A Data Mining Approach for Unification of Association Rules in Distributed and Parallel Databases

Imran Qureshi¹, Mohammed Ali Shaik², Kanchi Suresh³, G. Rama Murthy⁴

¹¹³ Associate Professor in CSE department at Guru Nanak Institute of Technology (GNIT), Ibrahimpatnam, Hyderabad, Telangana, India.
²²⁴ Assistant professor in Dept of CSE at Aurora’s Research and Technological Institute (ARTI), Warangal, Telangana, India.
⁴Associate Professor at IIFT, Hyderabad, Gachibowli

Abstract: Due to the development of Parallel and Distributed data mining techniques which intends to generate association rules and correlate relationships among large set of data items since its inception. The present association rule mining handles successfully only either in distributed environment or the parallel environment but not in both at once. This process requires immense integrated knowledge in multiple datasets with communication overheads with low response time. The proposed algorithm extends the most classical algorithm Apriori & Optimized Distributed Association Rule Mining (ODAM) based on distributed and parallel transactional database system. We intend the proposed system to reduce the communication and interpretation costs; short time complexity; improve anatomy and efficiency of distributed association rule mining tasks.

Keywords: Apriori, ODAM, Parallel computing, Distributed computing.

1. INTRODUCTION

Due to massive research and development of storage devices tend to increase large space storage at cheapest cost has enabled many organizations to build large storage database and also collect large volumes of data. Latest software’s and organizations intend to extract useful information from the ultra large amount of data as traditional methods will be able to handle such data. Association rule mining (ARM) will implement tries to find most frequent patterns, associations among patterns, correlations among them, or casual structures sets of items or related objects in transaction database. The main idea is to find out the relations or dependency of the occurrence of one item set based on occurrence of the other item sets. Massive number of algorithms and architectures has been proposed by many researchers in this area but none could be seen to fit real life situations and efficiently respond to unforeseen changes in both parallel and distributed system with time.

Automated acquisition of knowledge in the areas of Artificial Intelligence that implements the If-then rules which are well-known techniques for representing the knowledge. Almost all the association rules are dependency rules that predict the occurrence of an item based on occurrences of other items, this approach is simple but effective and can help the commercial decision making like the storage layout.

The distributed data mining (DDM) is a semi-automatic pattern extraction of distributed data sources which comprises valuable knowledge from raw data produced by distributed parties for producing a unified global model that may present various challenges that relates to either the huge amount of managed data or their physical location and ownership.

2. LITERATURE SURVEY

This section gives a brief review of some important existing works in parallel and distributed data mining, distributed association rule mining and other related fields and concepts.

2.1 Data Mining (DM)

Data mining has emerged as important research area that is defined as a powerful new technology with great potential to help companies that focus on the most important information in the data they have collected about the behavior of their customers and potential customers it involves the use of sophisticated data analysis tools to discover previously unknown or valid patterns and relationships in large data set.

2.1.1 Distributed Data Mining (DDM)

Distributed data mining refers to mining of distributed data sets which are stored in local databases that are hosted by local computers which are connected through a computer network that is undertaken in an environment where users or data or hardware and the mining software are geographically dispersed is called as distributed data mining. These types of environments are also characterized by heterogeneity of data with multiple users and large data volumes. Data mining takes place at a local level and at a global level where local data mining results are combined to gain global findings which involves local data analysis from which a global knowledge can be extracted using knowledge integration techniques.

New methods for mining enormous amounts of heterogeneous data from many data sources that are emerged all the time and are recognized with sets of
association rules that can rapidly grow to be unwieldy as lower frequency requirements.
Many current data mining tasks can be implemented successfully only in a distributed environment that only concentrates on agent based data classification but not agent based association rule mining as the data transfer costs are estimated for data to be transferred from the data site to the distributed association rule mining server for a speculated period of time which is enormous and is hard to select an appropriate association rule mining method when no algorithm fits at all. Also many of the existing algorithms are only suitable for real-world scenarios where the comprehensive and needed many reviews of issues and challenges in current agent-based distributed association rule mining.
The system architecture for both parallel and distributed computing is shown in below figure:

![System Architecture](image)

