SUMMARY AND CONCLUSION

5. SUMMARY AND CONCLUSION

The widespread availability of economical communication medium has revolutionized the concept of communication, transmission and interaction. It offers new ways of business-to-business and business-to-customer transactions, new mechanisms for person-to-person communication, new means of discovering and obtaining information, services and products electronically. The volume of business and communication data increases daily and is expected to grow more in the forthcoming years.

Knowledge discovery is the process of automatically processing a huge repository of data to identify frequently occurring patterns, which can then be used to expand or improve a business. This process requires powerful algorithms which can explore and discover interesting patterns. Many data mining techniques use transactional database for this purpose and this process is termed as frequent pattern mining. A transaction is defined as a set of items purchased by a customer at the same time. A transaction database consists of a set of transactions. The transaction database consists of various types of data. For example, it can be market basket data, where a single transaction indicates the various purchases made by a customer or it can be web log file, where each entry gives information on the various web pages visited by a web user.

The frequent pattern algorithms perform the mining process in two phases. In the first phase, all frequent itemsets that satisfy the user specified minimum support are generated and in the second phase uses these frequent itemsets in order to discover all the association rules that meet a confidence threshold. Out of the two phases, the first problem is more computationally expensive and less straightforward. In this research work, this problem is considered and two solutions which optimize the performance of frequent pattern by reducing the I/O time complexity during frequent pattern mining.

Another concern of database transaction processing systems is the overwhelming size that has to be handled by the knowledge discovery algorithms. The current need of the data mining field is to develop way of representation which reduces this size at the same time maintains all the important and relevant data needed to extract the desired knowledge. In other words, techniques which can produce a compact representation of the existing database and which can be later be used for extracting patterns is the current need of the business market. This research analyzes algorithms that produce compact databases for knowledge discovery from large transaction databases like market basket database and web log databases. From these compact representations, association rule mining is applied to mine frequent patterns.

Mining frequent patterns is one of the fundamental and essential operations in many data mining applications, such as discovering association rules. Many algorithms have been proposed to improve the performance of mining frequent patterns from transaction databases. Prominent among these proposals are the Apriori and FP-Growth algorithms. In this research, two variants of Apriori and FP-Growth algorithms, namely, CT-Apriori and CT-PRO are compared and their performances are analyzed.

The CT-Apriori algorithm uses a compact tree structure, called CT-tree, to compress the original transactional data. The tree representation allows the CT-Apriori algorithm, which is revised from the Apriori algorithm, to generate frequent patterns quickly by skipping the initial database scan and reducing a great amount of I/O time per database scan.

The CT-PRO algorithm uses a compact tree structure called CFP-Tree, which is more compact than the FP-Tree of the FP-Growth algorithm. An algorithm called CT-PRO is used to mine frequent patterns from CFP-Tree. The CT-PRO algorithm divides the CFP-Tree into several projections represented by CFP-Trees. Then CT-PRO conquers the CFP-Tree for mining all frequent patterns in each projection.

To evaluate the performance of both the algorithms, several experiments were conducted with two datasets, namely, synthetic dataset and web log dataset. Empirical evaluations showed that both approaches are effective, efficient and promising. The storage space requirement as well as the mining time was decreased dramatically on both synthetic and web log databases, while using both algorithms. Comparison with Apriori and FP-Growth showed that the performance of CT-PRO was better in all scenarios.

The compression efficiency in terms of storage size required showed a gain of 7.65% over CT-Apriori. Similarly, while considering the number of transactions, CT-PRO was more efficient (7.18%) than CT-Apriori.

The execution speed results also indicated that the CT-PRO algorithm was the fastest among all the algorithms. The efficiency achieved in terms of execution speed on average by CT-PRO algorithm over CT-Apriori was 4.91% while it was 14.54% and 10.60% for Apriori and FP-Growth algorithm. Similar trend was also observed for experiments with web log data.

All these results point CT-PRO is the right candidate for generating a compact version of the original transaction database, which is small in size and which performs frequent pattern mining in a fast and efficient manner.

RECOMMENDATION FOR FUTURE RESEARCH WORK

The following suggestions can be incorporated in future to attain a better frequent pattern mining algorithm based on CT-PRO and CT-Apriori.

1. The compact transaction database can be further compressed by any existing lossless data compression technique for storage and network transmission purposes.

2. Future research can combine the CFP-Tree with CT-Tree candidate generation method to perform pattern mining more effectively.