Sadak Srilekha *et al*, International Journal of Computer Science and Mobile Applications,
Vol.2 Issue. 9, September- 2014, pg. 39-44 **ISSN: 2321-8363**

Mining High Utility Itemsets by Ameliorate UP-Growth+

Sadak Srilekha[1], Laxman.Maddikunta[2]

[1]M.Tech Student, Department of Computer Science and Engineering, KCEA Armoor, 503224, Telangana, India

srilekha.sadak@gmail.com

[2]Department of Computer Science and Engineering, HOD Of CSE, KCEA Armoor, 503224, Telangana, India

laxman_maddikunta@yahoo.co.in

Abstract

The discovery of itemsets with high utility like profits is referred by mining high utility itemsets from a transactional database. From past few years the number of relative algorithms has been proposed, for high utility itemsets the problem of producing a large number of candidate itemsets is incurred. The performance of mining is degraded by such a huge number of candidate itemsets in terms of space requirement and execution time. From the past few years transaction of Internet and purchasing of internet is increased. The client or customer can select the products based on their interest. In the internet the product sellers publish their ads. Two algorithms are proposed in this paper for mining high utility itemsets with a set of effective strategies for pruning candidate itemsets, namely UP-Growth (Utility Pattern Growth) and UP-Growth. In a tree-based data structure named UP-Tree (Utility Pattern Tree) The information of high utility itemsets is maintained such that with only two scans of database candidate itemsets can be generated efficiently.

Keywords: data mining, utility mining, candidate pruning, frequent itemset, high utility itemset

I. Introduction

Data Mining refers to extracting or mining knowledge from large amounts of data. Over the past some years in huge databases finding of frequent patterns is very significant use in many applications. The primary goal is to discover hidden patterns, unexpected trends in the data. In order to get better understanding of the underlying processes, the mining of data is concerned with the extracting of huge volume of information to automatically find the interesting relationships or regularities. Data mining activities uses combination of techniques from database artificial intelligence, statistics, technologies machine learning. This includes bio informatics, genetics, medicine, clinical analysis, education, retail and marketing research.

Utility Mining is one of the most challenging data mining tasks is the mining of high utility itemsets efficiently. In Utility Mining the high utility itemsets are identified. The Utility can be measured as per the user preferences utility can be measured in terms of cost, profit or other expressions. The restriction of frequent itemset mining motivated researchers to conceive a utility based mining approach, which allows a user to freely or easily express their outlook concerning the usefulness of itemsets as utility values and then the itemsets which are having the utility value higher than the threshold value that itemsets will be discovered.

Mining high utility itemsets from databases refers to finding the itemsets with high profits. Itemset utility mining is importance, interestingness or portability of an item to users. With the broad applications the High utility itemsets mining has become the most useful and interesting data mining tasks. And with the given threshold value item futility mining identifies the itemsets whose utility satisfies a given threshold. The preferences and usefulness of item is quantified by the user using different values. In a transaction database this itemset consists of two aspects: First one is internal utility, is nothing but single transaction of an itemset. And second one is External utility, is nothing but different Transaction database of itemset [13,16,21].

II. Literature Survey

R. Agrawal et al in [2] proposed Apriori algorithm, from the database the rare itemsets are obtained that is used to obtain frequent itemsets from the database. In miming, we have the problem to generate all association rules that have support and confidence greater than the user specified minimum support and confidence respectively. In ordered to find the large-1 itemsets, the itemset occurrences will be counted by using the first pass of the algorithm. The candidate sequences are generated first and then it chooses the large sequences from the candidate ones. The scanning of database can be scanned next and candidates support is counted. From the frequent itemsets the association rules will be generated in the second step. Hash Tree is used to store the candidate itemsets. The list of itemsets is stored in the hash-tree node. Apriori Algorithm generates large number of candidate item sets and scans database every time. The total database can be scanned if a new transacting is added to the database.

J. Han et al in [11] 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. Using the FP-Growth the total set of rare patterns can be extracted in Pattern fragment growth. A pattern growth methods applied by T, which avoids costly candidate generation FP-growth is not able to find high utility Items.

W. Wang et al in [23] proposed weighted association rule. In WAR, we discover first frequent itemsets and the weighted association rules for each frequent itemset are generated. In WAR, we use a twofold approach. First it generates frequent itemsets; here we ignore the weight associated with each item in the transaction. In second for each frequent itemset the WAR finds that meet the support, confidence. Weighted association rule mining first proposed the concept of weighted items and weighted association rules. However, the weighted association rules does not have downward closure property, mining performance cannot be improved. By using transaction weight, weighted support can not only reflect the importance of an itemset but also maintain the downward closure property during the mining process.

