Distributed and Parallel Data Mining on the Grid

Tianchao Li

Institut für Informatik
Technische Universität München
Boltzmannstr. 3
D-85748 Garching bei München, Germany
tianchao.li@cs.tum.edu

Toni Bollinger

IBM Development Laboratory Böblingen
Schönaicher Str. 220
D-71032 Böblingen, Germany
toni.bollinger@de.ibm.com

Abstract: This paper presents the initial design and implementation of a Grid-based distributed and parallel data mining system. The Grid system, namely the Business Intelligence Grid or BIGrid, is based on heterogeneous Grid server configurations and service-oriented Grid architecture. The system follows a layered design, whose infrastructure is divided into three tiers in general - Grid tier, a service tier and a client/portal tier. Issues of design and implementation, including brokering, task scheduling, adaptive mining script preparation and parallelization are discussed. The design and implementation of BIGrid help identify the specific requirements of applying Grid-based data mining in business realm, thus pave way for future design and implementation of a real generic Grid-based data mining system.

1 Introduction

Grid computing marks a major milestone in the evolution of distributed computing. With a computing paradigm similar to an electric power grid, Grid computing is believed to be the solution for many major problems that are common in today's distributed and heterogeneous computational environments of many organizations and enterprises.

Data mining, which targets the goal of retrieving information automatically from large data sets, is one of the most important business intelligence technologies. Because of its high computational intensiveness and data intensiveness, data mining serves a good field of application for Grid technology.

The idea of data mining on the Grid is not new, but it has become a hot research topic only recently. The number of research efforts up to now is still quite limited (for a short summary see [CT03]). Many of the research efforts, such as NASA's Information Power Grid [HN00], TeraGrid [Co03] and Discovery Net [Ću02] are either utilizing non-standard data mining techniques, or restricted to a special domain in the scientific realm. The Knowledge Grid [CT03], which is an attempt towards domain-independent environment, is still restricted to experiences from relevant researches in the scientific realm.

The design of Grid-based data mining system can hardly be truly generic without taking business realm into considerations, as it will reveal the importance of some specific requirements that are not so evident in the scientific realm.

First, security is crucial for distributed data mining in enterprises. In many situations, people only want to share the data patters rather than the data itself. This, together with the fact that data mining is highly data intensive and often suffers from the bottleneck of network, leads to the fact that data shipping is not applicable for security and efficiency.

Second, enterprises tend to use standard data mining techniques from commercial data mining software rather than domain specific knowledge extraction techniques from self-implemented software. Thus, Grid-based data mining system for enterprises should be able to cope with the heterogeneity of not only computers, operating systems, networks, but also that from the different sources of data and different data mining software. The complexity and differences in installation and configuration procedures of those software also inadequate program shipping, especially in cases that data mining programs are tightly coupled with the database.

Additionally, in the highly competitive business environment, enterprises tend to be more flexible and dynamic. As a result, data mining systems in enterprises face stronger demand on the capability of processing data that are dynamically organized.

The Business Intelligence Grid (BIGrid)[1] [Li03] is constructed as a test-bed for researches of Grid-based data mining. Based on case studies of real application scenarios in commercial realm, the final destination of the research targets a generic Grid-based data mining system for both scientific and business realms. This paper presents our initial design and implementation of this Grid-based data mining system.

The reset of the paper is organized as follows. Section 2 describes a typical real-world application scenario of data mining in the business realm and the layered infrastructure of BIGrid. Section 3 discusses different issues of design and implementation of BIGrid, including information providers and brokers, task scheduling, adaptive mining script preparation, XML-based message formats, and the "embarrassingly parallel" paradigm. Finally, section 4 draws conclusion and discusses future work.

---

[1] BIGrid has been exhibited on IBM/Forbes Executive Forum, October 14-16, 2003, San Diego, CA

2 Business Intelligence Grid

2.1 The Typical Application Scenario

A typical application scenario of Grid-based data mining in real-life enterprises is presented in Figure 1, where there is a franchise supermarket formed by headquarter, regional branches and distributed member stores.

