Information and Knowledge Management
ISSN 2224-5758 (Paper) ISSN 2224-896X (Online)
Vol.5, No.12, 2015

A Survey on Discovering High Utility Itemset Mining from Transactional Database

Shekhar Patel B Madhushree
L.J Institute of Engineering and Technology, GTU,Ahmedabad, India

Abstract
Data Mining is the process of evaluating data from different outlooks and summarizing it into useful information. It can be defined as the process that extracts information contained in very large database. Traditional Data mining methods have been focused on to finding a correlation between items which are frequently appearing in the database. And relative importance of each item is not consider in frequent pattern mining. High utility mining is an area research where utility based mining can be done. Mining high utility itemset from a transactional database refers to the discovery of itemset with high utility in a terms like weight, unit profit or value. In this paper we present literature survey of currently used algorithms for high utility itemset mining.
Keywords: High utility, Transactional Database, HUI_Miner, FHM

1. Introduction

DATA mining is the process of revealing nontrivial, previously unknown and potentially useful patterns form large database Data mining can be used to transform the raw data into meaningful and useful information for business analysis processes referred to as Business intelligence. Business leaders have realized that "getting closer to the Customer" is crucial to the growth of the business. Discovering useful patterns hidden in a database plays an essential in several data mining task such as frequent pattern mining, weighted itemset mining, and high utility mining. Among them frequent pattern mining is a fundamental research topic (Visent Tseng, 2013) that has been applied to different kind of database such as transactional database , streaming database and time series database.

Nevertheless, relative importance of each item is not considered in frequent pattern mining. However the important limitation of FIM is that it assumes that each item cannot appear more than once in each transaction and that all items have the same importance (weights, unit profit or value)(Phillipe, 2014). To address this problem, weighted association rule mining was proposed (Shankar, 2008). In this framework, weights of items, such as unit profits of items in transaction databases, are considered. With this concept, even if some items appear infrequently, they might still be found if they have high weights. However, in this framework, the quantities of items are not considered yet. Therefore, it cannot satisfy the requirements of users who are interested in discovering the itemsets with high sales profits, since the profits are composed of unit profits, i.e., weights, and purchased quantities. In other words, statistical correlation may not measure how useful an itemset is in accordance with user's preferences (i.e. profit). The profit of an itemset depends not only on the support (total number of items an itemset occur in a transactional database out of the total number of transactions) of the itemset, but also on the prices of the items in that itemset. The above limitation motivated the researchers (Hamilton, 2006) to develop a utility based itemset mining approach, which allows a users to conveniently express his or her perspectives concerning the usefulness of itemset with utility values larger than threshold. Utility based data mining is a new research area (Shankar, 2008) interested in all types of types of utility factors in data mining process.

Mining high utility itemset from database refers to finding the itemset utility is interestingness, importance, or profitability, of an items to users. Utility of items in a transactional database consist of two aspects: 1) the importance of distinct item (Visent Tseng, 2013), which is called *external* utility, and 2) the importance of item in transactions, which is called *internal* utility..

Utility of an itemset is defined as the product of its external utility and internal utility (Hamilton 2006). An itemset is called a high utility itemset if its utility is no less than a user threshold minimum utility threshold; otherwise it is called a low- utility itemset (Shankar, 2008). Mining high utility itemset from database is an important task has a wide range of applications such as website click stream analysis (Visent Tseng, 2013), business promotion in chain hypermarkets, cross marketing in retail stores (Visent Tseng, 2013), online e-commerce management, mobile commerce environment planning and even in biomedical applications.

Example:
Example of a transaction database representing the sales data and the profit associated with the sale of each unit of the items.

Table 1: Transaction Database

TID	Item A	Item B	Item C
T1	5	0	10
T2	0	6	0
T3	4	0	1
T4	2	3	8
T5	3	7	6
T6	3	0	1
T7	0	6	0
T8	4	5	25
T9	3	0	0
T10	0	5	2

Table 2: Unit Profit

Item Name	Unit Profit
Item A	4
Item B	8
Item C	3

Let us consider the itemset AB. Since, there are only 3 transactions T4, T5 and T8 which Contains AB itemset out of 10 transactions. So, support for itemset AB is

Support (AB) = 3 / 10 * 100 = 30 %

In T4 transaction, units gain by item A and B are 2 and 4. Respectively, the profit earned from the sale of itemset AB. In T4 transaction is given by,

Profit (AB, T4) = 2 * profit (A) + 4 * profit (B)
= 2*4 + 3*8
= 32

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

Profit (AB) = profit (AB, T4) + profit (AB, T5) + Profit (AB, T8)
= (2*4+3*8) + (3*4+7*8) + (4*4+5*8)
= (8+24) + (12+56) + (16+40)
= 32+68+56
= 156

Similarly, we can calculate the support values for the different itemsets and also the profit obtained by the sale of those itemsets by all the ten transactions as indicated in table 3.

