CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

The method to gain new models for data analysis in order to get benefits of business is termed as Data mining (Jiawei Han et al. 2006). To get scarce events which are not shown explicitly within huge volume of data is a major problem of data mining. It is a process, which recognises patterns as well as relationships in data. Knowledge discovery in databases (KDD), synonym for data mining, is a method to extract notable forms and associations in huge amount of data. The narration is, suppose there is a certain amount of data, and the need is to search for specific events of a specific nature within data. Though data is arbitrary, as volume of data increases, the number of occurrences of these events will also increase. That random data which seems to have unique features, appears considerable however it does not. Likewise data mining performs a significant role in data analysis to enhance the business. Data mining is seen in a perspective of information era development.

Database Technology, Statistics, Artificial Intelligence, Data Visualization, Knowledge Engineering, Object-Oriented Methods, Machine Learning, High-Performance Computing and Information Retrieval (Jiawei Han et al. 2006) all together contributes to data mining which is an interdisciplinary field. Various tools and techniques are being used in the above mentioned subjects to explore large data sets. The areas of data mining applications are trade
such as banking & retail, research such as astronomy & medical and nation defence such as criminals & terrorists identification. Other research areas of data mining techniques are mathematics, genetics, marketing and cybernetics.

1.2 RESEARCH BACKGROUND

1.2.1 Distributed Data Mining

Integrating data at a single place and mining data will not satisfy the users in an ample way due to existence of distributed databases. Hence the distributed data mining research is changed into a primary focussing area. The process which helps to extract knowledge from huge volume of data is said to be Knowledge Discovery in Databases (KDD). It makes an illusion that data is available in a central location with an easier access mechanism. Virtual warehouses and distributed databases contribute to the above need of centralised access. Furthermore, information is accessible everywhere due to the improvements in information technology and networked computers. Extension of data mining is said to be distributed data mining which is giving focus to distributed data and resources. Hence it becomes a research problem to discover association rules from the huge volume and dimensions of datasets in a parallel way (Kotsis et al. 2006).

The valid reasons for major role of distributed computing in data mining process are as follows. The process of data mining leads to consumption of large storage space and high computation time. It is important to develop ways to distribute data among numerous sites in a dynamic way for the purpose of making system scalable. It is inefficient and involves a major security risk to make centralized processing of distributed data. Hence the techniques are offered to do data mining in a distributed way instead of centralised nature. There are
enhancements in communications for wired as well as wireless networks which showed the way for many pervasive distributed computing environments. Most of these environments work with various distributed sources of voluminous data, numerous compute centres, and distributed user groups. Investigating and testing the distributed data sources require a data mining technique planned for distributed applications. Distributed data mining (DDM) works for distributed data and also gives attention to distributed resources.

Lamine M Aouad et al. (2007) explains approaches of grid for distributed data mining applications, and be specific to generation of frequent itemsets. It has been said that only a limited number of algorithms deal with the grid distributed solutions compared to sequential execution algorithms and implementations. The paper makes it necessary to come with platform independent efficient distributed and local pruning grid-based approaches and also global strategy for gathering smaller globally frequent itemsets. This paper introduces decomposition of block based datasets to solve constraints of memory and workload issues. Communication, synchronization problems and tools overhead are reduced in this implementation. A grid algorithm is considered as a best one which avoids multiple communication phases and synchronizations as required by. The proposed approaches show the improvement compared to the FDM algorithm and it is confirmed that implementations in grid are having a necessity to eliminate synchronizations. Likewise, it is stated that the pruning strategies of global nature are performance limited for scalable environments such as grid and broad range conditions are applied on the datasets. Computing time of the Grid-based version of the approach shows better performance up to 37% and computing time middleware issues are not given importance. In addition, it is important to show the workload management of heterogeneous data distribution which is a dynamic one.
Scalability is an important issue to be considered in grid-based algorithms, the focus is given to increase processing resources in memory and environment with bandwidth constraints. The issue of memory constraints are tackled using previous approaches for frequent itemsets generation which is accepted locally. But environment specifications are not tackled using this approach. It is mandatory to take care of the constitution of novel distributed data mining system. Since Data mining system mines data from various datasets located at various sites, data mining system with distributed support is required. The essential requirements for the data mining technology on user part are as follows.