Each store collects transaction data by scanning bar codes at the till when customers buy products. Most stores rely on regional branch to store their transaction data, while some stores have local databases. For the sake of safety or convenience, the transaction data for certain stores are optionally replicated in several different data storages. Each regional branch or member stores have their own choice of data mining software, and deploys the software locally. The headquarter wants to apply data mining to analyze the transactional data of selected member stores.

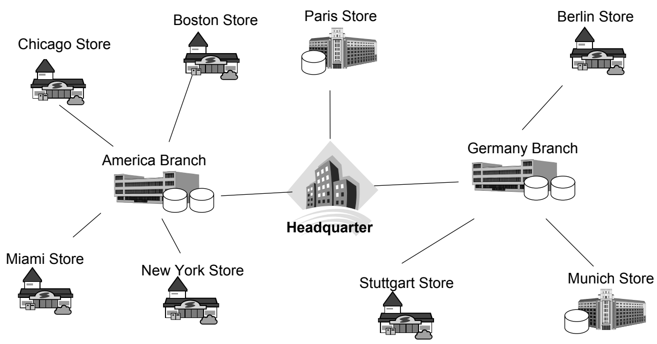

Figure 1: Franchise Supermarket Scenario

The franchise supermarket described above, though specific, is typical for Grid-based data mining in the business realm. Another example situation that follow similar scenario is the cooperation among several financial organizations. They need to share data patterns relevant to the data mining task, but they do not want to share the data since it is sensitive.

2.2 Business Intelligence Grid Infrastructure

BIGrid follows a layered design (Figure 2), where the infrastructure is divided into three tiers in general – Grid tier, service tier, and client/portal tier.

2.2.1 Grid Tier

The underlying Grid tier is composed of a set of heterogeneous compute nodes. The business data is partitioned and deployed on each server as databases. Software enabling data mining on local partial of data are installed and configured on each Grid node. Grid middleware, currently Globus Toolkit v2.2 [GT], enables a secure, coordinated and dynamic resource sharing among the Grid nodes. Custom information providers are implemented and deployed to provide additional information about configurations and data content.

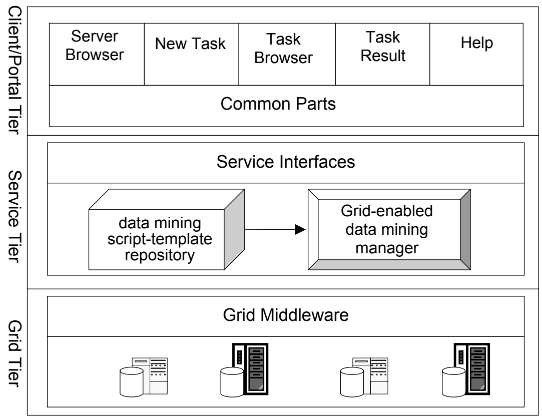

Figure 2: Business Intelligence Grid Infrastructure

2.2.2 Service Tier

The service tier implements a Grid-enabled data mining system and provides standard Web service interfaces. The Grid-enabled data mining system consists a data mining script template repository and a Grid-enabled data mining manager.

The data mining script template repository stores the scripts and script templates for all kinds of data mining software deployed on the servers. The script templates are transformed into actual data mining scripts by XSLT. The extensible script-template repository and the process of adaptive script template transformation will be detailed in section 3.3.

The Grid-enabled data mining manager controls the process of distributed data mining on the Grid servers until finally retrieving the result of data mining. The initial implementation of the data mining manager utilizing an “embarrassingly parallel” paradigm will be discussed in section 3.5.

2.2.3 Client/Portal Tier

The Grid-enabled data mining services can be accessed with custom applications that are either programmed or generated with tools according to the Web service interface definitions.

For casual users that do not intend to have custom implementation of service clients, a portal is implemented for BIGrid to provide easy access. The portal provides a user interface for the Grid system where the user submits new data mining tasks, browses the statuses of tasks, and explores the visualization of data mining results. The portal is implemented as a J2EE compliant application, so that it is accessible by the end user simply through a Web browser. The portal uses a database as means of data persistence for all information about each mining task and the resulting model.

