An Efficient Methodology For Mining Utility Items From Large Data Sets

Sonali Pawar, Prof. Hemant Gupta
[1, 2] Dept of Computer science
[1, 2] RIT Indore

Abstract- *Data Mining, also known as Knowledge Exploration in Database, is one of the new research areas that have arisen as a response to the tsunami data or data flood that the planet is now facing. Developing techniques which can help humans discover useful patterns in large data has taken up the challenge. Utility mining is one such important technique. Frequent item set mining works to discover item set that is regularly displayed in the transaction database, which can be discovered on the basis of specific item set support and trust value. Using the concept of frequent itemset mining as a basis, many researchers have also proposed different new concepts of utility based itemset mining. Based on a parameter called weighted transactional utility, our basic goal in this research work is to discover candidate item and itemset. We also propose a new algorithm for the mining of high utility items set. The new algorithm outperforms earlier algorithms in computation time.*

Keywords- Data Mining, KDD Process, Minimum Utility, High Utility Mining, Minimum Utility.

I. INTRODUCTION

Data mining [1] has become an essential technology for businesses and researchers in many fields, the number and variety of applications has been growing gradually for several years and it is predicted that it will carry on to grow. A number of the business areas with an early embracing of DM into their processes are banking, insurance, retail and telecom. More lately it has been implemented in pharmaceutics, health, government and all sorts of e-businesses.

There is a huge amount of data available in the Information Industry[2]. This data is of no use until it is converted into useful information. It is necessary to analyze this huge amount of data and extract useful information from it. Data mining also involves other processes such as Data Cleaning, Data Integration, Data Transformation, Data Mining, Pattern Evaluation and Data Presentation. Once all these processes are over, we would be able to use this information in many applications such as Fraud Detection, Market Analysis, Production .The overall goal of data mining process is extract information from data set and transforms it into understanding structure for further use.

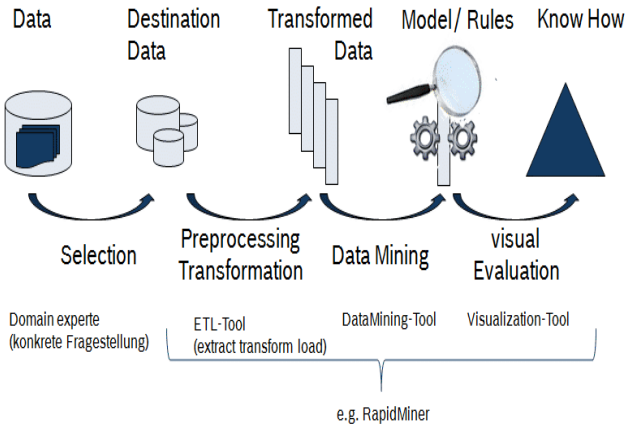

Fig. 1. The process of knowledge discovery in databases [1]

Frequent itemset mining[3][4] and high utility itemset mining, these two concepts used here. Frequent pattern mining is a popular problem in data mining, which consists in finding frequent patterns in transaction databases. The goal of frequent itemset mining is to find **frequent itemsets.** . In frequent itemset mining, there is a well-known property of the frequency (support) of itemsets that states that given an itemset, all its supersets must have a support that is lower or equal. Many popular algorithms have been proposed for this problem such as Apriori, FP Growth. These algorithms takes as input a transaction database and a parameter "**minsup**" called the **minimum support threshold.** This algorithm is very powerful to prune the search space because if an itemset is infrequent then we know that all its supersets are also infrequent. In high utility itemset mining there is no such property but it has some important limitations. To address these limitations, the problem of frequent itemset mining has been redefined as the problem of **high-utility itemset mining.** It cannot satisfy the requirement of users who desire to discover item sets with high utilities such as high profits .The problem of **high-utility itemset mining [5]** is to find the itemsets (group of items) that generate a high profit in a database, when they are sold together. The user has to provide a value for a threshold called "**min_util**" (the minimum utility threshold). An item set is called high utility item set (HUI) if its utility is no less than a user-specified minimum utility

threshold min_util. High utility itemsets (HUIs) mining is an emerging topic in data mining.

