Additionally, the number of computers used to execute the job (count variable), the type of job, label of job, the co-allocator (DUROC is used) directory of executable and other required information to execute the job on the grid environment.

The count attributes of the above RSL is set from 2 to 8 to determine the number of computers of each experiment. Figure 5 shows that the FFDM has an almost fixed run time, when the number of computers (computational resources) and the size of problems are increased simultaneously. In the experiment, attributes are partitioned into three fuzzy sets. Let minimum significance (minSign) be 0.01and let minimum certainty (minCert) be 0.1.

To see how the response time of the FFDM algorithm is reduced with respect to the number of nodes in the grid, we perform speed up experiments where we keep the dataset constant and vary the number of

```
+
(&(resourceManagerContact=
    "Server.eng4.shirazu.ac.ir/jobmanager-pbs")
(count=6)
(jobtype=mpi)
(label="subjob 0")
(environment=(GLOBUS_DUROC_SUBJOB_INDEX 0)
        (LD_LIBRARY_PATH /usr/local/globus-2.4.3/lib/))
(directory="/home/gird/globusTest/MPICH-G2")
(executable="/home/gird/globusTest/MPICH-G2/ffdm"))
```

Figure 4. Some information presented in the RSL file used to evaluate the FFDM.

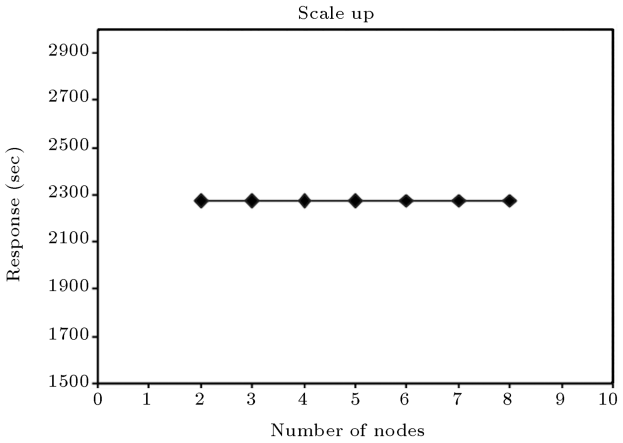

Figure 5. The scale up experiment.

computers. In each speed up experiment, the data set is partitioned horizontally, w.r.t the number of nodes. For example, the first speed up experiment is performed using two computers, each of them having half the generated dataset. The remaining parameters are set as in the scale-up experiment. Figure 6 shows the performance results of the speed up experiment and Table 1 shows the execution time for some of the grid configurations of this experiment.

As shown in Figure 6, with the increase in the number of computers, the speed up improves. This is due to the fair distribution of I/O costs and computations among all the grid nodes. Table 1 also shows that

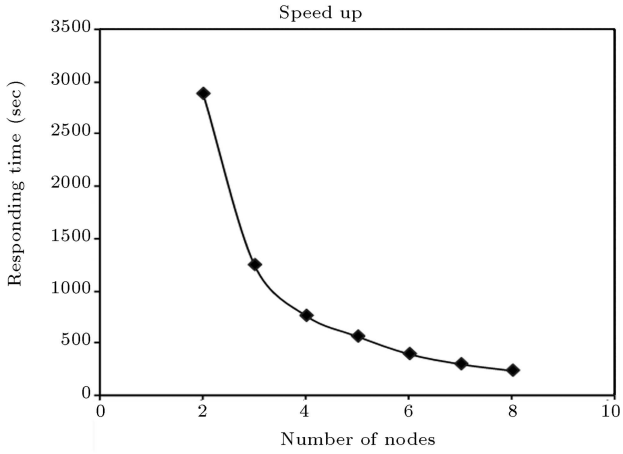

Figure 6. The speed up experiment.

Table 1. Execution time on different grid configuration.

No. of Nodes	Data Size per Node (MB)	Execution Time (sec.)
2	2.86	2894
4	1.43	774
6	0.953	409
8	0.715	250

better response time is achieved with an increase in the number of nodes and the distribution of I/O costs. Both grid technology and the distributed algorithm reduce computational time and increase the speed of the fuzzy mining process.

CONCLUSION

In this study, a fuzzy data mining grid service is designed and implemented based on recently proposed Data Mining Grid Architecture. The main aim of this fuzzy grid service is to extract fuzzy association rules in data grids from the user specified dataset locations. The fuzzy association rules mining grid service has the advantage of accessibility from different virtual organizations and can be used by other grid services like job scheduling. Our implementation is based on web services and is WSRF compliant. The performance evaluations show the acceptable performance of the grid service with little overhead of service invocation and the input-result data transfer. Since in the data grid environment, data collections are geographically distributed, to utilize the computational resources of the grid and to avoid moving raw data to the central grid node (which is infeasible in some situations due to large amounts of data and privacy concerns), a distributed algorithm, namely FFDM, is proposed and implemented to mine fuzzy association rules in a distributed manner. Experimental evaluations of FFDM on real-life datasets show that the approach scales well when the number of nodes and the size of the problems are increased simultaneously. Additionally, the FFDM has a superliner speed up as the number of grid nodes is increased. It is important to note that FFDM is not an enhancement of the FARM grid service. They can be used in different situations and scenarios. Where the aim is mining fuzzy ARs in remote site, the FARM grid service can be used. Where data collections are dispersed across data grid nodes and moving data is infeasible or very time consuming, FFDM can be used to mine fuzzy ARs in a distributed fashion.