The communication between Web browsers and application server, and communication between application server and Web services are accomplished through HTTPS to guarantee security additionally.

3 Issues of Design and Implementation

3.1 Information Providers and Brokers

On each Grid server, custom information providers are implemented and deployed to complement the standard information providers of Globus MDS. The information provided by those custom information providers includes properties of site-local data mining software, contents and structures of local data partitions, etc.

A custom Grid broker component is implemented for BIGrid, which queries MDS and provides necessary information for the Grid-enabled data mining system. The information provided by this broker includes not only relevant information provided by the standard Grid information providers, but also those provided by the custom information providers.

The information provided by the custom information providers and in turn by the brokers is the basis of dynamic resource discovery, task scheduling and adaptive mining script preparation, which will be detailed in the following sections.

3.2 Task Scheduling

A simple data mining task scheduler is implemented for BIGrid, which is based on a “greedy” and “single-minded” algorithm that always tries to minimize the execution time for the current data mining task.

Information provided by the broker provides the basis of task scheduling. The content of local partition of data to be mined is crucial for the scheduler to determine all possible combinations of target servers for a given list of target contents. Information about speed, processor usage, and the size of local partition of business data is important for the scheduler to determine the fastest scheduling among several possible combinations of target servers.

The scheduler constructs internally a table of all possible combinations of servers and target contents that match the user's request and compares the estimated execution time. The item with the shortest execution time will be chosen by the scheduler.

3.3 Adaptive Mining Script Preparation

It is common practice for Grid applications to decide on program shipping versus data shipping, i.e. to move program to the place of data or to move data to the place of program. However, as is discussed in section 1, neither of these two approaches is suitable for the application scenario of Grid-based data mining in business realm.

In BIGrid, we take a new approach, or more precisely, an amended program shipping approach. In this approach, neither the data to be mined nor the data mining program but necessary scripts required by the corresponding data mining software are shipped to the place of data and mining software.

The scripts and script templates are structurally stored in an extensible mining script template repository that has a predefined directory structure. Scripts and script templates for one specific data mining software are stored in a subdirectory under the repository. The repository is configurable and extensible, with XML documents storing all the information about the supported data mining program and the scripts or script templates for a specific data mining program.

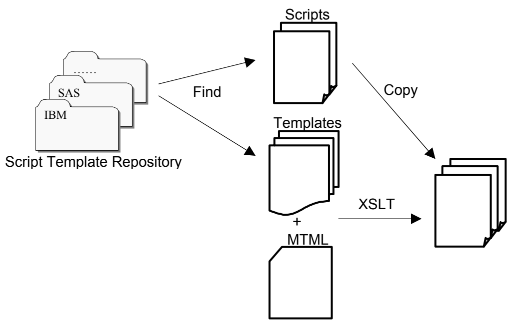

Figure 3: Adaptive Mining Script Preparation

The extensible mining script repository provides the basis for adaptive mining preparation, as is illustrated in Figure 3. The process starts from the selection of appropriate mining scripts and script templates. This is done by a matching of data mining software name and version between the configuration of the mining script template repository and the information for the target Grid server from the broker. Once appropriate mining scripts and script templates for the target system is selected, they are either copied or transformed into the actual scripts. The script templates in the repository are XSL Styles Sheets, which are XML documents following the definition of XSLT. XSLT provides the mechanism to generate document from data (a well-formed XML document) and templates (XSL Style Sheets) with XSLT processor.

During initial stages of the Grid job submission, the scripts are staged to the target Grid server with the help of Globus GASS.

3.4 XML-based Message Formats

For the sake of compatibility with different Web/Grid service servers and programming languages, any objects that contain complex information, such as the result of data mining, are expressed in XML-based message formats. These formats are used both between service clients and Web/Grid services and between Grid client and Grid servers.

