International Journal of Arts & Education Research

DISTRIBUTED AND IMPROVISED UP-GROWTH APPROACH FOR UTILITY BASED MINING

A.Pio Sajin*[1], R. Aswini[2]

[1]Asst. Prof, Computer Science and Engineering Department, Sathyabama University, Chennai, India.

[2]M.E Student, Computer Science and Engineering, Sathyabama University, Chennai, India.

ABSTRACT

Data mining is the well known methodology in extracting hidden predictive information from large databases. Number of algorithms like k-means, Apriori, FP-growth and Naïve Bayes has been emerged for mining patterns based on different perspectives. Numbers of researches are carried out in improvising these algorithms. In this system UP-growth, one of the popular algorithms in mining high utility itemset is considered and improvised under different constraints. The Node utility (NU) and Minimum Node Utility (MNU) are the aspects considered as the key term in the proposed system for mining high utility itemset from Transactional database. However, working the system as a sequential process will be time consuming. The proposed system overcomes the concern by introducing the system under distributed environment.

Keywords: Distributed environment, Node utility, Minimum Node utility.

INTRODUCTION

Data mining is an emerging methodology that extracts hidden predictive information from large databases. A number of algorithms have been proposed by the researchers under the field of data mining. The technology hence has got rapid growth in past few years. However researches are being carried out still to improve these algorithms in various perspectives. This paper proposes new ideas to improve the efficiency of a well known algorithm, UP-Growth.

The Improvised UP-Growth approach is an efficient algorithm for discovering high utility itemset from Transactional database. Two aspects of Transactional databases are internal utility and external utility. The product of internal utility and external utility is defined as utility of an itemset. The utility value of an itemset greater than user specified threshold value is called high utility itemset. The proposed system improvises high utility itemset mining by preceding the Utility Pattern-Growth (UP-Growth) approach including Minimum Node Utility (MNU) to the process. Thereby reducing the execution time compared with existing system. Time taking aspect in this entire mining methodology is the complexity of tree construction and computation. These complexities are simplified in the proposed system by partitioning the transaction level wise and assigning each as a sub process and producing them to a distributed environment.

LITERATURE SURVEY

Vincent S. Tseng, Bai-En Shie, Cheng-Wei Wu, fellow and Philip S. Yu proposed two different types of algorithm namely utility pattern growth (UP-Growth) and Utility Pattern

Growth+ (UP-Growth+) for mining high utility itemsets from transactional databases. Mining refers to extracting the relevant or the required details from the database. Itemsets are combination of items. Here this system refers to retrieving the high utility itemsets from the transactional database. Many algorithms have been proposed earlier but they have problems producing more candidate itemsets which degrades the performance. In this paper they have proposed two different types of algorithm to overcome the disadvantages. Here in the above two algorithms the concept of pruning the database is considered. First the itemsets are stored in a tree based structure and this tree is called UP-tree which reduces the number of scan count to two. During the first scan of database, the Transaction Utility (TU) and Transaction Weight Utilitiy(TWU) are computed for removing the unpromising item. Then the Transactions are inserted into the UP-tree by the second scan of original database. This tree structure is similar to the B+ trees. Especially the UP-Growth+ algorithm has shown the great difference in the runtime and in the performance. But these algorithms are mined under sequential environment and comparatively less efficient. It is highly complex process.

B.E Shie, H.-F.Hsiao, V.S.Tseng, and P.S.Yu developed the two tree based data structure named $UMSP_{DFG}$ (Mining high utility mobile sequential pattern with Depth First Generation Strategy) and $UMSP_{BFG}$ (Mining high utility mobile sequential pattern with Breadth First Generation Strategy) as their algorithm for Mining High Utility Mobile Sequential Patterns in Mobile Commerce Environments [2]. This system refers to retrieving the high utility mobile sequential patterns from mobile environments. Mobile Transaction Sequence Tree (MTS-Tree) is used in these algorithm to maintain the information such as paths, location, items and utilities in mobile transaction database. After one scan of original database, the MTS-tree construction is completed. Now $WUMSP_S$ (Weighted Utilization Mobile Sequence patterns) are generated from MTS-tree by using depth first generation strategy. To find $UMSP_S$ from the set of $WUMSP_S$, An additional scan of database is performed. The performance of mining will become better by using this process but it also consumes more time and the number of $WUMSP_s$ is quite large. To overcome this problem they proposed $UMSP_{BFG}$ Algorithm. Among the algorithm, $UMSP_{BFG}$ performance is best and it effectively improves the performance of mining process but it consumes more memory space. This is the first paper to combine mobility pattern mining and utility mining and work on mining high utility mobile sequential patterns.