- For example many enterprises have many branches and every branch creates its own structure, hence the scalability of the enterprise size is to be taken care.
- Protection of secure transmission and retaining privacy of data is most important to all enterprises because of market competition.
- The requirements of efficient association rule mining is more because of huge volume of data.

Due to the reasons stated above, traditional data mining techniques of centralized databases are unable to meet the data mining requirements. The drawbacks of the recent technology are also become apparent.

1.2.2 Cloud Data Mining

Cloud Computing becomes a new business model to distribute the computing process to many computers which are all comprised inside a resource pool. The result is to get a range of application systems with computing power, memory utilization and a variety of software services on demand. The “Cloud
computing” is being described as a system environment or a type of software application (Juan Li et al., 2012). Here in cloud the system environment brings a meaning in real time setup like, it is able to dynamically provision, configure, re-configure and de-provision a system. The server is considered to be either physical or virtual. The cloud computing which includes high computation resources is termed as high end. Storage and computing power seems to be inexpensive due to the innovation of cloud computing. It provides an environment for mining of large amount of data without storage hurdles. Cloud computing handles various approaches such that it tackles increasing demands of data mining for high-dimensional and big data. MapReduce is defined as a programming framework (Xin Yue Yang et al. 2010) and an implementation which is associated and designed for large data sets.

Cloud computing is proposed for the solution of increasing storage needs of IT enterprises. Data storage devices are arising with increasing costs as well as data generation is also in a fast pace. Enterprises or individual users finds it difficult and expensive to do changes in hardware of the system frequently. To reduce storage cost it is optimum to choose data outsourcing with cloud, it also reduces maintenance costs by a reasonable value. The technology followed in cloud is, it moves the user data to different data centres at various places and user is having restricted access and control. But security challenges are to be addressed definitely which should not overshadow the major benefits of cloud. Integrity which means correctness of data in cloud is to be assured. . The cloud also provides ways for the user to confirm the maintenance of integrity of data. The research work by Chery L. Shavers et al. (2011) provides integrity of data to customer data in the cloud that is being availed by the customer for verifying data perfection in cloud.
Thus sharing of resources is possible in distributed computing in a remarkable way which assists users of multiple environments. Sharing and mandatory synchronised use of various resources in an organized environment makes the remarkable performance and usage of cloud computing achievable. Cloud computing brings a nature of large-scale sharing in a controlled nature and interoperation between resources that are handled by various people. Security is a must (Zhidong Shen et al. 2010) in cloud computing to ensure authorized access and secure behaviour. The cloud participants and the cloud providers both should have a mutual trust and the communicating participants also have a mutual trust.

1.2.3 Association Rule Mining

Frequent itemset mining is defined as a technique of retrieving sets of items that occur frequently in a dataset. It becomes a main focus in data mining and existing algorithms figure it out as a difficult process to handle large datasets. A powerful way used to sort out regularities in trends of data is termed as Association rule induction. Association rules bring out sets of data instances which occur often together (Sotiris Kotsiantis et al. 2006). Rules are formed with these information. Using relationships among items or group of items an association rule is depicted. But only sensitive and consistent association rules are useful. The measures of association rules are Support and Confidence which are used as standards to assess association rules.

The purpose of finding out association rules for the set D of transactions, is to sort out all association rules with support value and confidence value above the minimum support value (called mnsup) and minimum confidence value (called minconf) respectively which is of user choice. In association, a pattern is discovered using relationship of a specific item with other
items in the same transaction. For example, in market basket analysis the association technique is used to identify what products a customer frequently purchases together. Based on this data business houses can have suitable marketing drive to sell more products and make more profit.