Whenever available, standard formats are utilized. PMML (Predictive Model Markup Language, ref. [PMML]), a XML-based industry standard language for defining predictive data models, is used to describe the intermediate and final data mining result. DSML (Directory Services Markup Language, ref. [DSML]), a XML-based standard markup language for describing and manipulating directory information, is used to represent the information from MDS.

Although PMML provides a standard format for describing the result of data mining, it is not suitable for describing the parameter for a data mining task. For this, an XML-based document format, the Mining Task Markup Language ("MTML") is applied, with a custom XML Schema definition.

3.5 "Embarrassingly Parallel" Paradigm

The current initial implementation of BIGrid implements association rule mining, one of the major standard data mining techniques. Based on algorithms like Apriori algorithm [AS94], this mining technique is applicable for "embarrassingly parallel" paradigm, which simply means that a task can be divided into small independent jobs.

The general process of a Grid-based association rule mining is illustrated in Figure 4. The mining manager starts a mining task with scheduling, when the scheduler matches the mining task to available servers in an optimal way (ref. section 3.2). The scripts for the mining task are prepared separately for each destination server, adapted to the specific data mining software on that server (ref. section 3.3). These scripts inherit data mining parameters from the mining request, especially the requirement of minimal support, minimal confidence, and maximal rule length. Grid jobs are then submitted to the Grid servers for performing data mining on local partition of data. The contents of Grid job results, i.e. the result of data mining on Grid servers, are read into memory and are merged together.

The result returned by data mining software usually only contain association rules that satisfy the requirements specified in mining parameters – minimal support, minimal confidence, maximal rule length etc. This fact might lead to an inaccuracy in the result of merging: It may happen that even if an association rule satisfies the conditions for minimal support and confidence on the whole dataset, these conditions are not fulfilled on one server. In this case, this rule would not be in the result set of this server and the support such that we will not be able to get the corresponding confidence and support values for this rule on this server. As a result, the confidence and support for this association rule after merging will not be exact, even though it can be quite close to the exact value.

Unlike "rough" mode, which accepts the inexact result of simple merging, the "exact" mode tries to make corrections. Rules that possibly suffer from the absence of information are detected. Scripts for makeup mining jobs, each with very small sets of items, are prepared while removing the requirements for minimal support and confidence. The makeup mining jobs are submitted, and the results are collected. A makeup merge is performed to update the association rule parameters from first merge.

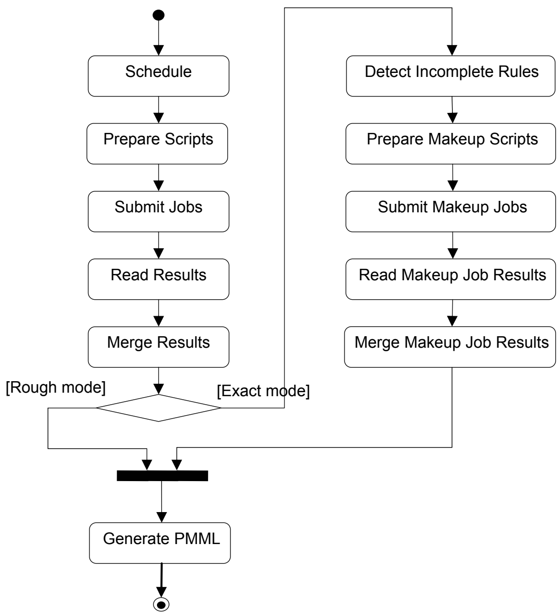

Figure 4: Grid-based Data Mining with "Embarrassingly Parallel" Paradigm

4 Conclusions and Future Work

The design of a generic Grid-based data mining system can hardly be successful without taking both scientific realm and business realm into considerations. Case studies of real application scenarios in business realm help reveal specific requirements that are not evident in scientific realm.