II. RELATED WORK

[6]Alva Erwin, Raj P. Gopalan, and N. R. Achuthan, proposed CTU-PROL algorithm for efficient mining of high utility itemsets from large datasets[6]. This algorithm finds the large TWU items in the transaction database. If data sets is too large to be held in main memory, the algorithm creates subdivisions using parallel projections and for each subdivision, a *Compressed Utility Pattern Tree (CUP-Tree)* is used to mine the complete set of high utility itemsets. If the dataset is small, it creates a single *CUP-Tree* for mining high utility itemsets.

[7]Shankar S., Purusothaman T., Jayanthi, S., suggested a novel algorithm for mining high utility itemsets[7]. This fast utility mining (FUM) algorithm finds all high utility itemsets within the given utility constraint threshold. The proposed FUM algorithm scales well as the size of the transaction database increases with regard to the number of distinct items available.

[8]R. Chan, Q. Yang, and Y. Shen, suggested mining high utility itemsets[8]. They proposed a novel idea of top-K objective-directed data mining algorithm, which mines the top-K high utility closed patterns that directly support a given business objective. To association mining, they add the concept of utility to capture highly desirable statistical patterns and present a levelwise itemset mining algorithm. They develop a new pruning strategy based on utilities that allow pruning of low utility itemsets to be done by means of a weaker but antimonotonic condition.

[9]Ramaraju C., Savarimuthu N., proposed a conditional tree based novel algorithm for high utility itemset mining[9]. A novel conditional high utility tree (CHUT) compress the transactional databases in two stages to reduce search space and a new algorithm called HU-Mine is proposed to mine complete set of high utility item sets.

[10]Y. Liu, W. Liao, and A. Choudhary, proposed a fast high utility itemsets mining algorithm [10]. They are present a Two-Phase algorithm to efficiently prune down the number of candidates and can precisely obtain the complete set of high utility itemsets. First phase proposes a model that applies the “transaction-weighted downward closure property” on the search space to expedite the identification of candidates. Second phase performs one extra database scan to identify the high utility itemsets.

[11]Adinarayanareddy B., O. Srinivasa Rao, MHM Krishna Prasad, suggested improved UP-Growth high utility itemset mining[11]. The compact tree structure, Utility Pattern Tree i.e. UP-Tree, maintains the information of transactions and itemsets and avoid scanning original database repeatedly. UP-Tree scans database only twice to obtain candidate items and manage them in an efficient data structured way. Applying this UP-Tree to the UP-Growth algorithm takes more execution time for Phase II. Hence they presents modified algorithm aiming to reduce the execution time by effectively identifying high utility itemsets.

[12]Adinarayanareddy B., O. Srinivasa Rao, MHM Krishna Prasad, suggested improved UP-Growth high utility itemset mining[12]. The compact tree structure, Utility Pattern Tree i.e. UP-Tree, maintains the information of transactions and itemsets and avoid scanning original database repeatedly. UP-Tree scans database only twice to obtain candidate items and manage them in an efficient data structured way. Applying this UP-Tree to the UP-Growth algorithm takes more execution time for Phase II. Hence they presents modified algorithm aiming to reduce the execution time by effectively identifying high utility itemsets.

[13] Efficient discovery of frequent itemsets in large datasets is a crucial task of data mining. In recent years, several approaches have been proposed for generating high utility patterns, they arise the problems of producing a large number of candidate itemsets for high utility itemsets and probably degrades mining performance in terms of speed and space. Recently proposed compact tree structure, viz. , UP-Tree, maintains the information of transactions and itemsets, facilitate the mining performance and avoid scanning original database repeatedly. In this paper, UP-Tree (Utility Pattern Tree) is adopted, which scans database only twice to obtain candidate items and manage them in an efficient data structured way. Applying UP-Tree to the UP-Growth takes more execution time for Phase II. Hence this paper presents modified algorithm aiming to reduce the execution time by effectively identifying high utility itemsets[13]

[14] Agarwal et al developed an algorithm for mining association rules between sets of items in large databases. Association rule mining are if/then statements that helps to uncover relationships between seemingly unrelated data in a [14] relational database or other information repository. Apriori Association rule mining technique uses a two step process. The first step is to identify all the frequent itemsets based on the support count value of the itemsets. It uses the download closure property of itemsets to remove the infrequent itemsets. The second step is the generation of

association rules from the frequent itemsets using the support and confidence .