One of the main focussed branch of data mining is Association Rule Mining (ARM). \( X \rightarrow Y \) is the form of Association Rule if sets of items are \( X \) and \( Y \), in the group of transactions considered for mining. This gives the meaning that \( X \) tend to contain \( Y \). For example, “30% of all transactions which contain beer also contain the item diapers; 2% of all transactions contain both these items”, is an Association rule. 30% is depicted as the confidence of the sample rule, and 2% is said to be the support of the sample rule. Association Rules which satisfies minimum support value and minimum confidence value as per the choice of user. These frequent patterns are interesting as well as helpful for data mining tasks of clustering, classification and so on.

Agarwal et al. (1993) describes that the effective innovation of association rules in large databases creates great contests for researchers and data mining application developers. To achieve scalability of data mining process necessary sampling, in addition memory vs. disk processing and high performance computing are used. In parallel computing platforms, Many KDD systems have been implemented to attain high performance in a parallel platform for the analysis of huge data sets. Thus the need for parallel and distributed infrastructure which is of high configuration is raised.
1.2.4 Apriori Algorithms

Rakesh Agrawal et al. (1994) states that multiple passes are needed in the discovery of large itemsets with frequent itemset algorithms. The support of individual items is counted and large one among that is determined having value equal or above to the minimum support. As subsequent passes proceed, a set of itemsets which are all large in earlier pass is taken and is used to generate new large itemsets which is hypothetically called as candidate itemsets. It counts the original support value for the candidate itemsets while performing the iteration over the data. Final stage of iteration identifies, large candidate itemsets and it becomes the seed for the subsequent iteration. This process stops when new large itemsets are not found further. This algorithm is termed as Apriori algorithm, popular in association rule discovery which was introduced in the year 1993 by Agrawal.

The principle of the Apriori algorithm is defined as an itemset is frequent only if all its non-empty subsets are also frequent. An iterative approach is used to extract the frequent itemsets in current iteration, where previous iteration results are used. It starts with finding frequent itemsets with one item where an item occurring many times than a minimum threshold value defined by the user, (support count) is termed frequent. A main shortcoming of the Apriori algorithm is that the candidate set for the second phase is too large if the number of items is huge and the entire dataset is scanned several times in subsequent iterations. Two algorithms Apriori and AprioriTid were introduced by R. Agarwal et al. (1994) to discover all significant association rules for the chosen transaction datasets. Later, many improvements were done to make Apriori better, efficient and faster.
Traditional model gives equal treatment to every transaction by using support measure for association rule mining. In real data sets, every transaction has unique weight. Weighted Association Rule Mining (WARM), has come up with WSupport, a new measure based on link that computes the authority weight and also focusses on significance of an item ‘i’. WSupport is considered as a generalization of support, by considering transaction weights. These are computed using global link structure of database. WSupport is appropriate when compared to count based measurement. The field of association rule mining (P. Laxmi et al. 2011) is a recent innovative field to find out knowledge with uncountable procedures and components such as utility, weight and so on. The association rule mining shows improved quality by bringing out the association relationships for the industry. The methodology which is apt is brought out using weight factor and utility. During first phase, traditional Apriori algorithm is applied, which finds out that if ‘n’ items are repeating then ‘n-1’ items will also repeat. The scores of weightage termed as W-Gain, utility termed as U-Gain and diminution termed as D-sum, are calculated. This is called as EUV score and used to derive a subset of important association rules. The results can be used for high utility association rules which helps for business improvement.

Item weight, transaction weight and weighted support can be calculated in different ways. Itemset Transaction Weight is the product of weights of all items in the itemset for a single transaction. Weighted Support may be the sum of itemset transaction weight of all transactions in which, all the itemset occurs, divided by the total number of transactions.

Hash tree algorithm (Shanthi.R et al. 2011) aims to solve the shortcomings of apriori algorithm by finding out way to reduce the generation of number of candidate k-itemsets, specifically candidate 2-itemsets. The count of
itemsets in C2 is reduced by applying hash techniques, hence effective scan is done to find out L2. A hash function is mapping itemsets into various buckets according to hash counts. If bucket count of a 2-itemset in the hash table 1 is below the support threshold it is assumed to be not frequent and hence should be eliminated from the candidate set.

