Literature Review on Efficient Algorithms for Mining High Utility Itemsets from Transactional Databases

Ketkee Kailas Gaikwad
PG scholar, Dept of Computer Engineering
KJ College Of Engineering and Management Research
Pune, India
gaikwadket@gmail.com

Mininath Nighot
Assistant Professor, Department of Computer Engineering
KJ College Of Engineering and Management Research
Pune, India
imaheshnighot @gmail.com

***Abstract*—** This paper presenting a survey on finding itemsets with high utility. For finding itemsets there are many algorithms but those algorithms having a problem of producing a large number of candidate itemsets for high utility itemsets which reduces mining performance in terms of execution. Here we mainly focus on two algorithms utility pattern growth (UP-Growth) and UP-Growth+. Those algorithms are used for mining high utility itemsets, where effective methods are used for pruning candidate itemsets. Mining high utility itemsets Keep in a special data structure called UP-Tree. This, compact tree structure, UP-Tree, is used for make possible the mining performance and avoid scanning original database repeatedly. In this for generation of candidate itemsets only two scans of database. Another proposed algorithms UP Growth+ reduces the number of candidates effectively. It also has better performance than other algorithms in terms of runtime, especially when databases contain huge amount of long transactions. Utility-based data mining is a new research area which is interested in all types of utility factors in data mining processes. In which utility factors are targeted at integrate utility considerations in both predictive and descriptive data mining tasks. High utility itemset mining is a research area of utility based descriptive data mining. Utility based data mining is used for finding itemsets that contribute most to the total utility in that database.

Keywords- *Data Mining, high utility itemset, utility mining.*

*****

I. INTRODUCTION

A. DATA MINING

Data mining and Knowledge Discovery from data bases are two of the major areas receiving much attention in recent years. Data mining, the extraction of hidden predictive information from large databases, has a great potential to help data owners in focusing on the most important information in their data warehouses. Knowledge Discovery in Databases (KDD) is the process of identifying valid , previously unknown and potentially useful patterns in data. These patterns becomes useful in order to explain existing data, predict or classify new data, to put the contents of a large database into a nutshell supporting decision making and graphical representation of data. Data mining is the process of revealing nontrivial, previously unknown and potentially useful information from large databases. Plenty of data mining tasks, such as frequent pattern mining, weighted frequent pattern mining, and high utility pattern mining are having important role of useful hidden pattern retrieval from a database. Out of these, frequent pattern mining is a fundamental research topic that has been applied many of databases, such as transactional databases, streaming databases, and time series databases, and in various application domains, such as bioinformatics, Web click-stream analysis, and mobile environments. Taking all this into an account, utility mining is being considered as an important topic in data mining field. Mining high utility itemsets from databases refers to finding the itemsets with high profits.

Here, the meaning of itemset utility is attraction, importance or profitability of an item to users. Utility of items in a transaction database consists of two aspects:

1. The importance of distinct items, which is called external utility, and
2. The importance of items in transactions, which is called internal utility.

Table 1.1: An Example Database

TID	*Transaction*	*TU*
T1	(A,1) (C,10) (D,1)	17
T2	(A,2) (C,6) (E,2) (G,5)	27
T3	(A,2) (B,2) (D,6) (E,2) (F,1)	37
T4	(B,4) (C,13) (D,3) (E,1)	30
T5	(B,2) (C,4) (E,1) (G,2)	13
T6	(A,1) (B,1) (C,1) (D,1) (H,2)	12

Table 1.2: Profit Table

Profit	5	2	1	2	3	5	1	1
Item	A	B	C	D	E	F	G	H

B. UTILITY MINING

Judging the utility of items by its presence in the transaction set is the older methods of ARM. The occurrence of an item is not enough to reflect the actual utility. One of the most challenging data mining tasks is the mining of high utility itemsets efficiently [4]. Identification of the itemsets with high utilities is called as Utility Mining. Cost, quantity, profit or any other user expressions of preference can be used to measure the utility. For example, a computer system may be more profitable than a telephone in terms of profit. Utility mining model was proposed to define the utility of itemset [5]. The utility is a measure of how useful or profitable an itemset X is. The utility of an itemset X, i.e., u(X), which is the sum of the all utilities of itemset X in all the transactions containing X. An itemset X is called a high utility itemset if and only if u(X) greater than or equal to min_utility, where min_utility is a user defined minimum utility threshold. In short, finding all itemsets having utility higher than user defined minimum utility is the goal of high-utility itemset mining [6]. In this paper we are presenting the literature survey study over the concept of high utility itemset mining using the concepts of data mining. In section II we are presenting the example of mining frequent itemset from transaction dataset. In section III we are presenting the different methods presented for high utility mining. Utility-based data mining is a broad topic that covers all aspects of economic utility in data mining. It covers predictive and descriptive methods for data mining among the later, especially detection of rare events of high utility. This paper describes methods for itemset mining or more specifically, mining utility-frequent itemsets which is a special form of high utility itemset mining. Utility of an itemset is defined as the product of its external utility and its internal utility. An itemset is called a low-utility itemset, if its utility is less than a user-specified minimum utility threshold.

