RESEARCH ARTICLE OPEN ACCESS

A Survey on Efficient Algorithm for Mining Top-k High Utility Itemsets

*Neha Goliwar[1], Prof. Mrs. R.A.Patankar[2]
[1,2]*Department of Computer Engineering Maharashtra Institute of Technology,, Pune, Maharashtra, India.*
*Corresponding author: *Neha Goliwar*

ABSTRACT
High utility thing sets (HUIs) mining is a rising subject in information mining, which alludes to finding all thing sets having an utility meeting a client determined least of utility edge min_util. Not with standing, setting min_util suitably is a troublesome issue for clients. As a rule, finding a fitting least of utility edge by experimentation is a monotonous procedure for clients. In the event that min_util low, an excessive number of HUIs produced, which may bring about the mining procedure to be exceptionally wasteful. Then again, if min_util is set high, so no HUIs will found. Therefore, we address this above issues by proposing the other structure for top-k high utility mining, where k is the number to be covered of HUI mining. Two sorts of proficient calculations named TKU (mining Top-K Utility sets) and TKO (mining Top-K utility sets in one phase) are propose for the mining such thing sets without set the min_util. We give an auxiliary examination of the two calculations with talks on their preferences and restrictions..

Keywords: - TKO, TKU, UP- growth, UP-growth+, high utility itemset, frequent itemset mining.

Date of Submission: 24-08-2017 Date of acceptance: 13-09-2017

I. INTRODUCTION

Frequent item sets mining is a research topic in data mining. FIM contain a frequent item set in large amount but value of itemset is low and if the frequency is low so loses the valuable itemsets. So, it cannot fulfills the users requirements who desire to itemsets with high utility because of valuable information's are lose such as high profits. To another solution of these problem is a utility mining, utility contain an each item (e.g. Unit profit) and each transaction occurrences count (e.g. Quantity). The utility itemsets is an important topic and it can be measure in terms of weight, value, quantity and all other information's depending on the user's requirements. If the utility itemset is no less than user specified min utility, so this itemset is called a utility of high itemset. It contains a many applications like biomedicine, mobile computing, market analysis, etc.

In database, the HUI is a difficult, because in FIM used the downward closer property is does not hold the utility of itemsets. Superset the low utility itemset can be a high utility so the3 HUI pruning search space is also difficult. The solution of these problem, the transaction weighted utilization model was given the performance of mining task to be facilitate. In this model, suppose it's TWU is no less than the min_util and TWU of an itemset represents an upper bound on its utility so this itemset is called a high transaction weighted utilization itemset. In all HUIs contain in the complete set of HTWUIs. The TWU model consists a two phases, in first phase, the complete set of HTWUIs found, called a phase1. In second phase, scanning the database at one time and after scanning all HUIs is obtained by calculating the exact utilities of HTWUIs, called phase2.

The threshold value decided the output size can be very small or very large. For users, to choose an appropriate value of minimum utility threshold is difficult. The threshold depends on the performance of an algorithm. The algorithm generated the more HUIs so the more resources they consume. If threshold is set too high so the HUIs will not be found. If users need to find out the appropriate value for the min_util threshold, so users need to try the different -different threshold at every time by guessing and recalculating the algorithms again and again until user being satisfied with the result. This process is time consuming.

Therefore, we build TKP and they have tradeoffs on memory usage. The reason of TKP build that TKO utilizes minimal node utilities for further decreasing the utilities of item sets. Even though it spends the low time and memory to check and store minimal node utilities, they are more effective especially when there are many longer transactions in databases. In contrast, UP-Growth performs is good when min_util is using the small dataset or minimum dataset. The reason is that UP

growth+ algorithm is used for the large dataset to find the minimum utilities. They are more effective especially when there are many longer transactions in databases. UP-Growth carries more computations and is thus slower. Finally, the high utility item sets are efficiently identify of the set of PHUIs, which is much smaller than HTWUIs generated, by IHUP.