**Fig.1. System Architecture**

2.1.2 Parallel Data Mining (PDM)
In the past decade there are revolutionary change been made to the hardware development related to the computation of the parallel instruction set. Many parallel computing hardware architectures have been evolved in recent times depending upon the cost and the type of computational problem parallel computing hardware is divided mainly into two categories that are: Common Parallel Computer Architecture(CPCA) and Super Computer Architecture (SCA). The classification of parallel computer is shown in Fig 2.
Super computers are very expensive and take long time to manufacture but faster in execution of instructions where as each parallel application does not need an dedicated super computer and more over many organizations cannot buy super computer due to its high cost.
After massive research a new alternative concept has emerged that is known as CPCA in which a large number of systems are combined on a network that comprises of cheap and easily available autonomous processors like workstations or PCs hence it become extremely popular for large computing purpose such as scientific calculations as compared to the SCA’s.

In most of the multiprocessor architectures more than one CPU is equipped with a single computer and the compiler is responsible for parallelizing the code automatically by implementing pass1 and pass 2 where as this type of architecture is not so efficient but better than a computer having a single CPU.
In a shared memory architecture a number of processors are inter connected to a common central memory using a bus is called as shared memory architecture and is also well known as Symmetric Multi Processor (SMP) where all the terminals will equal access to the other terminals operating system kernel which is designed to run on any machine.
In order to increase the speed of data sharing among all the processors a single address space is sufficient and there is a chance of processes corrupting each other’s data at the same time which can be overcome by using of semaphores and locks that are used to save the data from corruption which is mainly due to the bus contention which is implemented in SMP.
Distributed shared memory (DSM) is another type of shared memory architecture that is dedicated to each processor but the memories are connected through a common bus to form a shared memory and the inter process communication takes place through shared variables where the system hardware and software make it as single address architecture by removing the problem of bus contention and provides better performance than SMP.
In distributed shared memory all the nodes are connected through a network with their own independent local memory in distributed memory MIMD computer where each node is a fully connected through a network but the architecture is also known as loosely coupled because the nodes are not tightly integrated as that of in shared memory architecture each node cannot directly access the memory of other nodes so it is called No Remote Memory Access (NORMA) and the nodes can communicate with the others through the communication network by using message passing technique this type of network that connects the nodes may be of having different network topologies. Cluster of workstation (COW) and PC cluster fall under this categories which is a collection of independent computers that are physically interconnected.
through LAN.

Supercomputing has extremely high execution rate and extremely high I/O throughput as it needs very large primary and secondary memory hence the cost and the time is the two crucial factors for manufacturing a super computer. The principal architectures of supercomputing are Massively Parallel Processor (MPP) and Parallel Vector Processor (PVP) where MPP system is the collection of hundred or thousand of commodity processors interconnected by high speed and low latency communication network where the memory of the processors in MPP is distributed but the processors are synchronized by the blocking message passing operations as each process has its own physical address space and communicates with the others through message passing primitives where as PVP makes the use of huge number of vector registers and instruction buffer instead of cache.

2.2 Distributed Memory Parallel Computing Models

Many number of distributed memory with parallel computing models have been evolved in which each complete computer has its own memory which are connected through a communication network either it may be BSP model or LogP model which are most well known models which removes the shortcoming of the shared memory computational memory.

2.2.1 BSP Model

The bulk synchronization parallel model (BSP) has three components that are p-numbers of processors/memory, supersteps with periodicity L and the bandwidth factor g which is defined as the ratio of computation to communication where each processor or memory can carry out computation on local data after each L unit of time a global check is done to verify whether all the components are finished. If not the other superstep is allowed to finish all the components in any of the case bandwidth limitation exists with BSP model that is capable of sending maximum messages by a limiting factor h=L/g which is known as h-relations.

2.2.2 LogP Model

LogP model consists of four parameters- P numbers of computers, L (latency of message passing), O (overheads involved in message passing) and g (minimum time interval between successive messages) where at most L/g messages can be transmitted from one processor to another at any instance a process has more than this number of messages to transmit, then it stalls until the message can be sent without exceeding the capacity limit. This model is called as asynchronous in nature and thus message passing latency is unpredictable hence all the parameters are not considered at same time since some of them are neglected.