Liu et al in [15] proposes a Two-phase algorithm for finding high utility itemsets. Two-Phase algorithm, it efficiently prunes down the number of candidates and obtains the complete set of high utility itemsets. 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. In Two-phase, is focused on traditional databases and is not suited for data streams. In ordered to find the temporal high utility itemsets in data streams Two-phase was not proposed. When the new transactions are added from data streams, the data base is scanned twice.

J. Hu et al in [12] defines an algorithm for frequent item set mining, that identify high utility item combinations. In comparisons to the traditional association rule and frequent item mining techniques, which is defined as the combination of few items (rules), which satisfy certain conditions as a group and maximize a predefined objective function? The high utility pattern mining problem considered is different from former approaches, as it conducts "rule discovery" with respect to individual attributes as well as with respect to the overall criterion for the mined set, attempting to find groups of such patterns that combined contribute the most to a predefined objective function.

V.S. Tseng et al in [21] proposes a novel method THUI (Temporal High Utility Itemsets) -Mine for mining temporal high utility itemset mining. The temporal high utility itemsets are effectively identified by the novel contribution of THUI-Mine by generating fewer temporal high transaction weighted utilization 2- itemsets such that the time of the execution will be reduced substantially in mining all high utility itemsets in data streams. In ordered to generate a progressive set of itemsets THUI-Mine employs a filtering threshold in every partition. In this way, the process of finding all temporal high utility itemsets under all-time windows of data streams can be achieved effectively, it needs one more scan over the database. Huge memory requirement and lot of false candidate itemsets are the two problems of THUI- Mine algorithm.

Shankar [19] presents a novel algorithm Fast Utility Mining (FUM) which finds all high utility itemsets within the given utility constraint threshold. In ordered 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

III. Proposed Methods

There are three steps in the framework of proposed system. 1.In ordered to construct a global UP-Tree with the first two strategies we should scan the database twice. 2. From global local UP-Trees and UP- Tree potential high utility itemsets should be generated recursively by UP-GROWTH and 3. From the set of PHUIs identify actual high utility itemsets. To differentiate the patterns found by our methods from HTWUIs we used a new term "potential high utility itemsets" as our methods are not based on traditional TWU model. From the set of HTWUIs the set of PHUIs will become much smaller by our effective strategies.

i. Discarding Global Unpromising Items

In ordered to construct the global UP-Tree the original database can be scanned two times. In order to eliminate the low utility items Discarding global unpromising items (i.e., DGU strategy) is used and their utilities from the transaction utilities. TU of each transaction is computed in the first scan. At the same time each single items TWU is also accumulated. Promising and unpromising are two nodes of a node. With the selection of promising nodes high profits will be given and fewer profits will be given by discarding unpromising nodes. Only the supersets of the subsets of the item are high utility itemsets will give the less quantity.

Transactions are inserted into a UP-Tree during the second scan of database. From the transaction Unpromising items can be removed. A transaction is retrieved and also from the transaction's TU their utilities should be eliminated. After pruning unpromising items generated New TU is called reorganized transaction utility (abbreviated as RTU). The reorganized transaction T of a RTU, is denoted as RTU (Tr).

ii. Decreasing Global Node Utilities

Divide-and-conquer technique is applied in the tree-based framework for high utility itemset mining in mining process. Into smaller subspaces the search space can be divided. Discarding global node utilities (i.e., DGN strategy) during global UP-Tree construction the node utilities which are nearer to UP-Tree root node are effectively reduced. The utilities of the nodes further reduced that are closer to the root of a global UP-Tree By applying strategy DGN. For the databases containing lots of long transactions DGN is especially suitable.

DGN and DGU are applied to construct a global UP-tree. DGU is applied after getting all promising items. By pruning the unpromising items and by sorting the remaining promising items in a fixed order, the transactions are reorganized. Any ordering can be used such as the TWU order or lexicographic, support. The PHUI is similar to TWU, which compute all itemsets utility with the help of estimated utility. Finally, identify high utility itemsets (not less than min_sup) from PHUIs values the global UP-Tree is constructed.

b) UP-Growth+

The basic method for generating PHUIs After constructing a global UP-Tree is to mine. Candidates will be generated too m any. Thus by pushing two more strategies propose an algorithm UP-Growth (Utility Pattern

Growth). By the strategies the itemsets with overestimated utilities can be decreased and further the number of PHUIs can be reduced.

i. Discarding Local Unpromising Items

The global UP-Tree contains many sub paths. Each path is considered from bottom node of header table. This path is named as conditional pattern base (CPB). into conditional UP-Trees strategies DGN and DGU cannot be applied, a global UP- Tree actual utilities of items in different transactions are not maintained. Unless an additional database scan is performed the actual utilities of unpromising items that need to be discarded in conditional pattern bases cannot be known.