In the process of distributed data mining, synchronization and communication between grid nodes are necessary. Since the data grid environment is a large scale geographically distributed infrastructure in which many computational nodes are connected through various networks with different bandwidths and latency; these two problems are the main bottlenecks for high performance computing over the grid. In order to achieve a better performance, future work should address the problems of distributed data mining with little communication and synchronization. Another open issue is about mining classification rules. The classification problem can be solved using fuzzy association rules [18]. By proposing methods for fuzzy association rules mining in the grid, it is possible to mine classification rules as another function of the data mining, using data grid services of fuzzy association rules on the data grid environment.

REFERENCES

1. Perez, M., Sanchez, A., Robles, V., Herrero, P. and Pena, J.M. "Design and implementation of a data mining grid-aware architecture", *Future Generation Computer Systems*, **23**, pp. 42-47 (2007).
2. Agrawal, R. Imielinski, T. and Swami, A. "Mining association rules between sets of items in large database", *Int. Conf. of ACM SIGMOD*, Washington, DC, pp. 207-206 (1993).
3. Chan, M.K., Ada, F. and Man, H.W. "Mining fuzzy association rules in database", *ACM Sixth Int. Conf. on Information and Knowledge Management*, Las Vegas, Nevada, pp. 10-14 (1997).
4. Agrawal, R. and Srikant, R. "Fast algorithms for mining association rules", *Int. Conf. of VLDB*, Santiago de Chile, pp. 487-499 (Sept. 1994).
5. Aflori, C. and Craus, M. "Grid implementation of the Apriori algorithm", *J. of Advanced Engineering Software*, **38**(5), pp. 295-300 (May 2007).
6. *The European DataGrid Project* http://www.edg.org.
7. *The GridPP Project*, UK Computing for Particle Physics. http://www.gridpp.ac.uk/
8. The Globus Alliance: Globus Toolkit 4. http://www.globus.org/toolkit.
9. Natarajan, R., Sion, R. and Phan, T. "A grid-based approach for enterprise-scale data mining", *J. of Future Generation Computer Systems*, **23**(1), pp. 48-54 (January 2007).
10. Cannataro, M. and Talia, D. "The knowledge grid", *Communication of the ACM*, **46**(1) pp. 89-93 (2003).
11. Giannadakis, N., Rowe A., Ghanem, M. and Guo Y. "InfoGrid: providing information integration for knowledge discovery", *J. of Information Sciences*, **155**(3,4), pp. 199-226 (October 2003).
12. Hung, J., Jin, H., Xie, X. and Zhang, Q. "An approach to grid scheduling optimization based on fuzzy association rule mining", *First Int. Conf. on e-Science and Grid Computing*, pp. 189-195 (2005).
13. Hathaway, R.J., Davenport, J.W. and Bezdek, J.C. "Relational dual of the c-means algorithms", *J. of Pattern Recognition*, **22**(2), pp. 205-212 (1989).
14. Czajkowski, K. et al. "The WS-resource framework version 1.0", http://www.106.ibm.com/developerworks/library/ws-resource/ws-wsrf.pdf
15. Foster, I. et al. "The physiology of the grid: an open grid services architecture for distributed system integration", *Tech. Report, Globus Project* (2002).
16. Allcock, W., Bester, J., Bresnahan, J., Chervenak, A., Liming, L. and Tuecke, S. "GridFTP: Protocol extensions to FTP for the Grid", *Global Grid Forum Draft* (2001).

17. Aloisio, G., Cafaro, M. and Epicoco, I. "Early experiences with the gridftp protocol using the grb-gsiftp library", *J. Future Generation Computer Systems*, **18**(8), pp. 1053-1059 (2002).

18. Hu, Y.C., Chen, R.S. and Tzeng, G.H. "Mining fuzzy association rules for classification problems", *J. of Computer & Industrial Engineering*, pp. 735-750 (2002).

19. Cheung, D.W. et al. "A fast distributed algorithm for mining association rules", *Int. Conf. of Parallel and Distributed Information Systems*, IEEE CS Press, pp. 31-42 (1996).

20. Foster, I., Kesselman, C., Nick, J. and Tuecke, S. "The physiology of the grid", in *Grid Computing: Making the Global Infrastructure a Reality*, Berman, F., Fox, G.A. and Hey, A., Eds.,Wiley, pp. 217-249 (2003).

21. UCI Machine Learning Repository: http://archive.ics.uci.edu/ml/

BIOGRAPHIES

Mahmood Deypir is currently a PhD student at the Computer Science and Engineering Department in Shiraz University, Iran. He received his MS degree from Shiraz University in 2006 and his BS degree from Shahid Sattari University in Tehran, Iran, in 2003. His research interests include Data Mining and Distributed Computing.

Gholamhossein Dastghaibyfard is an Assistant Professor in the Department of Computer Science and Engineering in the College of Engineering at Shiraz University, Iran. In 1977, he received his BS degree from RCD International School of Insurance in Tehran. His MS and PhD degrees were both received from the University of Oklahoma, USA, in 1979 and 1991, respectively. His research interests are Parallel and Grid Computing.

Mohammad Hadi Sadreddini received his BS degree in Computer Science in 1985, his MS in Information Technology in 1986 and a PhD degree in Distributed Information Systems in 1991 from Ulster University in the UK. He has been working in the department of Computer Science and Engineering at Shiraz University, in Iran, since 1993. His research interests include Association Rules Mining, Bioinformatics, and Distributed Systems.