II. TERMS AND DEFINITIONS

Here we are discussing some basic definitions about utility of an item, utility of itemset in transaction, utility of itemset in database and also related works and define the problem of utility mining and then we will introduce related strategies. Given a finite set of items I= {i_1, i_2, i_3…i_m} each item $i_p(1\leq p\leq m)$ has a unit profit $pr(i_p)$. An itemset X is a set of k distinct items I= {i_1, i_2, i_3…i_k} , where ij I, $1\leq j\leq k$. k is the length of X. An itemset with length k is called a k itemset. A transaction database D ={T_1; T_2; . . . ; T_n } contains a set of transactions, and each transaction Td($1\leq d\leq n$) has a unique identifier d, called TID. Each item i_p in transaction Td is associated with a quantity $q(i_p,Td)$, that is, the purchased quantity of i_p in Td.

Definition 1: Utility of an item i_p in a transaction Td is denoted as u (i_p, Td) and defined as $pr(i_P)\times q(i_p,Td)$

Definition 2: Utility of an itemset X in Td is denoted as U(x,Td) and defined as $\Sigma i_p \in X \vee X\subseteq Tdu\ (i_p,Td)$

Definition 3: Utility of an itemset X in D is denoted as u(X) and ΣX Td Td D u(X, Td)⊆ ∧ ⊆

Definition 4: An itemset is called a high utility itemset if its utility is no less than a user-specified minimum utility threshold or low-utility itemset represented by min-util.

Problem Statement: Given a transaction database D and a user-specified minimum utility threshold min_util, the problem of mining high utility itemsets from D is to find the complete set of the itemsets whose utilities are larger than or equal to min_util.

III. RELATED WORK

This paper was to application spectrum is wide in many real-life applications and is an important research issue in data mining area. Utility mining emerges as an important topic in data mining field. Here high utility item sets mining refers to importance or profitability of an item to users. Number of algorithms like apriori (level – wise search) has been proposed in this area, they cause the problem of generating a large number of candidate itemsets. That will lead to high requirement of space and time and so that performance will be less. It is not at all good when the database contains transactions having long size or high utility itemsets which also having long size. Mining high utility item sets from databases refers to finding the itemsets with high profits. Here, the meaning of item set utility is interestingness, importance, or profitability of an item to users.

IV. LITERATURE REVIEW

Several researchers have done the research in many areas:

1. R. Agrawal et in [1] taken some mining information and in [2] proposed apriori algorithm, it is used to obtain frequent itemsets from the database. In miming the association rules we have the problem to generate all association rules that have support and confidence greater than the user specified minimum support and minimum confidence respectively. The first pass of the algorithm simply counts item occurrences to determine the large 1-itemsets. First it generates the candidate sequences and then it chooses the large sequences from the candidate ones. Next, the database is scanned and the support of candidates is counted. The second step involves generating association rules from frequent itemsets. Candidate itemsets are stored in a hash-tree. The hash-tree node contains either a list of itemsets or a hash table. Apriori is a classic algorithm for frequent itemset mining and association rule learning over transactional databases. After identifying the large itemsets, only those itemsets are allowed which have the support greater than the minimum support allowed. Apriori Algorithm generates lot of candidate item sets and scans database every time. When a new transaction is added to the database then it should rescan the entire database again.

2. J. Han et al in [6] proposed frequent pattern tree (FP-tree) structure, an extended prefix tree structure for storing crucial information about frequent patterns, compressed and develop an efficient FP-tree based mining method is Frequent pattern tree structure. Pattern fragment growth mines the complete set of frequent patterns using the FP-growth. It constructs a highly compact FP-tree, which is usually substantially smaller than the original database, by which costly database scans are saved in the subsequent mining processes. It applies a pattern growth method which avoids costly candidate generation. FP-growth is not able to find high utility itemsets.