Discarding local unpromising items (i.e, DLU strategy) to discarding utilities of low utility items from path utilities of the paths. It reduce the overestimated utilities for second scan by this the complete set of PHUI are found. In the database to keep minimum item utilities for all global promising items we maintain a minimum item utility table. Here bottom entry nodes in header table are traced and nodes which are found traced to root. From the path utility of an extracted path an estimated value for each local unpromising item is subtracted. To reduce overestimated utilities locally is provided by a simple but useful schema without an extra scan of original database.

ii. Decreasing Local Node Utilities

Discarding local node utilities (i.e, DLN strategy) to discarding item utilities of descendant nodes during the local UP-Tree construction. Actual utilities of the descendant nodes cannot be known here. To estimate the discarded utilities we use minimum item utilities. The paths are recognized which are discussed here are by pruning unpromising items by DLU and resorted by a fixed order. And those paths are called reorganized paths. By the two strategies, for itemsets the overestimated utilities can be locally reduced without losing any actual high utility itemset in a certain degree.

Mining a UP-Tree by UP-Growth: by two scans of a conditional pattern base conditional UP-Tree can be constructed. For the first scan, by summing the path utility for each item in the conditional pattern base local unpromising and promising items are learned. During the second scan of the conditional pattern base reduce overestimated utilities DLU is applied. From the path and its path utility items and their estimated utilities are eliminated when a path is retrieved. **B**y the descending order of path utility of the items the path is reorganized in the conditional pattern base .At the time of inserting reorganized paths into a conditional UP-Tree DLN is applied.

Algorithm: UP-Growth+ (TX, H, X)
Input: A UP-Tree TX, a header table H for TX, an itemset X, Transactional database D, user defined threshold value.
Output: high utility itemsets.
Begin

```
1. Scan database for transactions Td ϵ D
2. Determine transaction utility of Td and TWU of itemset (X)
3. Compute min_sup
4. If (TWU(X) ≤ min_sup) then remove items from transaction database
5. Else insert into header table H and to keep the items in the descending order.
6. Repeat step 4 & 5 until end of the D.
7. Insert Td into global UP-Tree
8. Apply DGU and DGN strategies on global UP- tree
9. Re-construct the UP-Tree
10. For each item ai in H do
11. Generate a HUI = X U ai
12. Estimate utility of Y is set as ai's utility value in H
13. Put local promising items in CPB into H
14. Apply strategy DLU to reduce path utilities of the paths
15. Apply strategy DLN and insert paths into Td
```

```
16. If Td ≠ null then call for loop
End for
End
```

iii. Efficiently Identify High Utility Itemsets In UP-GROWTH+

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. After mining the whole UP-Tree by UP- Growth+, we can obtain all PHUIs the number of PHUIs of UP-Growth+ is less than that of UP-Growth. It means the overestimated utilities of itemsets as well as the itemsets, reduces in UP-Growth+.

The third step is to identify high utility itemsets and their utilities after finding all PHUIs from the set of PHUIs by scanning original database once. This step is called phase II. Fewer candidates in phase I are generated by our method, as original database is large and it contains lots of unpromising items scanning original database is still time consuming. By this, scanning reorganized transactions high utility itemsets can be identified in our framework.

IV. Conclusion

In this paper, we have proposed two efficient algorithms named UP-Growth and UP-Growth+ for mining high utility itemsets from transaction databases. UP-Tree data structured is used to store or maintain the information about high utility itemsets. From UP-Tree Potential high utility itemsets can be efficiently generated by two database scans only. In ordered to perform a thorough performance evaluation both real datasets and synthetic datasets were used in the experiments. The performance will be improved by reducing both the search space and the number of candidates. Moreover, the proposed algorithms, UP- Growth+, outperform the state-of-the-art algorithms substantially especially when databases contain lots of long transactions or a low minimum utility threshold is used.

References

[1] R. Agrawal, T.Imielinski, A.Swami, "Mining association rules between sets of items in large databases", in proceedings 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," in Proc. of the 20th VLDB Conf., pp. 487-499, 1994.
[3] R. Agrawal and R. Srikant, "Mining Sequential Patterns," in Proc. of the 11_{th} Int'l Conference on Data Engineering, pp. 3-14, Mar., 1995.
[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," IEEE Transactions on Knowledge and Data Engineering, 21, Issue 12, pp. 1708-1721, 2009.
[5] C. H. Cai, A. W. C. Fu, C. H. Cheng and W. W. Kwong, “Mining Association Rules with Weighted Items,” in Proc.of the Int’l Database Engineering and Applications Symposium (IDEAS 1998), pp. 68-77, 1998.
[6] R. Chan, Q. Yang and Y. Shen. “Mining high utility itemsets,” in Proc. of Third IEEE Int'l Conf. on Data Mining, pp. 19-26, Nov., 2003.
[7] M.-S. Chen, J.-S. Park and P. S. Yu, “Efficient data mining for path traversal patterns,” IEEE Transactions on Knowledge and Data Engineering, Vol. 10, no. 2, pp. 209- 221, 1998.
[8] 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.
[9] J. Han, G. Dong, Y. Yin, “Efficient Mining of Partial Periodic Patterns in Time Series Database,” in Proc. of the Int’l Conf. on Data Engineering, pp. 106-115, 1999.
[10] J. Han and Y. Fu, “Discovery of multiple-level association rules from large databases,” in Proc. 21th VLDB Conf.,Sep. 1995, pp. 420–431.
[11] J. Han, J. Pei, Y. Yin, “Mining frequent patterns without candidate generation,” in Proc. of the ACM-SIGMOD Int'l Conf. on Management of Data, pp. 1-12, 2000.