Hence the weight concept of weighted apriori and hash tree construction concept of hash tree apriori can be integrated to get a novel hybrid apriori algorithm for association rule mining.

1.2.5 Hadoop MapReduce

A framework that helps to write software for handling data which is huge in volume and supports parallel execution is called as MapReduce (Juan Li et al. 2012). It is a processing technique and distributed programming model which is java based. Map and Reduce are two jobs performed using a MapReduce algorithm. Map task inputs a set of data and changes it into other set of data, where elements are split down into tuples of key/value pairs. Next reduce task inputs the map output, those data tuples are combined into a smaller set of tuples. Map task is followed by reduce task.

The benefit of MapReduce is to expand and process data with many computing nodes with less complex items. MapReduce model consists of data processing primitives, mappers and reducers. Next process to decompose a data processing application into mappers and reducers. After an application is written in MapReduce form, it is scaled up to run over numerous machines in a cluster by changing the configuration in a less complex way. MapReduce model is being
attracted by programmers due to its scalable nature. Hadoop MapReduce is capable of processing data-sets of multi-terabyte size, on thousands of nodes in an error free parallel way. Hence distributed data mining can be achieved successfully using hadoop MapReduce framework.

1.3 MOTIVATION

Association Rule Mining remains a very familiar and effective way to extract meaningful information from big datasets. It tries to find out possible associations between items in large transaction based datasets. Frequent patterns are generated to create these associations between items. The apriori algorithm along with its set of enhanced alternatives, which were one of the earliest proposed frequent pattern generation algorithms still remain a preferred choice due to the easy implementation and natural way to be parallelized. While many efficient single-machine methods for apriori exist, the massive amount of data available these days is far beyond the capacity of a single machine. The apriori algorithm proposed by R. Agarwal et al. (1993) works on the fundamental statement that an itemset is frequent only if all non-empty subsets are also frequent. It uses an iterative approach where results of the previous iterations are used to identify frequent itemsets in the present iteration.

Scalability and high dimensionality of databases (EhabEzat et al. 2012) lead to the need for distributed data mining algorithms. Apriori algorithm for association rule mining serves well and efficiency of apriori algorithm (Keun Ho Ryu et al. 2013) can be applied in distributed environment also. A novel hybrid apriori algorithm can be developed for distributed environment suitable for handling massive amount of data. Integrity and security (Minqi Zhou et al. 2010)
are the threats in distributed data, hence it has to be protected from unauthorized access and the data has to be safe guarded with novel algorithm.

The motivation specified above clearly depicts the importance of distributed data mining, significance of apriori algorithms in association rule mining, cloud approach of data mining, supporting framework of cloud for mining data using the above algorithm and the necessity of introducing techniques for maintaining integrity and security of data in the cloud environment.

1.4 PROBLEM STATEMENT

Weighted and Hash tree apriori algorithms consume more memory and time when it is executed in sequential computation. Weighted apriori utilizes link based measures to discover effective associations. However, weight calculation of weighted apriori algorithm is time and memory consuming even if it is executed in a distributed environment. Integrity and security issues are to be addressed in distributed data by applying novel Hybrid Weighted-HashT Apriori algorithm.

1.5 OBJECTIVE

The research objectives are listed out as follows: 1) To design and implement weighted and hash tree apriori algorithms in a distributed environment in order to reduce time and memory complexity of execution, 2) To effectively mine the data with a novel hybrid algorithm, 3) To solve the integrity and security issues in distributed data.
1.6 CONTRIBUTIONS

Based on the motivation described above, a detailed literature survey is carried out, problem statement is defined and objective of the research is determined. The research has primarily made three contributions to the field of distributed cloud data mining which are all listed below

Contribution 1:- Design and implementation of weighted and hash tree algorithms in distributed environment.
Contribution 2:- Design and development of a hybrid Weighted-HashT apriori algorithms in distributed environment.
Contribution 3:- Incorporation of integrity and security of data in hybrid algorithm.