K.Sun and F.Bai introduced a link-based measure called w-support. Normally during the process of mining we ignore the difference between the transactions and the databases that has only binary attributes does not supports weighted association mining. Here in this system they are taking into account the quality of the transaction and the links established between them. Here they use a new technique w-support that does not use pre-assigned weights. Here the weight of the transaction is calculated by the algorithm used to derive the weights and these weights are used for mining. The outcome from this type of algorithm is more cost effective as they use the link based model. It cannot improve the mining performance instead of using downward closure property along with weighted association rules.

A.Erwin, R.P.Gopalan, and N.R.Achuthan proposed an algorithm called CTU-PROL for mining high utility itemsets from Large Datasets. High utility itemsets mining which mines

for itemsets with high profits or quantities is totally different from the frequently used itemsets mining which mines for itemsets that are more frequently being used. Mining high utility itemsets are more challenging than that of the frequently used itemsets mining which is mainly due to the fact that they lacks in anti-monotone property of frequent itemsets. But ever since after the Transaction Weighted Utility (TWU) that has the anti-monotone property which is proposed by the researchers in the recent days the challenge has become little bit easier. There is also a disadvantage of using this TWU that it requires larger space. Here in this system they use TWU along with the compact utility pattern tree data structure as their algorithm. Here they uses the parallel projection scheme, this scheme is used as they requires large disc spaces so that in case of dealing with the large datasets when the main memory storage is inadequate they uses the disk storage as the parallel projection. It has also proved to be better in the experimental evaluations. It consumes more processing time and it generates a large number of itemsets so the mining performance is degraded.

H.Yao, H.J.Hamilton, and L.Geng developed a unified framework for utility based measures for mining itemsets. Here in this system the main concept used is the pattern of itemsets and as these patterns of itemsets are more used in reaching the goal of the person. Here they uses the utility based measures of the pattern of the itemsets and uses the same in their concept of data mining. All the utility based patterns are studied in detail first and then all of them are combined to form a single unified utility based function. In the existing models we use different type of representation of the itemsets for the same datasets, but here we use the same representation as they are grouped under different three levels like item level, transaction level and cell level. Every utility pattern has certain mathematical properties used in them and these are all combined and the separate unified mathematical function to be used is also derived from the above mathematical properties. But to derive the separate mathematical expression and the pattern for all the utility based itemsets is made more complex.

Ying Liu,Wei-Keng Liao and Alokchoudary proposed an algorithm named Two-phase algorithm for mining high utility itemset. It can trim the candidate itemsets effectively and also makes the utility calculation is very simple. The phase-1 of this algorithm have defined the Transaction-weighted utilization mining method, which is the sum of transaction utilities of all transaction containing the particular itemset by using “Transaction-weighted downward closure property”. The less time is spent in phase-1. In phase-2, by the first scan of original database, the high utility itemsets are computed from high transaction weighted utilization which are identified in phase-1. Phase-2 requires only one extra database scan for selecting the high utility itemsets. Common Count Partitioned Database (CCPD) strategy are used for parallelize the Two-phase algorithm on shared memory multi-process architecture. The disadvantage of this system is that it suffers from test methodology and level wise candidate generation and it requires multiple databases scans to select the high utility itemsets and also generates more number of candidates.

J. Han, J. Pei, and Y. Yin proposed two types of algorithm namely FP-tree algorithm which use the FP-growth process. There are many types of databases. Many researches are in progress for generating frequent patterns in data mining. Here the objective of this system is

to generate frequent patterns without generating candidate sets. This is done without candidate sets because the candidate set generation is too costly. Mostly the previous studies and the algorithms use the Apriori like candidate set generation and cause the process costly as they are of long patterns .Here they used these algorithm to mine the complete set of frequent pattern. This involves three techniques are as follow, 1. They use the technique of compressing the larger database which makes the database more condensed. 2. Here in the generation of the tree structure they uses the fragment growth which reduces the size of the tree thus causing the tree structure to be less deeper and less number of candidate sets. 3. At last they uses the most pruning divide and conquer method which splits the larger mining ask into smaller tasks and finding the solution for each task. All the solutions of the tasks are then combined to produce the required solution which will be the confined solutions. Even though the FP-growth achieves a better performance, it consumes more memory and FP-tree may not fit into the memory space.