III. PROPOSED ALGORITHM WITH EXAMPLE

Step 1: Input:

- A Transaction data Base T & correspondent Profit table P
- Minimum utility value is 6

Table 1:Transaction Data Set (T)

TID	TRANSACTION
T1	A C D
T2	A C E G
T3	A B C D E F
T4	B C E
T5	B C E G

Table 2: Item & correspondent profit (P)

ITEM	A	B	C	D	E	F	G
PROFIT	5	2	1	2	3	1	1

Step 2: We scan above table T & P and calculate the weighted transaction utility (WTU) of each item

WTU(A)= A is pre sent in transaction number T1, T2, T3 in table 1. Also profit of A is 5 as mentioned in table 2. So the weighted transaction utility of A is calculated as follows:

WTU(A)= 5+5+5 = 15
WTU(B)= 2+2+2 = 6
WTU(C)= 5
WTU(D)= 4
WTU(E)= 12
WTU(F)= 1
WTU(G)= 2

Now we compare the wtu of each item with minimum utility which is 6 & include only those items in high utility list whose wtu is greater than or equal to the minimum utility

Now we see that the wtu of A , B, & E is greater then are equal to 6. So A, B & E are included in high utility item list

Step 3: we also sort all items found in step 2 in decreasing order of their utility

Table 3: Sorted High Utility Items of size1

Item	WTU
A	15
E	12
B	6

Step 4: In this step, we eliminate all those items from the transaction data base T, whose utility is less than the minimum utility.

In previous step, we see that item C, D, F & G are not high utility item sets so we eliminate these items from Table 1. Then we get a new table as follows:

Table 4: Updated Table 1

TID	TRANSACTION
T1	A
T2	A E
T3	A B E
T4	B E
T5	B E

Step 5: Now the high utility items of size 1 are A, B & E. we use these items to generate candidates items of size 2. The candidates of size 2 are obtained by finding all possible combinations of A, B & E. these are

AB, BE, AE.

Now we calculate WTU of AB BE and AE by using the updated table 1 .

WTU(AB)= AB together are present in transaction number T3 of updated table 1. So wtu of AB is (5 + 2 = 7)
WTU(BE)= present in 3 transactions of updated table 1 (15)
WTU(AE)= 16

Now we compare wtu of all these with minimum utility (6). We see that all these three items are also high utility items so we add these three items in the list of high utility items.

Step 6: Now the high utility items of size 2 are AB, BE & AE. we use these items to generate candidates items of size 3.

The candidates of size 3 are obtained by finding all possible combinations of AB, BE & AE. Only possible combination of size 3 is ABE

Now we calculate WTU of ABE by using the updated table 1 .

WTU(ABE)= ABE together are present in only 1 transaction T3 in updated table 1= 5+2+3=10. ABE is also a high utility item because its wtu is greater than minimum utility.

Now we donot have items , which can be combined to generate a larger item so our algorithm terminates here. The complete list of high utility item is as follows:

A
B
A
AB
AE
BE
ABE

Figure given below, shows Comparison based on the existing and proposed algorithm. This experiment use a Traffic Accidents Data Set (https://archive.ics.uci.edu/ml/index.php).