This work starts with the study of various association mining algorithms especially apriori mining algorithms and finds out the fact that compared to conventional apriori, weighted and hash tree apriori performs well and explains its improvements with the help of literature review. It shows performance improvement when implemented in a distributed environment. The main task of this work is to come up with a new modified algorithm in a eucalyptus environment which overcomes drawbacks of normal weighted apriori and hash tree apriori algorithms. Results of novel implementation are being discussed. The work concentrates on deriving a new apriori algorithm which performs well compared to others in cloud using eucalyptus nodes and hadoop distributed file system. The work also discusses the results obtained from new implementation in a cloud eucalyptus environment and how it overcomes the existing one. Weighted apriori algorithm (Fengshan Bai et al. 2008) introduces
weight which uses hubs and authorities (Ke Wang et al. 2002). The new research work proceeds with weighted support calculation based on the costs assigned to both items as well as transactions. The algorithm finds out the rules that have a weighted support larger than a specified threshold. The method also follows downward closure property. Hash Tree apriori algorithm counts all the 1-itemsets for each transaction. In the same pass all the possible 2 itemsets in the transactions are calculated as hash values and mapped to a hash table. Hash table is used to reduce the candidate itemsets. Using the support count calculated, the algorithm finds out the frequent itemsets.

To implement the distributed algorithms, MapReduce model is applied and the functions performed by MapReduce model are partitioning the input, scheduling the map reduce jobs among participating nodes, handling node failures and also achieving the required network communications. One implementation of MapReduce is Apache Hadoop which gives a setup for distributed computing and available as open source hosted by the Apache Software Foundation. Hadoop with Hadoop Distributed File System (HDFS) is based on MapReduce which is a distributed data processing model and execution setup that executes on large clusters of commodity machines.

In the background study conducted it has been found that weighted and hash tree algorithms are performing well compared to remaining apriori algorithms, but they have been implemented in standalone machines used for data mining. It has been found that it is better to have the same algorithms in distributed environment as discussed earlier. Execution time of both algorithms gets improved because in normal apriori the process will happen in single node, but in hadoop the process will happen parallel in all the four nodes using MapReduce framework. Number of candidate set generation is less for hash tree
apriori as compared to weighted apriori. Since the hash tree takes up the complex execution process, time and memory usage will be high. Though weighted apriori performs well and gives more number of association rules, hash tree gives better rules. Though hash tree gives better rules it occupies more memory and time.

To overcome the negatives and to combine the positive approaches of both weighted and hash tree apriori algorithms a new tree based hybrid apriori approach is introduced in the proposed work. This approach will reduce the computation time as well as the accuracy of the frequent item prediction. The new hybrid approach will avoid the candidate set generation. The new algorithm will generate the tree structure and analyse their height, weight and reach ability for each node. Reachability refers to the ability to reach the end point of the transaction of each item in the tree. Data mining in a distributed environment surely has to prove the data integrity before and after the distribution, whether any integrity and security violations have been done to data. Here in this algorithm this process has been done using Hash Message Authentication Code and it is proved to be correct. As well as cryptographic technique is also applied for data security.

As the research initiates the discussion from Data mining which finds out unknown facts from huge amounts of data in database to make decisions. Next Association rule mining is one of the primary focussing area of research in data mining. Next, due to improvement of networks and IT, Distributed database draws attraction. From geographically distributed database, Distributed data mining finds out useful knowledge for management decision making. Distributed data mining aims to do mining tasks with computers on various sites of internet. Transmitting large amount of network data, is being decreased as well as efficiency of mining is improved. This thesis concentrates on study of distributed
association rule mining algorithm based on recent research scenario. Further it proceeds to design a novel distributed association rule mining algorithm.

1.7 ARCHITECTURE DIAGRAM

The contributions of this research work as explained in section 1.6, are depicted in the proposed architecture diagram of figure 1.1.