2.2.3 Agent Model

An Agent is a computer software that are situated in some environment and are capable of autonomous action in this environment in order to meet their design objectives the intelligent agents can react to change in their environment that have social ability and the ability to use computational intelligence to reach their goals by being proactive where as all the agents are designed to perform specific tasks and capable of autonomous action and decision making by combining multiple agents in one system to solve a problem where the resultant system is a Multi-Agent System (MAS). Datasets generated from Parallel and Distributed environments are converted into agents that comprises of agents that individually solve problems that are simpler than overall problem. All the agents can communicate with each other and assist each other in achieving larger and more complex goals with the Foundation for Intelligent Physical Agents (FIPA) is a non-profit making international organization dedicated to promoting the industry of intelligent agents by openly developing specifications to support interoperability amongst agents and agent-based systems.

3. Research Methodology

In a distributed or parallel environment I is a itemset that comprises of goods or commodities ranging from I1 to In in a sample database D where every transaction has a unique id called as TID and a itemset, the association rule implicates the form for evaluating the matrices is:

$$\sum_{x \in \text{TID}} x \rightarrow y$$

(1)

The confidence is measured based on how often the itemsets in dataset Y appear in dataset X and the frequent itemsets are those whose support is greater than or may be equal to minimum support threshold supplied to the algorithm.

We are considering huge database D with millions of ongoing transactions T and our goal is to find association rule mining that finds all the rules having support ≥ minimum support threshold and confidence ≥ minimum confidence threshold.

Duo-step approach for mining association rules is preferable in our study:

- Frequent Itemset Generation is used to generate all itemsets whose support ≥ minsup provided.
- Rule Generation is used to generate high confidence rules from each frequent itemset where each rule is a partitioned based on binary partitioning method on frequent itemset.

The most expensive part of the approach in terms of CPU cycles utilization is Frequent itemset generation so we concentrate mainly on the first step in the following discussion.

Suppose a transactional database D is stored in parallel and distributed spots ranging from s1 to sn.

$$D = \sum_{i=1}^{n} D_i$$ a local dataset.

(2)

$$|D| = \sum_{i=1}^{n} D_i$$ a global dataset.

(3)

A local data set is said to be frequent item set when:

$$\text{I.sup} \geq |D| \times \text{minsup}$$

(4)

According to the above equation I is said to be global frequent item set.

4. Implementation

Designing the parallel programming is always a challenging matter that focuses on imposing and
designing parallel programming which is implemented widely using two methodologies are widely used for the purpose of parallel programs. They are auto-parallelization compiler and library based software. Whereas opposite in distributed environment.

The implementation of both parallel and distributed environments is done using the algorithm that generates global frequent k-itemset we scan the local databases only once (during constructing the new storage structure) and prune step doesn’t need so I/O spending is saved by which time complexity of the algorithm is reduced and efficiency is improved.

```c
for(i=0;i<k-1;i++)
{
    for(j=0;j<k-1;j++)
    {
        if(item[i]==item[j])
        {
            c[k]=0;
            tedRec[k]=0;
            count[k]=0;
            for(l=0;l<k-1;l++)
            {
                c[k]=c[k]*item[l];
                tedRec[k]= intersect(tedRec[i], tedRec[j]);
                count[k]=mod(tedRec[k]);
            }
        }
        if(count[k] >= D*minsup)
        {
            Li[k]=Li[k]+c[k]
        }
    }
}
```

In the above program snippet the candidate itemset requires a tidRec structure that uses a large memory space when transaction database is huge and we are still concentrating on how to balance the time and space complexity to get the best solution need yet further study.

In parallel and distributed systems websites generate frequent itemset from their local database our system is responsible for generating the superset of all local frequent itemsets and computing the support of each itemset then confirming the frequent itemsets based on the minsup given by the user or the software, the association rules based on the different databases can be acquired and stored in the knowledge database this implementation can be realized using:

- By generating global frequent itemset scanning all the local databases and exchanging the support count between local websites by generating frequent k-itemsets (k>1).
- Parallel computing websites generate parallel frequent itemsets based on local database by AprTidRec algorithm and send to global website which combines all the frequent itemsets from parallel computing websites then generate candidate global frequent itemsets broadcast the candidate global frequent itemsets in distributed computing website determines the newly added itemsets and their distributed local support given by the user.