3. Liu et al in [10] proposes a Two-phase algorithm for finding high utility itemsets. The utility mining is to identify high utility itemsets that drive a large portion of the total utility. Utility mining is to find all the itemsets whose utility values are beyond a user specified threshold. Two-Phase algorithm, it efficiently prunes down the number of candidates and obtains the complete set of high utility itemsets. We explain transaction weighted utilization in Phase I, only the combinations of high transaction weighted utilization itemsets are added into the candidate set at each level during the level-wise search. In phase II, only one extra database scan is performed to filter the overestimated itemsets. Two-phase requires fewer database scans, less memory space and less computational cost. It performs very efficiently in terms of speed and memory cost both on synthetic and real databases, even on large databases. In Two-phase, it is just only focused on traditional databases and is not suited for data streams. Two-phase was not proposed for finding temporal high utility itemsets in data streams. However, this must rescan the whole database when added new transactions from data streams. It need more times on processing I/O and CPU cost for finding high utility itemsets.

4. Shankar [11] presents a novel algorithm Fast Utility Mining (FUM) which finds all high utility itemsets within the given utility constraint threshold. To generate different types of itemsets the authors also suggest a technique such as Low Utility and High Frequency (LUHF) and Low Utility and Low Frequency (LULF), High Utility and High Frequency (HUHF), High Utility and Low Frequency (HULF).

V. METHODS

There are various methods for mining high utility itemsets. Mining high utility itemsets has four main methods used for from transactional databases that are given as follows:

a. Data Structure

Data Structure is nothing but organizing the data so that we can use that data efficiently. Mining high utility itemsets Keep in a special data structure called UP-Tree. This, compact tree structure, UP-Tree, is used for make possible the mining performance and avoid scanning original database repeatedly. It will also keep the transactions information and high utility itemsets.

b. UP-Growth Mining Method

In the first step we get the global UP tree that is mining UP-Tree by FP-Growth. Which can be used for generating PHUIs will generate so many candidates in order to avoid that UP-Growth method is used with two techniques mainly: First one is discarding unpromising items during constructing a local UP-Tree and second is discarding local node utilities.

c. An Improved Mining Method: UP-Growth+

As compared with UP-Growth FP-Growth gives the better performance. FP growth is used to find the frequent itemsets. FP-Growth uses DLU and DLN to decrease overhead utilities of itemsets. However, the overestimated utilities can be closer to their actual utilities by eliminating the estimated utilities that are closer to actual utilities of unpromising items and descendant nodes. In this section, we propose an improved method, named UP-Growth+, for reducing overestimated utilities more effectively. In UP-Growth, minimum item utility table is used to reduce the overestimated utilities. In UP-Growth+, minimal node utilities in each path are used to make the estimated pruning values closer to real utility values of the pruned items in database.

d. Efficiently Identify High Utility Itemsets

After finding all PHUIs, the third step is to identify high utility itemsets and their utilities from the set of PHUIs by scanning original database once [3], [11]. However, in previous studies, two problems in this phase occur: 1) number of HTWUIs is too large; and (2) scanning original database is very time consuming. In our framework, overestimated utilities of PHUIs are smaller than or equal to TWUs of HTWUIs since they are reduced by the proposed strategies. Thus, the number of PHUIs is much smaller than that of HTWUIs. Therefore, in phase II, our method is much efficient than the previous methods. Moreover, although our methods generate fewer candidates

VI. CONCLUSION

In this we are presenting a literature survey on various algorithms used for mining high utility itemsets. In this we are having comparative analysis of different algorithms present for mining high utility itemsets. Furthermore, we have focused on mainly two algorithms that is UP-Growth and UP-Growth$^+$ for mining high utility itemsets from transaction databases. . Mining high utility itemsets Keep in a special data structure called UP-Tree. This, compact tree structure, UP-Tree, is used for make possible the mining performance and avoid scanning

original database repeatedly. With only two scan of the UP-Tree, PHUIs can be efficiently generated. Another proposed algorithms UP Growth+ reduces the number of candidates effectively. It also has better performance than other algorithms in terms of runtime, especially when databases contain huge amount of long transactions. Utility-based data mining is a new research area which is interested in all types of utility factors in data mining processes. In which utility factors are targeted at integrate utility considerations in both predictive and descriptive data mining tasks. High utility itemset mining is a research area of utility based descriptive data mining. Utility based data mining is used for finding itemsets that contribute most to the total utility in that database. We are use those systems in applications in Website click stream analysis, Cross marketing in retail stores, online e-commerce management, Mobile commerce environment and for finding important patterns in biomedical applications.

ACKNOWLEDGMENT

The authors would like to thanks the Dept of Computer Engineering,KJ College Of Engineering and Management Research Pune, Maharashtra, India. Authors also thanks for the guidance and cooperation of all the faculty members.

REFERENCES

[1] R. Agrawal and R. Srikant, T. Imielinski, A. Swami, "Mining association rules between sets of items in large databases", in proceedingss of the ACM SIGMOD International Conference on Management of data, pp. 207-216, 1993.