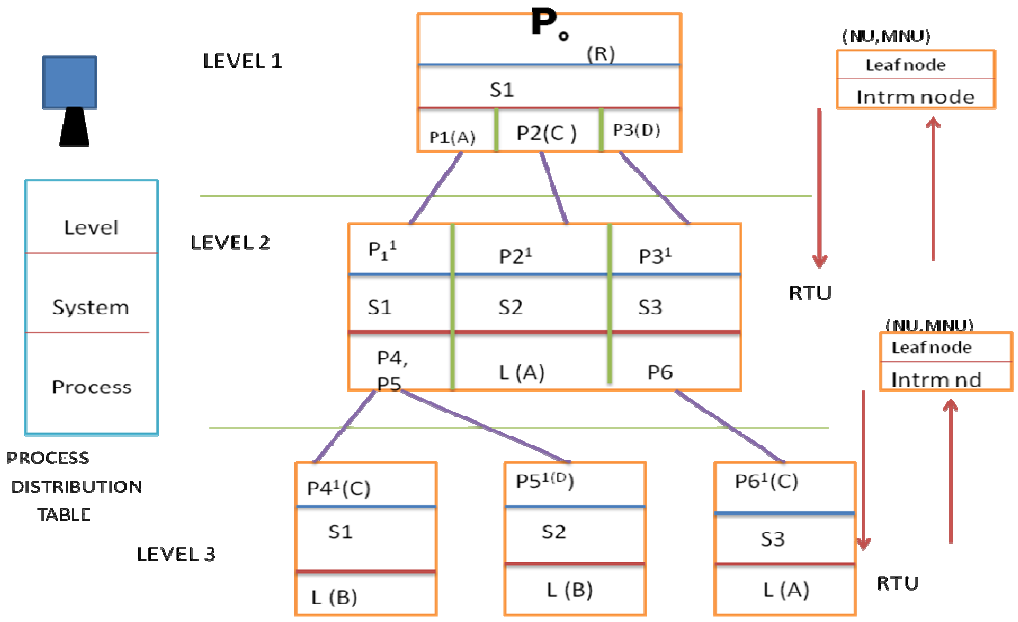
R. Agrawal and R. Srikant proposed a Fast Algorithms for Mining Association Rules. Here in this system the main factor for consideration is the association rules between the different items in the large database. These are mostly used in the sales transactions and the proposal of this paper is to reduce the complexity and improve the efficiency and performance in mining the required itemsets. They are implementing two new algorithms which seem to outperform the existing algorithms. They do combine these two algorithms to produce a new hybrid algorithm which is called as the AprioriHybrid. This AprioriHybrid proves to be better than all other algorithms used by proving themselves in reducing the transaction size and number of items in the database. Here there is no route for the items that has no association rules and here in this algorithm they are not considering the quantities items brought in the transaction. it also sometimes has many association rules for one particular item.

PROPOSED SYSTEM

The framework of proposed system consists of following steps: 1) Compute Transaction Utility (TU) of each transaction and Transaction Weight Utility (TWU) of each single item to identity the unpromising item 2) construct Utility Pattern Tree (UP-Tree) under Distributive environment 3) generate Potential High Utility Itemsets (PHUI) from UP-Tree by Using Improvised Utility Pattern-Growth approach 4) Identify high utility itemsets from Potential High Utility Itemsets (PHUI) sets.

The first step is to identify the unpromising items. By using TWU we can identify the unpromising items. The unpromising item holds the meaning that, if its TWU is less than a user-specified threshold value, otherwise it is called a promising item. Now, we should remove the unpromising items from transaction.

After removing the unpromising items from all transaction, we have a Reorganized Transaction Utility (RTU) for each transaction.