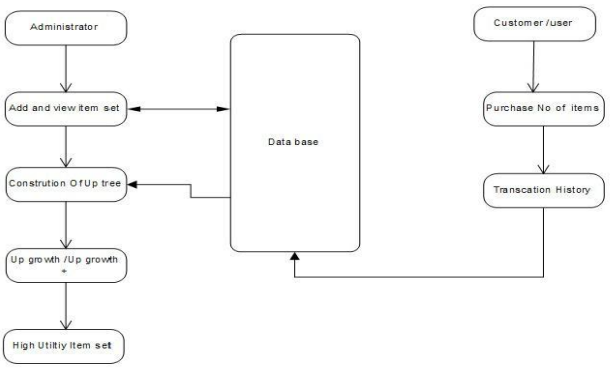
II. MOTIVATION

Many studies have to propose for mining HUIs, including Two-Phase, IHUP, HUP and UP-Growth. Two-phase and IHUP utilize is transaction-weighted downward closure property used for to find high utility item sets. They consist of two phases. In phase I, they find to all HTWUIs from the database. And in phase II, first the all database scanned at once and after scanning the database all high utility itemsets are set of identified the high transaction weighted utilities. Although these methods capture the complete set of HUIs, they may generate too many candidates in phase I, i.e. HTWUIs, which degrades the performance of phase II and the overall performance (in terms of time and space). In phase1 the number of candidate to be reduced, so the various methods have been proposed. Recently, proposed UP-Growth with four effective strategies DGU (Discarding global unpromising item), DGN (Discarding global node utilities), DLU (Discarding local unpromising item) and DLN (Discarding local node utilities), for mining HUIs. Experiments showed that the number of candidates generated by UP-Growth in phase I can be order of magnitudes smaller than that of HTWUIs. To the best of our knowledge, UP-Growth is the state-of-the-art method for mining high utility item sets. Although many studies addressed the topic of mining high utility item set from transaction databases, few of them showed the flexibility of mining top-k high utility item sets.

III. LITERATURE SURVEY

The basic concept is proposing a top-k high itemsets utility mining. The advantage of that proposed system is the two algorithms are propose for mining itemsets so such itemsets without setting a minimum utility threshold. However, the disadvantage is also there and the disadvantage is not incorporated with other utility mining task. E.g. Web access pattern. The differentially private FIM algorithm. The advantages of that proposed algorithms the pre-processing phase needs to perform only once. The disadvantage is the amount of noise added to guarantee privacy during the mining process. Proposed the TUS algorithm is design. The advantages are TUS guarantees there no sequence missed during the itemset mining process. The disadvantage is difficult for users to set a proper minimum utility. External utility and internal utility. UP growth and Up growth+ algorithm proposed. The transactions for large datasets. The advantage to improvement in the run time especially when database contains many long transactions. The disadvantage is the Up-growth algorithm is very slow. The advantage is Combine association rules, you can represent the regular textual and random textual both. The disadvantages when min_utility is low AprioriHC-D and AprioriHC both are algorithms can't perform well on dense databases since they suffer from the problem of a large amount of candidates. FUM algorithm scales strong as the capacity of the transaction database increases with regard to the number of distinct items available. Proposed a novel algorithm for mining high utility itemsets. This fast utility mining (FUM) algorithm finds all high utility itemsets within the disposed utility constraint threshold. The advantage is the proposed FUM algorithm scales strong as the capacity of the transaction database increases with regard to the number of distinct items available. Simpler and executing faster than other Umining algorithms. In addition, the disadvantage is in this algorithm is efficient when utility threshold is low. The basic concept is Incremental mining, Interactive mining. The advantage is, When the databases are incremental updated so does not require any restructuring operation. The disadvantage is, three tree structures require the maximum two database scan.

Architecture



Module 1: Administrator
The administrator saves the database of the transactions made by customers. In the daily market basis, each day a new product is launched, so after that the administrator would add the product or items, and update the new product information view in daily stock details

Module 2: Customer
Customer can purchase the number of items in daily market so after purchasing the items history is stored in the transaction database.
Module 3: Construction of UP-Tree
They are containing the two types of scanning:
First scan:-Initially Transaction Utility (TU) of each transaction is count. So TWU of each single item is assembling and that why removing the global unpromising items. The Utilities of unpromising items are excluding from the TU of the transaction. And then remaining promising items in the transaction are arranged according to the descending order of TWU.

Second scan: UP-Tree is generated by inserting transactions.

Module 4: UP-Growth Algorithm
The two strategies of the up growth high utility itemset, they namely DLN (Decreasing local node utilities) and DLU (Discarding local unpromising items). In DLN (Decreasing local node utilities), the minimum items are decreases during the construction of local tree. It is test during the insertion of the reorganized paths. In DLU (Discarding local unpromising items) strategy, removing the unpromising items from path during construction of up tree.

The Up-tree are generate in UP-Growth algorithm. In UP-tree, each node includes name, count, nu, parent, hlink and a set of child nodes. The details are: Name is a item name. Count is support count of node. Nu is called node utility, which is an estimate utility value of node. Parent records the parent node of the node. Hlink is a node link, which points to a node whose item name is the same as node name.