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

[3] C.F. Ahmed, S.K. Tanbeer, B.-S. Jeong, and Y.-K. Lee, "Efficient Tree Structures for High Utility Pattern Mining in Incremental Databases," IEEE Trans. Knowledge and Data Eng., vol. 21, no. 12, pp. 1708-1721, Dec. 2009.

[4] R. Chan, Q. Yang, and Y. Shen, "Mining High Utility Itemsets," Proc. IEEE Third Int'l Conf. Data Mining, pp. 19-26, Nov. 2003.

[5] J.H. Chang, "Mining Weighted Sequential Patterns in a Sequence Database with a Time-Interval Weight," Knowledge-Based Systems, vol. 24, no. 1, pp. 1-9, 2011.

[6] M.-S. Chen, J.-S. Park, and P.S. Yu, "Efficient Data Mining for Path Traversal Patterns," IEEE Trans. Knowledge and Data Eng., vol. 10, no. 2, pp. 209-221, Mar. 1998.

[7] C. Creighton and S. Hanash, "Mining Gene Expression Databases for Association Rules," Bioinformatics, vol. 19, no. 1, pp. 79-86, 2003.

[8] M.Y. Eltabakh, M. Ouzzani, M.A. Khalil, W.G. Aref, and A.K. Elmagarmid, "Incremental Mining for Frequent Patterns in Evolving Time Series Databases," Technical Report CSD TR#0802, Purdue Univ., 2008.

[9] Erwin, R.P. Gopalan, and N.R. Achuthan, "Efficient Mining of High Utility Itemsets from Large Data Sets," Proc. 12th Pacific-Asia Conf. Advances in Knowledge Discovery and Data Mining (PAKDD), pp. 554-561, 2008.

[10] A. Erwin, R.P. Gopalan, and N.R. Achuthan, "Efficient Mining of High Utility Itemsets from Large Data Sets," Proc. 12th Pacific-Asia Conf. Advances in Knowledge Discovery and Data Mining (PAKDD), pp. 554-561, 2008.

[11] E. Georgii, L. Richter, U. Ru¨ ckert, and S. Kramer, "Analyzing Microarray Data Using Quantitative Association Rules," Bioinformatics, vol. 21, pp. 123-129, 2005.

[12] J. Han, G. Dong, and Y. Yin, "Efficient Mining of Partial Periodic Patterns in Time Series Database," Proc. Int'l Conf. on Data Eng., pp. 106-115, 1999.

[13] J. Han and Y. Fu, "Discovery of Multiple-Level Association Rules from Large Databases," Proc. 21th Int'l Conf. Very Large Data Bases,pp. 420-431, Sept. 1995.

[14] J. Han, J. Pei, and Y. Yin, "Mining Frequent Patterns without Candidate Generation," Proc. ACM-SIGMOD Int'l Conf. Management of Data, pp. 1-12, 2000.

[15] S.C. Lee, J. Paik, J. Ok, I. Song, and U.M. Kim, "Efficient Mining of User Behaviors by Temporal Mobile Access Patterns," Int'l J. Computer Science Security, vol. 7, no. 2, pp. 285-291, 2007.

[16] H.F. Li, H.Y. Huang, Y.C. Chen, Y.J. Liu, and S.Y. Lee, "Fast and Memory Efficient Mining of High Utility Itemsets in Data Streams," Proc. IEEE Eighth Int'l Conf. on Data Mining, pp. 881- 886, 2008.

[17] Y.-C. Li, J.-S. Yeh, and C.-C. Chang, "Isolated Items Discarding Strategy for Discovering High Utility Itemsets," Data and Knowledge Eng., vol. 64, no. 1, pp. 198-217, Jan. 2008.

[18] C.H. Lin, D.Y. Chiu, Y.H. Wu, and A.L.P. Chen, "Mining Frequent Itemsets from Data Streams with a Time-Sensitive Sliding Window," Proc. SIAM Int'l Conf. Data Mining (SDM '05), 2005.

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

[20] R. Martinez, N. Pasquier, and C. Pasquier, "GenMiner: Mining nonredundant Association Rules from Integrated Gene Expression Data and Annotations," Bioinformatics, vol. 24, pp. 2643-2644,2008.

[21] J. Pei, J. Han, H. Lu, S. Nishio, S. Tang, and D. Yang, "H-Mine: Fast and Space-Preserving Frequent Pattern Mining in Large Databases," IIE Trans. Inst. of Industrial Engineers, vol. 39, no. 6, pp. 593-605, June 2007.

[22] J. Pei, J. Han, B. Mortazavi-Asl, H. Pinto, Q. Chen, U. Moal, and M.C. Hsu, "Mining Sequential Patterns by Pattern-Growth: The Prefixspan Approach," IEEE Trans. Knowledge and Data Eng.,vol.16, no.10, pp. 1424-1440, Oct. 2004.