BIGrid is constructed as a test-bed for researches of Grid-based data mining. Our initial design and implementation of a Grid-based data mining system implements one of the major data mining technologies – association rule mining, following "embarrassingly parallel" paradigm. The infrastructure of the system follows a layered design and can be divided into three tiers in general – Grid tier, service tier and client/portal tier. The achievement of desired features and properties – heterogeneity support, dynamic organization, scalability, security, configurability, extensibility, increased availability, ease-of-use etc – proves the feasibility of this approach.

The ultimate destination of our research targets a generic Grid-based data mining system for both scientific and business realms. Future work will regard the full implementation of other data mining techniques and examination of the current implementation against more case studies of real application scenarios in both commercial realm and scientific realm, as well as migration of the current implementation to Globus Toolkit v3.

When implementing for other data mining techniques, it is important to examine the concrete algorithm for the necessity of inter-process communication. The association rules mining technique is applicable for an "embarrassingly parallel" paradigm, where a job can be divided into small independent sub-jobs. However, for many other data mining techniques and algorithms, a find-grained parallelism must be employed. That is, mechanism of inter-process communication (for example, the Grid-enabled implementation of MPI v1.1 standard – MPICH-G2 [KT03]) must be provided so that the sub-jobs can exchange data. Usually a modification of the data mining software is necessary in this case. In the rough sense, regression should be possible for a similar "embarrassingly parallel" paradigm, while the other data mining techniques are likely to be obliged to have inter-process communications.

The primary impacts by migrating to Globus Toolkit v3 come with Grid service [FK02]. Instead of a stateless, persistent Web service, a Grid service is stateful and transient, and can be referenced with a handle (GSH). In the current implementation presented in this paper, the Grid portal has a long running process for each Web service invocation to keep track of it. With Grid service, this long running process can be broken into several very short processes. Each of those processes is responsible for either invoking the service, for querying the status or for retrieving the result of the data mining. Thus, the implementation of the Grid portal can be greatly simplified.

Acknowledgements

We would like to express our gratitude to Nikolaus Breuer and Hans-Dieter Wehle from IBM Development Laboratory Böblingen for all the support during the development of BIGrid. Thanks to Frank Kirschner for solving many Grid configuration problems.

References

[AS94] Agrawal, R.; Srikant, A.: Fast algorithms for mining association rules, Proc. VLDB'94, pp487-499, 1994

[Co03] Conover, H. et.al.: Data Mining on the TeraGrid, Poster Presentation, Supercomputing Conference 2003, Phoenix, AZ, Nov. 15, 2003

[CT03] Cannataro, M; Talia, D.: The Knowledge Grid, Communications of the ACM, Vol. 46, No. 1, pp89-93, Jan. 2003

[Ću02] Ćurčin, V. et.al.: Discovery Net: Towards a Grid of Knowledge Discovery, Proc. 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Edmonton, Alberta, Canada, pp658-663, 2002

[DSML] Michael, H: Introduction to Directory Services Markup Language (DSML), http://www.wdvl.com/Authoring/Languages/XML/DSML/Intro/

[FK02] Foster, I.; Kesselman, C.; Nick, J.; Tuecke, S.: The Physiology of the Grid: An Open Grid Services Architecture for Distributed Systems Integration, Open Grid Service Infrastructure WG, Global Grid Forum, June 22, 2002

[GT] Globus Toolkit, http://www-unix.globus.org/toolkit/

[HN00] Hinke, T.; Novotny, J.: Data Mining on NASA's Information Power Grid, Proc. 9th IEEE International Symposium on High Performance Distributed Computing, Pittsburgh, Pennsylvania, Aug. 1-4, 2000

[KT03] Karonis, N.; Toonen, B.; Foster, I.: MPICH-G2: A Grid-Enabled Implementation of the Message Passing Interface, Journal of Parallel and Distributed Computing (JPDC), Vol. 63, No. 5, pp551-563, May 2003

[Li03] Li, T.: A Grid-based Business Intelligence Infrastructure – Concept, Design and Prototype Implementation of Business Intelligence Grid, Master Thesis, Technische Universität München, 2003

[PMML] Data Mining Group, PMML 2.0 Specification, http://www.dmg.org/v2-0/GeneralStructure.html