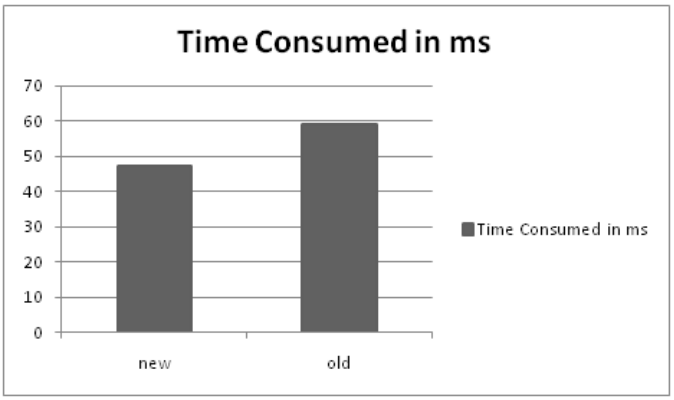

Figure. 2 Depicts the Time Consumption Comparison

IV. CONCLUSION

In this paper we reviewed the rundown of current methods of high use mining. We have limited ourselves to the exemplary problem of high utility mining. It's the era of all the high utility stuff set that exists for support and assurance in every regular knowledge index concerning negligible edges. This paper also suggests a technique for the mining of high utility element sets from a transaction database. The suggested algorithm requires less time on data set for experimental research.

REFERENCES

[1] Tan P.-N., Steinbach M., and Kumar V. ―Introduction to data mining, Addison Wesley Publishers‖. 2006
[2] Fayyad U. M., Piatetsky-Shapiro G. and Smyth, P. ―Data mining to knowledge discovery in databases, AI Magazine‖. Vol. 17, No. 3, pp. 37-54, 1996.
[3] https://www.sas.com/en_us/insights/analytics/data-mining.html
[4] C. F. Ahmed, S. K. Tanbeer, B.-S. Jeong, and Y.-K. Lee. Efficient tree structures for high utility pattern mining in incremental databases. In *IEEE Transactions on Knowledge and Data Engineering*, Vol. 21, Issue 12, pp. 1708-1721, 2009.
[5] A. Erwin, R. P. Gopalan, and N. R. Achuthan. Efficient mining of high utility itemsets from large datasets. In *Proc. of PAKDD 2008, LNAI 5012*, pp. 554-561.
[6] Y. G. Sucahyo and R. P. Gopalan. "CT-ITL: Efficient Frequent Item Set Mining Using a Compressed Prefix Tree with Pattern Growth". Proceedings of the 14th Australasian Database Conference, Adelaide, Australia, 2003.
[7] Vincent S. Tseng, Bai-En Shie, Cheng-Wei Wu, and Philip S. Yu, Fellow, IEEE "Efficient Algorithms for Mining High Utility Itemsets from Transactional Databases" IEEE Trans. Knowledge and Data Engineering, vol. 25, no. 8, August 2013
[8] Chowdhury Farhan Ahmed, Syed Khairuzzaman Tanbeer, Byeong-Soo Jeong, and Young-Koo Lee, Member, IEEE "Efficient Tree Structures for High Utility Pattern Mining inIncremental Databases" IEEE Trans. Knowledge and Data Engineering, Vol. 21, Issue 12, pp. 1708-1721, December 2009.
[9] Alva Erwin, Raj P. Gopalan, and N. R. Achuthan, "Efficient Mining of High Utility Itemsets from Large Datasets", In Proc. of PAKDD 2008.
[10] Shankar, S.; Purusothaman, T.; Jayanthi, S. "Novel algorithm for mining high utility itemsets" International Conference on Computing, Communication and Networking, Dec. 2008.
[11] Raymond Chan; Qiang Yang; Yi-Dong Shen, "Mining high utility itemsets" In Proc. of Third IEEE Int'l Conf. on Data Mining ,November 2003.
[12] Ramaraju, C., Savarimuthu N. "A conditional tree based novel algorithm for high utility itemset mining", International Conference on Data mining, June 2011.
[13] Ying Liu, Wei-keng Liao, Alok Choudhary "A Fast High Utility Itemsets Mining Algorithm" In Proc. of the Utility-Based Data Mining Workshop, 2005.
[14] Adinarayanareddy B ,O Srinivasa Rao, MHM Krishna Prasad, "An Improved UP-GrowthHigh Utility Itemset

Mining" International Journal of Computer Applications (0975-8887) Volume 58-No.2, November 2012.

[15] P. Asha, Dr. T. Jebarajan, G. Saranya, "A Survey on Efficient Incremental Algorithm for Mining High Utility Itemsets in Distributed and Dynamic Database" IJETAEJournal, Vol.4, Issue 1, January 2014.