5. EXPERIMENTAL RESULTS

We have taken a data set of size 970791 instances with four itemsets and used WEKA tool to generate the results and the results are:

<table>
<thead>
<tr>
<th>Scheme: weka.associations.Apriori</th>
<th>N 10 -T 0 -C 0.9 -D 0.05 -U 1.0 -M 0.1 -S -I -O -C 1.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Relation: Distributed-Parallel-Computing</td>
<td></td>
</tr>
<tr>
<td>Instances: 970791</td>
<td></td>
</tr>
<tr>
<td>Attributes: 4</td>
<td></td>
</tr>
<tr>
<td>i1</td>
<td></td>
</tr>
<tr>
<td>i2</td>
<td></td>
</tr>
<tr>
<td>i3</td>
<td></td>
</tr>
<tr>
<td>i4</td>
<td></td>
</tr>
</tbody>
</table>

--- Associate model (full training set) ---

Apriori

Minimum support: 0.25 (242698 instances)
Minimum metric <confidence>: 0.9
Number of cycles performed: 15

Generated sets of large itemsets:
Size of set of large itemsets L(1): 7
Size of set of large itemsets L(2): 11
Size of set of large itemsets L(3): 6
Size of set of large itemsets L(4): 1

Best rules found:

1. i3=1 642099 ==> i4=1 642099 conf:(1)
2. i1=0 603876 ==> i4=1 603876 conf:(1)
3. i1=0 i2=1 458640 ==> i4=1 458640 conf:(1)
4. i1=0 i3=1 386022 ==> i4=1 386022 conf:(1)
5. i2=1 i3=1 378378 ==> i4=1 378378 conf:(1)
6. i2=0 336339 ==> i4=1 336339 conf:(1)
7. i1=0 i2=1 i3=1 313404 ==> i4=1 313404 conf:(1)
8. i2=0 i3=1 263721 ==> i4=1 263721 conf:(1)
9. i1=1 i4=1 256077 ==> i3=1 256077 conf:(1)
10. i1=1 i3=1 256077 ==> i4=1 256077 conf:(1)

The implementation of proposed system in WEKA 3.5 is:
6. Conclusion and Future Work

In this paper we tried to demonstrate the new approach for mining ordering rules from ordinal distributed database system where mining association rules in distributed database and parallel databases is an important aspect of data mining domain the algorithms based on AprTidRec and Apriori for mining association rules is discussed thoroughly in this paper. The efficiency of the algorithm is verified and the overall system realization is given based on the algorithm with the implementation of WEKA 3.5 tool.

It is believed that the system will help to discover new, hidden, previously undiscovered patterns in both parallel and distributed environments or systems which leads to a dynamic improvement in business yield and turnover. Future works will treat details about the adaptivity of the mining agent in mobile environments and the implementation of the architecture proposed in this work.

References

AUTHOR

Imran Qureshi is presently working as Associate Professor in CSE department at Guru Nanak Institute of Technology, Hyderabad, and A.P. He received his B. Tech and M.Tech. Degree with specialization in Information technology and Computer Science respectively from JNTUH, Telangana. His main research interest includes Data mining, Data Structures and algorithms and Distributed operating systems. He is Pursuing his PhD from Jawaharlal Nehru Technological University, Hyderabad.

Mohammed Ali Shaik received his M.Tech degree in Computer Science and Engineering from JNTUH, Andhra Pradesh, India. Currently he is working as Assistant professor in Computer Science and Engineering department at Aurora’s Research and Technological Institute (ARTI).

Kanchi. Suresh is presently working as Associate Professor in CSE department at Guru Nanak Institute of Technology, Hyderabad, and Telangana State. He received his B. Tech Degree in Computer Science and Engineering from V.R.Siddhartha Engineering College, Vijayawada, and M.Tech. With specialization in Software Engineering from SIT-JNTUH, Hyderabad. His main research interest includes Data mining, Data Structures and Algorithms and Compilers.

Dr. Garimella Rama Murthy completed his PhD during 1986-1989 in computer engineering from Purdue University, USA. He completed MS in electrical Engineering from Baton Rouge University, USA. He Received Rashtriya Gaurav award in the year 2009 from IIFS, New Delhi. He has published 200 Publications (Journal and conference). He has been visiting scholar to many reputed Universities like University of Illinois at Urbana, IIISc Bangalore. He is Currently Working as Associate Professor at IIIT, Hyderabad, Gacchibowli. His main research interest includes Wireless sensor networks, Adhoc networks, Data Mining and performance evaluation.