Analysis on Parallelization of Apriori Algorithm in Data Mining

Anupriya
Department of Computer Application
Graphic Era Hill University, Dehradun

Ashok Kumar, Ph.D.
Department of Computer Science & Engineering
GRD IMT, Dehradun

ABSTRACT

Many algorithms are designed to analyse volumes of data automatically in an efficient way so that the users don't have to look through that massive amount of data manually for generating various association rules among them. Apriori algorithm, which is the most famous and frequently used data mining algorithm. Our main focus is to parallelize the Apriori algorithm in such a new way that when we will implement on a large database , it will lead to less time consuming and fast execution for generating frequent itemset.

Keywords

Apriori algorithm, Association rule mining, frequent itemset, Parallelize the Apriori algorithm

1. INTRODUCTION

Data mining which is a relatively young field of computer science, is the process by which new patterns are generated in a large data set. The main goal of the data mining process is to extract information or knowledge from an existing data set and transform it into a human-understandable structure for further use. That information can be used in various forms such as cost cutting or increasing revenue. The term data mining is somehow new, but the technology has been there for many years.

In this information age, we have been collecting tremendous amounts of information because we believe that information, from the technologies such as computers, satellites etc leads to power and success. With the invention of data and other storage devices helps us to collect all type of data

Unfortunately, this massive amount of data stored on disparate structures became overwhelming very rapidly. The initial chaos led to the creation of database management systems (DBMS) and structured database. For the large amount of data we have database management systems (DBMS) which very crucial assets for managing this huge data and especially for getting effective and efficient retrieval of particular information from it The generation of database management systems has also contributed to massive accumulation of all sorts of information. Today, we have a lot more information than we can handle, from scientific data and business transactions, to satellite images, text reports and military intelligence. Information retrieval or data mining is not enough for decision-making. With huge amalgamation of data, we have now created new desires to help us make better managerial choices. These needs are automatic summarization of data, extraction of the "essence" stored information, and the patterns discovery in raw data [10-15].

2. A SURVEY

Due to the development of new businesses and interest people in these businesses there has been a dramatic increase in the amount of information from GB's to TB's .It has been figured out that the amount of information in the world doubles every 20 months and the number and size of databases are increasing even more rapidly. The increase in use of electronic data gathering devices such as point-of-sale or remote sensing devices has added to this explosion of available data. Below graph illustrates the data explosion [2-10].

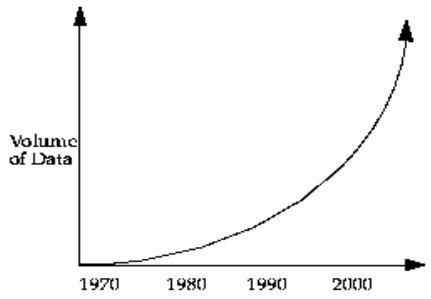

Fig 1.shows the data explosion

The storage of data became easier and cheaper as the cost of computing power and electronic data storage devices decreased rapidly. The organisations had concentrated so much attention on the accumulation of data; the problem was now what to do with this valuable resource. It was realized that information is at the core of business operations and that decision-makers could make use of the data stored to gain valuable insight into the business. Traditional on-line transaction processing systems are good at feeding and saving data into databases quickly, safely and efficiently but are not good at delivering meaningful analysis in return. Data analysis can provide more knowledge about a business by going beyond the data explicitly stored, to derive important knowledge about the business. This is where Data Mining or Knowledge Discovery in Databases (KDD) has obvious gains for any enterprise. Data Mining, or Knowledge Discovery in Databases (KDD) as it is also known, is the nontrivial extraction of implicit, previously unknown, and potentially useful information from data. This includes a number of different technical approaches, such as clustering, data reduction, data summarization, learning classification rules, finding dependency networks, analyzing changes, and detecting anomalies.

In data mining analysis, the best techniques are those which are developed with an orientation towards large volumes of data, making use of as much of the collected data as possible to arrive at reliable conclusions and decisions which is a parallel to a mining operation where huge amounts of low-grade materials are sieved through in order to find something of value.

3. ASSOCIATION RULE MINING

Association rules are one of the most researched areas of data mining and have recently grabbed much attention from the database community. They have been proved to be quite useful in the marketing and retail communities as well as other more diverse fields

Association rule mining (ARM) is a core technique for data mining which discovers patterns or rules among items from large database of variable-length transactions. The goal of ARM is to identify groups of items that are often occur together.

One of the most important and well researched techniques of data mining, ARM was first introduced in [1]. Association rules are widely used in various areas such as telecommunication networks, market and risk management, inventory control etc.

4. ARM ALGORITHMS

4.1 APRIORI ALGORITHM

The Apriori-based algorithms find frequent itemsets based upon an iterative bottom-up approach to generate candidate itemsets. An Apriori algorithm generally works on databases containing transactions. The algorithm works until no more frequent itemsets are found. After frequent itemsets are obtained they are used to generate association rules having confidence larger than or equal to minimum confidence which is specified by user

4.2 ECLAT ALGORITHM

It is also used to perform mining on itemsets. It is a depth first search algorithm using intersection of sets. The basic idea is to perform intersection on transaction id's to find the candidate itemset support avoiding the generation of those subsets that are not in the prefix tree.

4.3 FP GROWTH ALGORITHM

FP growth or frequent pattern growth algorithm uses a data structure (FP tree or prefix tree) to store the entire database in a compressed form. By storing the database in a tree like structure, the costly step of database scan is avoided. FP growth uses a divide and conquer approach which is a tree based frequent pattern mining method used to avoid costly process of candidate generation.