The construction of UP-tree can be performing with two scans of the original database.
Module 5: UP-Growth+ Algorithm Applying UP-Tree to the UP-Growth takes further execution time for Phase II.
The second algorithm i.e. UP-Growth+ curtail the execution time of high utility itemset. It measure the Maximum transaction Weighted Utilization (MTWU) from all items and considering multiple of min-sup as a user specified threshold value.

Module 6: UP-growth and UP-growth+ for transactional Database
Propose system will work, where continuous updating goes on emerging in a database. If the data is continuously insert to the original transaction database, then the database size becomes increase and mining the entire lot would take high computation time, hence proposed system will mine only the updated portion of the database. It will use earlier mining results to avoid unnecessary calculations.

IV. CONCLUSION

The problem of high utility item sets mining, where k is the desired number of high utility item sets mining. In this paper we have explain the which algorithms are used for to search a high utility itemsets. In addition, studied which algorithm is good for small dataset and large dataset. Two efficient algorithms TKU and propos for mining such item sets no need to set the minimum threshold.TKU is the first two-phase algorithm for mining. TKO is the first one-phase algorithm developed for top-k HUI mining. The UP-growth and UP- growth+ algorithm to use for generating a tree format of dataset. UP-growth+ used for a large dataset.

REFERENCE

[1]. V.S.Tseng, C-Wei Wu, P.F. Viger, P.D. Yu., "Efficient algorithms for mining top-k high utility itemsets", IEEE Transactions on Knowledge and Data Engineering, vol. 28, IEEE Journals and Magazines, 2015.
[2]. Sen Su, Shengzhi Xu, Xiang Cheng, Zhengyi Li, and Fangchun Ya, "Differentially Private Frequent Itemset Mining via Transaction Splitting", IEEE Transactions on Knowledge and Data Engineering, vol. 27, IEEE Journals and Magazines, 2015
[3]. XUE LiXia, Cheng- Wei Wu, Philippe FournierViger, and Philip S. Yu, "Fast algorithms for mining association rules," IEEE 5th International Conference of Software Engineering and Service Science,vol. 25, IEEE Conference Publication,2014.
[4]. Vincent S. Tseng, Cheng-Wei Wu, Viger, Philip S. Yu, " Efficiency mining top-k high utility sequential patterns", IEEE 13th International Conference on Data Mining, vol. 23, IEEE Conference Publication, 2013.
[5]. Vincent S. Tseng, BaiEn Shie, Cheng-Wei Wu, and Philip S. Yu, "Efficient algorithm for mining high utility itemsets from transactional databases," IEEE Transaction on Knowledge Data Engineering, vol. 25, IEEE Journals and Magazines, 2013.

[6]. Ramaraju C., Savarimuthu N., " A conditional tree based novel algorithms for high utility itemsets mining" , IEEE-International Conference on Recent Trends in Information Technology, ICRTIT, vol. 26, IEEE Conference Publication,2014.
[7]. C. Ahmed, S. Tanbeer, B. Jeong, and Y. Lee, "Efficient tree structures for high-utility pattern mining in incremental databases," IEEE Transaction on Knowledge Data Engineering, vol. 21, IEEE Journals and Magazines, 2009.
[8]. Shankar S., Purusothaman T., Jayanthi, S., "A Fast Algorithm for Mining High Utility Itemsets", IEEE International Advance Computing Conference (IACC 2009) Patiala, vol. 30, IEEE Conference Publication, 2009.
[9]. R. Chan, Q. Yang, and Y. Shen, "Mining High Utility Itemsets", Third IEEE International Conference on Data Mining (ICDM'03), IEEE Conference Publication, 2003.
[10]. P. Fournier-Viger and V. S. Tseng, "Mining top-k sequential rules," in Proc. International Conference Advanced Data Mining Application, IEEE Journals and Magazines, 2011
[11]. P. Fournier-Viger, C.Wu, and V. S. Tseng, "Mining top-k association rules," in Proc. Int. Conf. Can. Conf. Adv. Artif. Intell. IEEE Journals and Magazines, 2012.

International Journal of Engineering Research and Applications (IJERA) is **UGC approved** Journal with Sl. No. 4525, Journal no. 47088. Indexed in Cross Ref, Index Copernicus (ICV 80.82), NASA, Ads, Researcher Id Thomson Reuters, DOAJ.

Neha Goliwar. " A Survey on Efficient Algorithm for Mining Top-k High Utility Itemsets." International Journal of Engineering Research and Applications (IJERA) , vol. 7, no. 9, 2017, pp. 76–79.