International Journal of communication and computer Technologies, ISSN: 2278-9723
Available at http://www.ijccts.org

Survey on High Utility Itemset Mining

[1]Yamine.R.T, [2]Nithya.N.S

[1]PG Student ,[2]Assistant Professor, Department of Computer Science and Engineering,

K.S.R.College of Engineering, Tiruchengode

Received: 15-06-2015, **Revised:** 28-07-2015, **Accepted:** 30-10-2015, **Published online:** 09-12-2015

ABSTRACT

Utility-based data mining is a new research area involved in all types of utility factors in data mining processes. The main objective of Utility Mining is to categorize the itemsets with highest utilities, by making an allowance for other user preferences such as quantity, profit and cost.Mining high utility itemsets from a transactional database refers to the encounter of itemsets with high utility like profits. It experience the problem of producing a large number of candidate itemsets used for high utility itemsets. Such a large number of candidate itemsets destroys the mining concert in terms of execution time and space requirement

Keywords - Data Mining, Association Rule Mining, Systolic tree mechanism, Utility-based mining

I. INTRODUCTION

Data mining is the process of realizing useful knowledge from a group of data. Association Rule Mining (ARM) is an important data mining technique which is used to discover the rules/patterns among items in a huge database. Association rule mining with itemset frequencies are used to mine itemset interactions. Frequent pattern mining algorithms are designed to find frequently occurring sets in databases. Memory and run time requests are very high in frequent pattern mining algorithms. A transaction database consists of two features such as internal and external utility. Capacity of a product in a particular transaction is called the internal utility and the profit rate of a product is called external utility. The utility of itemset is well-defined as the product of external utility and internal utility.

Utility of Itemset (U) = external utility * internal utility

In many areas of business like inventory, retail, etc. decision making is very significant. In a transaction database each item is represented by a binary value, without allowing for its profit.

In several applications like cross-marketing in retail stores, website click- stream analysis, online e-commerce management and finding the dynamic pattern in bio- medical applications High utility mining are widely used.

Example

Consider, a transaction database representing the sales data and the profit associated with the sale of every unit of the items.

Table I

Transaction Database

TID	Item sold in transaction		
	Item A	**Item B**	**Item c**
T1	0	18	0
T2	6	0	0
T3	0	1	1
T4	4	8	2
T5	2	4	5
T6	0	2	3
T7	10	0	0
T8	1	25	6
T9	0	0	1
T10	6	2	0

Table II

Unit Profit Associated with items Item

Name	Unit Profit (in INR)
Item A	10
Item B	3
Item C	5

Let us consider the itemset BC.

Since, there are only 3 transactions T4, T5 and T8 which contains BC itemset out of 10 transactions.

So, support for itemset BC is
Support (BC) = 3 / 10 * 100 = 30 %

In T4 transaction, units gain by item B and C are 2 and 4 respectively, the profit earned from the sale of itemset BC in T4 transaction is given by,

Profit (BC, T4) = 2 * profit (B) + 4 * profit (C)
= 2*5 + 4*10
= 50

Since BC appears in transactions T4, T5 and T8, So, total profit of itemset BC is given by

profit (BC) = profit(BC,T4) + profit(BC,T8) + profit(BC,T5)
= (2*5+4*10) + (6*5+1*10) + (5*5+2*10)

= (10+40+ (30+10)) + (25+20)
= 50 + 40+ 45
= 135

Similarly, we can calculate the support values for the dissimilar itemsets and also the profit obtained through the sale of those itemsets by all the ten transactions as indicated in table III.

Table III
Profit and Support for All Itemsets

Itemset	Support (%)	Profit
A	60	290
B	70	180
C	60	90
BC	30	135
AC	40	247
ABC	30	246

If we consider minimum support 50%, then we can detect that there are only 4 itemsets A, B, C and AC which have the support greater than the threshold value (min_sup). So, they succeed as frequent itemsets. But if we consider it profit wise then we can find out of 4 most profitable itemsets A, AB, CB, ABC only A and BC are frequent itemsets. Itemsets AB and ABC are not frequent but still they fetch the more profit than further itemsets. As we can see from table III, single unit of item A fetch more profit than single unit of Itemset A and B. From this Example, we can explain frequent Itemset mining may not always satisfy profit wise requirements of sales manager .In this instance, the support (%) attribute of the itemsets reflects the statistical correspondence not the semantic significance of items.