4.4 OPUS SEARCH

Opus is an efficient association rule mining that does not require constraints such as support or confidence. Initially it was used to find rules for a fixed items but gradually it has been extended to find rules for any items.

5. DEFICIENCIES IN APRIORI BASED ALGORITHM

The apriori like algorithms suffer from various deficiencies like too many scans of the transaction database when seeking frequent itemsets (after every iteration, the algorithm scans the whole database to find frequent itemsets), too large amount of candidate itemsets generated unnecessarily (large number of candidate itemsets are generated even though their count is less than minimum count and are then pruned after generation), the redundant generation of identical sub-itemsets and the repeated search for them in the database (itemsets like ab and ba are considered to be same but still they are generated), and so on.

6. PARALLEL APRIORI METHODS

To improve the performance of apriori based algorithms, many ways also have been proposed. Many ways are their to parallelize apriori based algorithms. They can be categorized into Count Distribution, Data Distribution and Candidate Distribution methods[7].

6.1 COUNT DISTRIBUTION

This method adapts a data parallel strategy which divides the database into horizontal partitions and then scanned independently for obtaining the local counts of all candidate itemsets on each process. The local counts are summed up after every iteration to obtain global count to find frequent itemsets.

6.2 DATA DISTRIBUTION

This method is helpful in using average main memory of machines in parallel by partitioning both the database and the candidate itemsets. As each candidate itemset is counted only one process, all processes have to exchange database partitions during each iteration for each process to get the global counts of the assigned candidate itemsets.

6.3 CANDIDATE DISTRIBUTION

This method also partitions candidate itemsets but selectively replicas instead of partition and exchanging of database transactions, so that each process can proceed independently.

The algorithms performance in almost similar fashion (time) that is for lower number of transactions because the number of itemsets created were low due to which the thread execution time was almost similar to the sequential execution time. But as we moved to higher number of transactions, the difference between execution time became more prominent. Those is because in the exponential increase in the number of itemsets created in various dimensions due to which the thread execution became more effective because of it's simultaneous executions. The algorithm has been compared only with the original apriori algorithm and not with other association rule mining algorithms.

7. PURPOSED WORK

The basic Apriori algorithm discussed above used in a very naive way of finding association among various data objects, by using frequent itemsets at each iteration and then finding the items having the count lower than the minimum count and then later removing them from the frequent itemset so that the next generation of the itemset does not contain any infrequent items relating to the database.

We have looked through the working of apriori data mining algorithm with all the previous work done in the field of

apriori algorithm and deficiencies that it had, Parallelizing the Apriori along with finding a new way to implement it so that when it is implemented on a database , it leads to lower read for generating frequent itemset along with fast executions

8. SUMMARY

For this sole purpose we have gathered the database in a file and created a numerical image of the database which is inform of a file in our algorithm because the read operations are faster in the files, We will develop an algorithm and that algorithm performed on to higher number of transactions, almost similar fashion (time) that is for lower number of transactions because the number of itemsets created were low due to which the thread execution time was almost similar to the sequential execution time. But as we moved to higher number of transactions, we can see the difference between execution time became. The algorithm would be developed and been compared only with the original apriori algorithm and not with other association rule mining algorithms.

9. REFERENCES

[1] M. S. Chen, J. Han, and P. S. Yu. Data mining: An overview from a database perspective. IEEE Trans. Knowledge and Data Engineering, 8:866-883, 1996.

[2] U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy. Advances in Knowledge Discovery and Data Mining. AAAI/MIT Press, 1996.

[3] W. J. Frawley, G. Piatetsky-Shapiro and C. J. Matheus, Knowledge Discovery in Databases: An Overview. In G. Piatetsky-Shapiro et al. (eds.), Knowledge Discovery in Databases. AAAI/MIT Press, 1991.

[4] J. Han and M. Kamber. Data Mining: Concepts and Techniques. Morgan Kaufmann, 2000.

[5] T. Imielinski and H. Mannila. A database perspective on knowledge discovery. Communications of ACM, 39:58-64, 1996.

[6] G. Piatetsky-Shapiro, U. M. Fayyad, and P. Smyth. From data mining to knowledge discovery: An overview. InU.M. Fayyad, et al. (eds.), Advances in Knowledge Discoveryand Data Mining, 1-35. AAAI/MIT Press, 1996.

[7] G. Piatetsky-Shapiro and W. J. Frawley. Knowledge Discovery in Databases. AAAI/MIT Press, 1991.

[8] An Effective Hash-Base Algorithm for Mining Association Rules. Jong Soo Park,* Ming Chen and Philip S . Yu, IBM Thomas J. Watson Research Center ,New York 10598.

[9] An Hash – Mine algorithm for discovery of frequent itemsets, Marek Wojciechowski , Maciej Zakrzewicz Institute of computer science, ul. Piotrowo3a Poland.

[10] An Efficient Algorithm for mining Association rules in Large databases , Ashok Savasere,Edward Omiecinski, Shamkant Navathe ,college of computing , Georgia Institute of technology, Atlanta , GA 30332 .

[11] A Fast Apriori implementation, infomatics Laboratory, Computer and Automation Research institute, Hungarian academy of sciences.

[12] Mining Large Itemsets for Association Rules ,Charu C. Aggarwal ,IBM research Lab.

[13] Mining association rules between sets of items in large databases, Rakesh Agarawal,Tomasz Imielinski*,Arun swami,IBM research lab.

[14] Fast algorithm for mining association rules,Rakesh Agarwal Ramakrishna Srikannt*,IBM research labs 650 Harry Road , San Jose , CA 95120.

[15] J Han, Y. Cai , and N,Cercone . Knowledge Discovery in database: An attribute – oriented approach. Proceeding of the 18 th International Conference on very large data bases, Page 547-559,august 1992