Next step is to insert the transaction into UP-Tree under Distributive environment in order to reduce the complexity. For instance, if the dataset contains more than 50 transactions and each transaction contains more than 30 items means, it will be difficult to construct tree and also takes more time to compute Node Utility (NU) and Minimum Node Utility (MNU) for each item and generate more number of PHUI.

To minimize the complexity during the construction of UP-Tree by split the tree by some level and distribute that portion of tree to the new system for computing node utility of each items. Likewise all other levels of tree are distributed to other system.

Intermediate System takes responsible for assigning the system for each process and it also maintains Process Distribution table. It contains level, system and process of UP-Tree.

After constructing level1, now the system1 is idle, it tells the intermediate system about its status. After that, Intermediate system will assign the system1 to next level. At the same time process P1 send their RTU to next level. By using RTU, NU and MNU of each item is computed.

Next step is to generate PHUI by using Improvised UP-Growth approach. The algorithm starts from bottom entry of the tree. Select the item in leaf node that should not be the intermediate node of any other transaction. For that, all the levels send their leaf node and intermediate node to the previous level.

In the above figure, last level send A and B as a leaf node and C and D as the intermediate node to previous level. Now, level2 checks leaf node of level3 with its intermediate node. If the item in leaf node of level 3 is present in intermediate node of level2 means it will remove that node from leaf node table. Likewise, all other levels update the leaf node table.

Finally, it reached level1. Now, we calculate NU for that leaf node (sum of node utility of all transaction containing the selected leaf node). If the NU of leaf node is greater than threshold

value means, it will be consider as PHUI, otherwise, it is unpromising item. Likewise generate PHUI for all items.

Final step is to identify the high utility itemsets from PHUI sets.

ADVANTAGES

- Conversion of sequential to distributed execution provides with reduced execution time.
- Improvised approach of UP-Growth algorithm reduces number of scans which in turn reduces the process time further.
- System complexity is reduced as the process is been split-up.
- Potential High Utility Itemset is obtained in greater detail in comparison.
- More efficient in fields where large transactions are processed.

CONCLUSION

A system to mine high utility itemset from a huge transactional database is proposed. UP-Growth approach is improvised by considering minimum node utility as a key constraint in producing the PHUI in more accurate. From the PHUI obtained, high utility itemset is generated further. This increases the accuracy in high utility data items mined. As above discussed, the concept is too complex in order of implementation. Hence, the entire process is split up into sub process based on transactional level and these partitions are distributed as the process to different processors. This introduces our system to distributed environment, which ensures reduced complexity and improved efficiency of mining.

REFERENCES

[1] Tseng VS, Shie BE, Wu CW, Yu PS. Efficient Algorithms for Mining High Utility Itemsets from Transactional Databases. IEEE transactions on knowledge and data engineering 2013; 25(8).

[2] Shie BE, Hsiao HF, Tseng VS, Yu PS. Mining HighUtility Mobile Sequential Patterns in Mobile Commerce Environments. Proc. 16th Int'l Conf. Database Systems for Advanced Applications (DASFAA '11) 2011 ; 6587 : 224-238.

[3] Sun K, Bai F. Mining Weighted Association Rules without Preassigned Weights. IEEE Trans. Knowledge and Data Eng 2008; 20(4): 489-495.

[4] Erwin A, Gopalan RP, Achuthan NR. Efficient Mining of High Utility Itemsets from Large Data Sets. Proc. 12th Pacific-Asia Conf. Advances in Knowledge Discovery and Data Mining (PAKDD), 2008; 554-561.

[5] Yao H, Hamilton HJ, Geng L. A Unified Framework for Utility-Based Measures for Mining Itemsets. Proc. ACM SIGKDD Second Workshop Utility-Based Data Mining 2006; 28-37.

[6] Liu Y, Liao W, Choudhary A. A Fast High Utility Itemsets Mining Algorithm. Proc. Utility-Based Data Mining Workshop, 2005.

[7] Han J, Pei J, Yin Y. Mining Frequent Patterns without Candidate Generation. Proc. ACM-SIGMOD Int'l Conf. Managementof Data, 2000; 1-12.

[8] Agrawal R, Srikant R. Fast Algorithms for Mining Association Rules. Proc. 20th Int'l Conf. Very Large Data Bases (VLDB), 1994; 487-499.