![Figure 1.1 Proposed Architecture Diagram](image_url)
Data is distributed in Hadoop Distributed File System (HDFS) using MapReduce framework, whereas the algorithm is defined as map tasks to be executed on the distributed data which is available on various clusters of hadoop nodes. Then the results are combined using reduce task. Data is encrypted before distributing as well as decryption is done after the receipt of the results. HMAC (Hash based Message Authentication Code) is generated before sending and after receiving data in order to ensure that no data is missed or manipulated.

1.8 EXPERIMENTAL ANALYSIS

After designing and implementing weighted and hash tree apriori algorithms in distributed environment the following experimental analysis are done.

1. Investigations of execution time and memory utilization for weighted apriori algorithm (Retail itemset)
2. Investigations of execution time and memory utilization for hash tree apriori algorithm (Retail itemset)
3. Investigations of execution time and memory utilization for weighted apriori algorithm (Mushroom dataset)
4. Investigations of execution time and memory utilization for hash tree apriori algorithm (Mushroom dataset)
5. Comparison of the performance of proposed work with Chess, Kosarak, T40I10D100K and T10I4D100K datasets

Then with the hybrid algorithm implementation the following investigations are done.
1. Investigations of execution time and memory utilization for hybrid Weighted-HashT apriori algorithm (Retail itemset)
2. Investigations of execution time and memory utilization for hybrid Weighted-HashT apriori algorithm (Mushroom dataset)
3. Performance Comparison of proposed work with Chess, Kosarak, T40I10D100K and T10I4D100K datasets
4. Investigations of execution time for datasets with various sizes (with and without Hadoop)
5. Investigations on scalability
   a) Effect of hadoop nodes on memory utilization
   b) Effect of hadoop nodes on execution time

Finally after incorporating the integrity and security concept the following experiment analysis is done.
1. Investigations of execution time and memory utilization for secure hybrid Weighted-HashT apriori algorithm (Mushroom dataset)
2. Investigations of encryption time for secure hybrid Weighted-HashT apriori algorithm for various datasets
3. Investigations of decryption time for secure hybrid Weighted-HashT apriori algorithm for various datasets
4. Investigations of execution time for datasets with various sizes (with and without Hadoop)

1.9 OUTLINE OF THE THESIS

The proposed thesis work is organised as follows.

Chapter 2: Review of Literature
This chapter explains about the literature survey being conducted. The chapter depicts in a clear understandable way about

- Distributed data mining,
- Cloud,
- The need to go for cloud data mining,
- Apriori algorithms and its variations developed throughout the years,
- Map reduce hadoop framework which is the apt platform for cloud data mining,
- Security and integrity measures of distributed data to be followed specifically while moving to a cloud platform.

**Chapter 3**: Analysis of Existing Approaches of Apriori and Distributed Data Mining

This chapter describes

- Analysis of sequential and grid based approaches of apriori algorithms,
- The experimental results of both,
- The experimental results of various apriori algorithms to prove which one is better.

**Chapter 4**: Design and Implementation of Apriori Algorithms for Distributed Framework

This chapter pronounces the initial work of this research, i.e.

- Distributed design of Weighted Apriori and HashT Apriori algorithm,
- Execution environment to implement the above algorithms,
- The interpretation of results obtained in the experiment,
- Merits and demerits of both the above algorithms

**Chapter 5**: Design and Development of a Hybrid Apriori Algorithm

This part of the discussion describes the proceeding work of

- Design of a new hybrid algorithm,
- Execution environment to implement the novel algorithm,
• Interpretation of results with detailed inference to the field of distributed data mining,
• Merits and demerits of the novel algorithm.

Chapter 6: Incorporation of Integrity and Security in Hybrid Apriori Algorithm

This chapter explains the third contribution of this research to the field of secure distributed data mining and depicts which techniques are applied for upholding integrity and security of data in the process of mining.

Chapter 7: Conclusion and Future Work

The final chapter concludes with inference of results, summary, suggestions, and scope of proceeding further on the basis of key findings in the research.