Distributed Association Rule Mining

Mafruz Zaman Ashrafi
Monash University, Australia

David Taniar
Monash University, Australia

Kate A. Smith
Monash University, Australia

INTRODUCTION

Data mining is an iterative and interactive process that explores and analyzes voluminous digital data to discover valid, novel, and meaningful patterns (Mohammed, 1999). Since digital data may have terabytes of records, data mining techniques aim to find patterns using computationally efficient techniques. It is related to a subarea of statistics called exploratory data analysis. During the past decade, data mining techniques have been used in various business, government, and scientific applications.

Association rule mining (Agrawal, Imielinsky & Sawmi, 1993) is one of the most studied fields in the data-mining domain. The key strength of association mining is completeness. It has the ability to discover all associations within a given dataset. Two important constraints of association rule mining are support and confidence (Agrawal & Srikant, 1994). These constraints are used to measure the interestingness of a rule. The motivation of association rule mining comes from market-basket analysis that aims to discover customer purchase behavior. However, its applications are not limited only to market-basket analysis; rather, they are used in other applications, such as network intrusion detection, credit card fraud detection, and so forth.

The widespread use of computers and the advances in network technologies have enabled modern organizations to distribute their computing resources among different sites. Various business applications used by such organizations normally store their day-to-day data in each respective site. Data of such organizations increases in size everyday. Discovering useful patterns from such organizations using a centralized data mining approach is not always feasible, because merging datasets from different sites into a centralized site incurs large network communication costs (Ashrafi, David & Kate, 2004). Furthermore, data from these organizations are not only distributed over various locations, but are also fragmented vertically. Therefore, it becomes more difficult, if not impossible, to combine them in a central location. Therefore, Distributed Association Rule Mining (DARM) emerges as an active subarea of data-mining research.

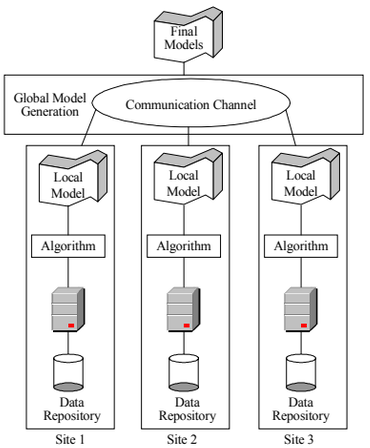
Consider the following example. A supermarket may have several data centers spread over various regions across the country. Each of these centers may have gigabytes of data. In order to find customer purchase behavior from these datasets, one can employ an association rule mining algorithm in one of the regional data centers. However, employing a mining algorithm to a particular data center will not allow us to obtain all the potential patterns, because customer purchase patterns of one region will vary from the others. So, in order to achieve all potential patterns, we rely on some kind of distributed association rule mining algorithm, which can incorporate all data centers.

Distributed systems, by nature, require communication. Since distributed association rule mining algorithms generate rules from different datasets spread over various geographical sites, they consequently require external communications in every step of the process (Ashrafi, David & Kate, 2004; Assaf & Ron, 2002; Cheung, Ng, Fu & Fu, 1996). As a result, DARM algorithms aim to reduce communication costs in such a way that the total cost of generating global association rules must be less than the cost of combining datasets of all participating sites into a centralized site.

BACKGROUND

DARM aims to discover rules from different datasets that are distributed across multiple sites and interconnected by a communication network. It tries to avoid the communication cost of combining datasets into a centralized site, which requires large amounts of network communication. It offers a new technique to discover knowledge or patterns from such loosely coupled distributed datasets and produces global rule models by using minimal network communication. Figure 1 illustrates a typical DARM framework. It shows three par-

Figure 1. A distributed data mining framework



ticipating sites, where each site generates local models from its respective data repository and exchanges local models with other sites in order to generate global models.

Typically, rules generated by DARM algorithms are considered interesting, if they satisfy both minimum global support and confidence threshold. To find interesting global rules, DARM generally has two distinct tasks: (i) global support count, and (ii) global rules generation.

Let D be a virtual transaction dataset comprised of $D_1, D_2, D_3 \ldots \ldots D_m$ geographically distributed datasets; let n be the number of items and I be the set of items such that $I = \{a_1, a_2, a_3, \ldots\ldots a_n\}$, where $a_i \subset n$. Suppose N is the total number of transactions and $T = \{t_1, t_2, t_3, \ldots\ldots t_N\}$ is the sequence of transaction, such that $t_i \subset D$. The support of each element of I is the number of transactions in D containing I and for a given itemset $A \subset I$; we can define its support as follows:

$$Support(A) = \frac{A \subseteq t_i}{N} \ldots\ldots\ldots (1)$$

Itemset A is frequent if and only if $Support(A) \geq minsup$, where $minsup$ is a user-defined global support threshold. Once the algorithm discovers all global frequent itemsets, each site generates global rules that have user-specified confidence. It uses frequent itemsets to find the confidence of a rule $R1$ and can be calculated by using the following formula:

$$Confidence(R) = \frac{Support(F_1 \cup F_2)}{Support(F_1)} \quad \ldots \ldots \ldots (2)$$

CHALLENGES OF DARM

All of the DARM algorithms are based on sequential association mining algorithms. Therefore, they inherit all drawbacks of sequential association mining. However, DARM not only deals with the drawbacks of it but also considers other issues related to distributed computing. For example, each site may have different platforms and datasets, and each of those datasets may have different schemas. In the following paragraphs, we discuss a few of them.

Frequent Itemset Enumeration

Frequent itemset enumeration is one of the main association rule mining tasks (Agrawal & Srikant, 1993; Zaki, 2000; Jiawei, Jian & Yiwen, 2000). Association rules are generated from frequent itemsets. However, enumerating all frequent itemsets is computationally expansive. For example, if a transaction of a database contains 30 items, one can generate up to 2^{30} itemsets. To mitigate the enumeration problem, we found two basic search approaches in the data-mining literature. The first approach uses breadth-first searching techniques that search through the iterate dataset by generating the candidate itemsets. It works efficiently when user-specified support threshold is high. The second approach uses depth-first searching techniques to enumerate frequent itemsets. This search technique performs better when user-specified support threshold is low, or if the dataset is dense (i.e., items frequently occur in transactions). For example, Eclat (Zaki, 2000) determines the support of k-itemsets by intersecting the tidlists (Transaction ID) of the lexicographically first two (k-1) length subsets that share a common prefix. However, this approach may run out of main memory when there are large numbers of transactions.

However, DARM datasets are spread over various sites. Due to this, it cannot take full advantage of those searching techniques. For example, breath-first search performs better when support threshold is high. On the other hand, in DARM, the candidate itemsets are generated by combining frequent items of all datasets; hence, it enumerates those itemsets that are not frequent in a particular site. As a result, DARM cannot utilize the advantage of breath-first techniques when user-specified support threshold is high. In contrast, if a depth-first search technique is employed in DARM, then it needs large amounts of network communication. Therefore, without very fast network connection, depth-first search is not feasible in DARM.