[12] J. Hu, A. Mojsilovic, “High-utility pattern mining: A method for discovery of high-utility item sets”, Pattern Recognition 40 (2007) 3317 – 3324.

[13] 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,” in Proc. of the 8th IEEE Int'l Conf. on Data Mining, pp. 881-886, 2008.

[14] 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,” in Proc. of the SIAM Int’l Conference on Data Mining (SDM 2005), 2005.

[15] Y. Liu, W. Liao and A. Choudhary, “A fast high utility itemsets mining algorithm,” in Proc. of the Utility-Based Data Mining Workshop, 2005.

[16] B.-E. Shie, V. S. Tseng and P. S. Yu, “Online mining of temporal maximal utility itemsets from data streams,” in Proc. of the 25th Annual ACM Symposium on Applied Computing, Switzerland, Mar., 2010.

[17] K. Sun and F. Bai, “Mining Weighted Association Rules without Preassigned Weights,” IEEE Trans. On Knowledge and Data Engineering, Vol. 20, No. 4, 2008.

[18] S. K. Tanbeer, C. F. Ahmed, B.-S. Jeong and Y.-K. Lee, “Efficient frequent pattern mining over data streams,” in Proc. of the ACM 17th Conference on Information and Knowledge Management, 2008.

[19] S.Shankar, T.P.Purusothoman, S. Jayanthi,N.Babu, A fast algorithm for mining high utility itemsets , in :Proceedings of IEEE International Advance Computing Conference (IACC 2009), Patiala, India, pp.1459-1464

[20] F. Tao, F. Murtagh and M. Farid, “Weighted Association Rule Mining using Weighted Support and Significance Framework,” in Proc. of the ACM SIGKDD Conf. on Knowledge Discovery and Data Mining (KDD 2003), pp.661-666, 2003.

[21] V. S. Tseng, C. J. Chu and T. Liang, “Efficient Mining of Temporal High Utility Itemsets from Data streams,” in Proc. of ACM KDD Workshop on Utility-Based Data Mining Workshop (UBDM’06), USA, Aug., 2006.

[22] V. S. Tseng, C.-W. Wu, B.-E. Shie and P. S. Yu, “UPGrowth: An Efficient Algorithm for High Utility Itemsets Mining,” in Proc. of the 16th ACM SIGKDD Conf. on Knowledge Discovery and Data Mining (KDD 2010), pp. 253-262, 2010.

[23] W. Wang, J. Yang and P. Yu, “Efficient mining of weighted association rules (WAR),” in Proc. of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD 2000), pp. 270-274, 2000.

[24] H. Yao, H. J. Hamilton and L. Geng, “A unified framework for utility-based measures for mining itemsets,” in Proc. of ACM SIGKDD 2nd Workshop on Utility- Based Data Mining, pp. 28-37, USA, Aug., 2006.

[25] C.-H. Yun and M.-S. Chen, “Using pattern-join and purchase-combination for mining web transaction patterns in an electronic commerce environment,” in Proc. of 24th IEEE Annu. Int. Computer Software and Application Conf., pp. 99–104, Oct., 2000.

[26] B Adinarayanareddy, O Srinivasa Rao and MHM Krishna Prasad, “ An Improved UP-Growth High Utility Itemset Mining”, in proc of International Journal of Computer Applications, Volume 58– No.2, November 2012.

[27] C.C Switi and Vijay K.V, “Mining High Utility Item Set From Large Database: - A Recent Survey “, in proc of International Journal of Emerging Technology and Advanced Engineering, Volume 3, Issue 5, May 2013.

SADAK SRILEKHA Received B.Tech from JNTU Hyderabad in the year 2011, I currently M.Tech student in computer science department From kshatriya college of engineering and research interested in Cloud computing and Data Mining.

MADDIKUNTA LAXMAN M.Tech (CSE) having 10 years of experience in teaching, Present working as head of the department of CSE in KSHATRIYA COLLEGE OF ENGINEERING, Armoor Interested in data mining, DBMS and Operating System.