Table 3: Support and Profit

Itemset	Support (%)	Profit
A	70	96
B	60	256
C	70	159
AB	30	156
BC	40	283
AC	60	237
ABC	30	273

If we consider minimum support 50%, then we can observe that there are only 4 itemsets A, B, C and AC which have the support greater than the threshold value (min_sup). So, they qualify as frequent itemsets. But if we consider it profit wise then we can find out of 4 most profitable itemsets B, BC, AC, ABC only B and AC are frequent itemsets. Itemsets BC and ABC are not frequent but still they fetch the more profit than other itemsets.

As we can see from table 3 single unit of item B fetch More profit than single unit of Itemset A and B.

From this Example, we can illustrate frequent Itemset mining may not always satisfy profit wise requirements of sales manager .In this case, the support (%) attribute of the Itemsets reflects the the statistical correlation not the semantic significance of items.

2. Literature Review

R. Agrawal et al in (Agrawal 1994) 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.

Liu et al in (Yung 2005) 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.

Yao et al. (Hamilton 2006) proposed two utility mining algorithms UMining and Umining_H based on efficient pruning strategies using upper bound by applying an estimation method to prune the search space. However it cannot capture the complete set of high utility itemsets, since some high utility patterns may be pruned during the mining process. This algorithm overestimates too many patterns in the beginning and also suffers from excessive candidate generations. The pruning strategy used in Umining_H may miss some of high utility itemset.

Shankar (Shankar, 2008) presents a novel algorithm Fast Utility Mining (FUM) which finds all high utility itemsets within the given utility 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). The proposed FUM algorithm scales well as the size of the transaction database increases with regard to the number of distinct items available.

Alva Erwin (Alva 2008) Advised CTU-PROL algorithm for efficient mining of high utility itemsets from large datasets these algorithms search the large TWU items in the transaction database. If data sets is too large to be held in main memory, the algorithm generates 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 Limited, it built a single CUP-Tree for mining high utility itemsets.

Ahmed et al. (Ahead 2009) implied a structure named IHUP-Tree for maintaining essential information about utility mining. It avoids scanning of database for multiple times and generating candidates or patterns during the mining process. However, although IHUP-Tree produces better performance than Two-Phase and IIDS, it still provides too many HTWUIs.

Menghi Liu et al. (Liu 2012) proposed a novel data structure, utility-list, and developed an efficient algorithm, HUI_Miner, for high utility itemset mining. It does not generate too many candidate key as previous algorithms are generated. It is a single phase algorithms and it does not required multiple scans. But main disadvantage of this algorithm is that calculating the utility of an itemset joining

Utility list is very costly.

Tseng et al. (Viscent 2013) proposed a novel algorithm named UP-Growth which applies several pruning and counting strategies during the data mining processes. By the proposed strategies, the estimated utilities are effectively decreased in UP-Trees during the data mining processes and the number of HTWUIs is further reduced. Therefore, the system performance of utility mining can be improved significantly.

Philippe Fournier et al. (Phillipe 2014) presented a novel algorithm for high utility itemset mining named FHM (Fast High-Utility Miner). This algorithm integrates a novel strategy named EUCP (Estimated Utility Co-occurrence Pruning) to reduce the number of join operations when mining high utility itemset using the utility list data structure. It works only on static database. We should try to develop it for dynamic database also.