II. DATA FLOW DIAGRAM

The following Diagram shows the sequence process of calculating and demonstrating a high utility Itemsets. From this Fig. , the comparison of frequent Itemset with given threshold value and by considering a profit, gives High utility Itemsets.

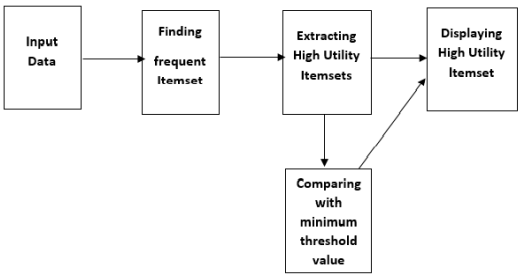

FIG. DATA FLOW DIAGRAM

III. LITERATURE SURVEY

J Hu et al established an algorithm for frequent item set mining that identify high utility item combinations. The objective of this algorithm is to find sections of data, defined through groupings of some items (rules), which satisfy certain circumstances as a maximize and group a predefined detached function. The high utility pattern mining problem reflected is different from former approaches, as it conducts rule discovery with respect to individual attributes as well as with high opinion to the complete criterion for the mined set, attempting to find groups of such patterns that together contributes to the most to a predefined detached function [1].

Y-C. Li, J-S. Yeh and C-C. Chang suggested an isolated item discarding strategy (IIDS). In this paper, they learned high utility itemsets and also compact the number of candidates in every database scan. They rescued efficient high utility itemsets by the mining algorithm called FUM and DCG+. In this technique they exhibited a better performance than all the preceding high utility pattern mining technique. However, their algorithms still suffer with the problem of test problem and level wise generation of apriori and it require multiple database scans [2].

Liu Jian-ping, Wang Ying, Yang Fan-ding et al recommended an algorithm called tree based incremental association rule mining algorithm (Pre-Fp). It is established on a FUFP (fast update frequent pattern) mining method. The major goal of FUFP is the re-use of previously extracted frequent items while affecting onto incremental mining. The advantage of FUFP is that it reduces the quantity of candidate set in the updating procedure. In FUFP, all links are bidirectional whereas in FP-tree, relations are only unidirectional. The advantage of bidirectional is that it is easy to add, remove the child node without much rebuilding. The FUFP structure is used as a input to the pre-large tree which gives positive count difference consistently small data is added to original database. It deals

with few changes in database in case of introducing new transaction. In this paper the algorithm classifies the items into three categories: frequent, pre-large and infrequent. Pre-large itemsets has two supports threshold value i.e. upper and lower threshold. The drawback of this approach is that it is time consuming [3].

Ahmed CF, Tanbeer SK, Jeong BS et al established HUC-Prune. In the existing high utility pattern mining it generate test methodology and a level wise candidate generation to maintain the candidate pattern and they need several database scans which is directly reliant on the candidate length. To overcome this, they proposed a novel tree based candidate pruning technique named HUC-tree, (high utility candidate tree) which captures the important utility information of transaction database. HUC-Prune is completely independent of high utility candidate pattern and it requires three database scans to

calculate the consequence for utility pattern. The drawback of this approach is that it is very difficult to endure the algorithm for larger database scan regions [4].

Shih-Sheng Chen et al (2011) suggested a method for frequent periodic pattern using multiple minimum supports. This is an effective approach to find frequent pattern because it is based on multiple minimum threshold support based on real time event. All the items in transaction are ordered according to their minimum item support (MIS), and it does not hold download closure property, as an alternative it uses sorted closure property based on ascending order. Then PFP (periodic frequent pattern) algorithm is realistic which is similar as that of FP- growth where conditional pattern base is used to determine frequent patterns. This algorithm is more efficient in terms of memory space, thereby reducing the number of database scans [5].

Chowdhury Farhan Ahmed, Syed Khairuzzaman Tanbeer, Byeong-Soo Jeong, Young-Koo Lee, Ho- Jin Choi et al proposed a Single-pass interactive and incremental mining for finding weighted frequent patterns. The existing weighted frequent pattern (WFP) mining cannot be applied for interactive and incremental WFP mining and also for stream data mining because they are based on its require multiple database scans and static database . To overcome this, they proposed two novel tree structures IWFPTWA (Incremental WFP tree based on weight ascending order) and IWFPTFD (Incremental WFP tree based on descending order) and two new algorithms IWFPWA and IWFPFD for incremental and interactive mining using a single database scan. IWFPFD confirms that any non-candidate item cannot appear before candidate items in any branch of IWFPTFD and thus speeds up the prefix tree. The weakness of this approach is that large memory space, time consuming and it is very difficult to support the algorithm for larger databases [6].