3. Comparison Between Algorithms

In the previous section we introduced the overview of Data Mining, Frequent Itemset Mining and High Utility

Itemset Mining. A comparison of the various Algorithms, Techniques, approaches and limitations that have been Defined in various research publications have been given in this section

No	Title of Paper	Year	Author	Name Of Algorithm	Limitation
1	Mining association rule between sets of items in large databases.	1994	R. Agrawal, T. Mielinski, A. Swami.	Appriori	Large no scan is required, time Consuming
2	A Two-Phase Algorithm for Fast Discovery of High Utility Item sets.	2005	Ying Liu, Wei-keng Liao, and Alok Choudhary	Two Phase	Multiple scans of database And generates many candidate Itemsets
3	Mining itemset utilities from Transactional databases.	2006	H Yao, H J Hamilton	U_Mining	Pruning Strategy of this algorithm miss some utility itemset.
4	Novel Algorithm foe Mining High utility Itemsets.	2008	Shankar S, Dr. Pursothaman, Jayanthi S	FUM	Time and Memory Consuming
5	Efficient Mining of High Utility Itemset From Large Database	2008	Alva Ervin, Raj Gopalan, N.R Achutan	CTU-PROL	Large Computation Time
6	Efficient Tree Structures For High Utility Pattern Mining in Incremental Database	2009	Choudhary Ahmed, Sayed Tanber, Beyong Jeong, Young Lee	IHUP	Huge Set of PHUI, Low Threshold for long Transaction
7	Mining High Utility Itemset without Candidate Generation	2012	Mengchi Liu, Jufeng Qu	HUI_Miner	Perform Costly Join Operations on each pattern search
8	Efficient Algorithm for Mining High Utility Itemset From Transactional Database	2013	Vincent Tseng, Bai- Shie, Cheng wu, Phillipe S. yu.	UP_Growth UP_Growth+	Complex for evaluating due to tree structure
9	FHM-Fast High Utility Mining For	2014	Philipe Viger, Cheng Wu, S Zida, Vincent Tseng	FHM	Large Memory Ovrhead

4. Conclusion

In Data Mining, Association Rule Mining is one of the most important tasks. A large number of efficient algorithms are available for association rule mining, which considers mining of frequent itemsets. But an emerging topic in Data Mining is Utility Mining, which incorporates utility considerations during itemset mining. In this paper we detailed study about the different High utility mining algorithm, their work flow and their limitations. This paper provides an overview a comparative study of various algorithms that are used to improvise the efficiency of mining high utility itemsets. In the future scope, we will be proposing algorithms for mining high utility itemset.

References

Vincent S. Tseng, Bai-En Shie, Cheng Wu, philip S. Yu (2013). " Efficient Algorithms For Mining High Utility Itemset from Transactional Databases", IEEE transactions on knowledge and data engineering Vol 25(2), pp. 1772-1786.

Philippe Viger, Cheng Wu, Souleymane Zida, Vincent S. Tseng (2014). " FHM: Faster High Utiltiy Mining Itemset mining Using Estimated Utility Co-occurrence purning". Springer International, Switzerland, pp. 83-92.

Shankar S, Dr. Pursothaman T, Jayanthi S (2008). "Novel Algorithm for mining High Utility Itemsets**,** International Conference on Computing, Communication and Networking, St. Thomas, 2008, pp. 1-6.

H J Hamilton, H Yao (2006). "Mining Itemset Utilities from Transactional Database", pp. 88-96

Yao H and Hamilton H j (2006). “Mining itemset utilities from transaction database”, Data and Knowledge Engineearing, pp. 603-626.

C.F.Ahmed, S. Khairuzzaman Tanbeer,Byeong-Soo Jeong, Young-Koo Lee (2009). “Efficient Tree Structure for High Utility Pattern Mining in Incremental Databases”, IEEE Transactions on knowledge and data Engineering ,Vol 21(12), pp. 1708-1781.

Agrawal R, Srikant R (1994). “Fast algorithms for mining association rules”, Procedings of 20th International conference on Very Large Databases, Santiago, Chile, pp. 487-499.

Yung Liu, Woe heng Liao, and Alok Choudhary (2005). “A two Phase algorithm for fast discovery of high utility Itemsets”, Springer International, Berlin, pp. 689-695

Alva Erwin, Raj P. Gopalan, N.R. Achuthan (2008). “Efficient Mining ofHigh Utility Itemset From Large Datasets”, Springer , Berlin, pp. 554-561.

Menghchi Liu, Junfeng Qu (2012). “Mining High Utility Itemsets without Candidate Generation”, CIKM, Maui. USA, pp. 55-63.