IV. CONCLUSION

In this paper, a dynamic and distributed method is suggested to generate complete set of high utility itemsets from large databases. Mining high utility itemsets from databases represents to finding the itemsets with high profit. In distributed, it arranges the unpromising items recognized on the minimum utility itemsets from transactions database. This approach creates distributed surroundings with one master node and two slave nodes scans the database once and counts the reality of each item. The large database is distributed to all slave nodes. The global table has the final resultant. Incremental Mining Algorithm is used where continuous updating goes on appearing in a database. Finally incremental database is rearranged and the high utility itemsets is discovered. Hence, it provides faster execution, that is reduced time and cost.

REFERENCES

[1] Agrawal, R., Imielinski, T., Swami, A.: Mining Association Rules between Sets of Items in Large Database. In: ACM SIGMOD International Conference on Management of Data (1993) .

[2] Yao, H., Hamilton, H.J., Buzz, C. J., “A Foundational Approach to Mining Itemset Utilities from Databases”, In: 4th SIAM International Conference on Data Mining, Florida USA (2004).

[3] “A Two-Phase Algorithm for Fast Discovery of High Utility Itemsets”, Ying Liu, Wei-Keng Liao, and Alok Choudhary, Northwestern University, Evans

[4] “CTU-Mine: An Efficient High Utility Itemset Mining Algorithm Using the Pattern Growth Approach” In: Seventh International Conference on Computer and Information Technology (2007).

[5] J.Hu, A. Mojsilovic, ―High utility pattern mining: A method for discovery of high utility itemsets‖, in: pattern recognition. PP: 3317-3324, 2007.

[6] A.Erwin, R.P. Gopalan, and N.R. Achuthan, “Efficient Mining of High Utility Itemsets from Large Datasets”, T. Washio et al. (Eds.): PAKDD2008, LNAI 5012, pp. 554–561, 2008. © Springer-Verlag Berlin Heidelberg 2008.

[7] Y.-C. Li, j,-s. Yeh, and C.-C. Chang, ―Isolated Items Discarding Strategy for Discovering High Utility Itemsets, ‖ Data and Knowledge engg. pp: 198-217, 2008.

[8] Liu Jian-Ping, Wang Ying Fan-Ding, ‖Incremental Mining algorithm Pre-FP in Association Rule Based on FP-tree‖, Networking and Distributed Computing, International Conference, pp: 199-203, 2010.

[9] Ahmed CF,Tanbeer SK,Jeong B-S, Lee Y-K (2011) ―HUC-Prune: An Efficient Candidate Pruning Technique to mine high utility patterns‖ Appl Intell PP: 181–198, 2011.

[10] Shih-Sheng Chen, Tony Cheng-Kui Huang, Zhe-Min Lin, ―New and efficient knowledge discovery of partial periodic patterns with multiple minimum supports‖, The Journal of Systems and Software 84, pp. 1638–1651, 2011, ELSEVIER.

[11] Chowdhury Farhan Ahmed, Syed Khairuzzaman Tanbeer, Byeong-Soo Jeong, Young-Koo Lee a,Ho-Jin Choi(2012) ―Single-pass incremental and interactive mining for weighted frequent patterns‖, Expert Systems with Applications 39 pp.7976– 7994, ELSEVIER 2012.

[12] “UP-Growth: An Efficient Algorithm or High Utility Itemset Mining ”, Vincent S. Tseng, Cheng-Wei Wu, Bai-En Shie, and Philip S. Yu. University of Illinois at Chicago, Chicago, Illinois, USA, 2010.

[13] Mengchi Liu Junfeng Qu, “Mining High Utility Itemsets without Candidate Generation”, 2012.

[14] “FHM: Faster High-Utility Itemset Mining using Estimated Utility Co-occurrence Pruning”, Philippe Fournier-Viger1, Cheng-Wei Wu 2014.

[15] Smita R. Londhe,, Rupali A. Mahajan,, Bhagyashree J. Bhoyar,”Overview on Methods for Mining High Utility Itemset from Transactional Database”, International Journal of Scientific Engineering and Research (IJSER), Volume 1 Issue 4,December2013