Volume 5, Issue 2, February 2015 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

A Survey on Mining Maximal High Utility Itemsets from Transactional Databases

Geetika Narang, Vishakha Singh, Renu Rawat
Department of CSE, Pune University
India

Abstract- Data mining is a wide spreading research topic with its frequent applications in online e-business and web click stream analysis.Mining high utility itemsets from a transactional database relates to the discovery of itemsets with high utility like profits or gains. In data mining, high utility itemsets are an essential aspect to be considered while analyzing profits. There have been a large amount of research that solved the problems of generating highly frequent and high utility itemsets. But these algorithms largely generate huge number of candidate itemsets. This in turn affects the performance in terms of run time and memory requirements. It results in inaccurate output when there is need of large datasets. For this we need to generate long utility patterns. In this paper we present an overview of different models developed to find maximal high utility itemsets. The purpose of the paper is to give a top view of the different models published.

Keywords: High Utility Itemsets, frequent itemsets, tree structure, data mining, transactional database

I. INTRODUCTION

Data mining is a booming area of research in today's era. Data mining helps to produce profitable conclusions from unstructured and structured data. It is concerned with examining of large volumes of data to automatically find interesting similarities or relations which is proportional to better understanding of the underlying processes. Data mining actions use combination of techniques from database like artificial intelligence, statistics, and technologies based machine learning. Data mining is symbolized as knowledge mining from data.

Utility Mining is among one of the most difficult data mining activity which is the mining of high utility itemsets efficiently. Discovering the itemset with high utilities is called as Utility Mining. A high utility itemset is an itemset which is used frequently and is a profitable itemset, also it is measured according to user preference utility or other expressions. The researchers came up with the idea of utility based mining which involves a user to freely express his or her view for the usefulness of itemsets as utility values and among them find the high utility values greater than threshold due to the limitations of frequent and rare itemsets. The term utility is the quantitative measure of user preference that is user's view about the utility value of itemset.

Mining high utility itemsets from databases refers to discovering itemsets with high profits or high utility values. The meaning of itemset utility is profitability, characteristics or importance of an item in user's point of view or users need. A high utility itemset can be elaborated as: A bunch of itemsets in a transactional database. This itemset in a transactional database includes two concepts: Firstly, itemsets in a single transaction are called Internal utility and Secondly, itemsets in multiple transaction are called External utility. High utility itemsets mining is growing with the more innovative mining techniques with wider applications are being developed. Mining high utility itemsets from transactional databases is very important and has wide range of applications like online e-commerce management ,website click stream analysis[13,16,21], mobile commerce environment planning, business promotion in chain hypermarkets, cross marketing in retail stores[4,9,14,22,24] and even in finding important patterns in biomedical applications.

Frequent itemset mining [2] refers to discovering itemsets that occur in a transactional database beyond the users given frequency threshold, without involving the profit or quantity of the item. The itemsets which occur again and again or frequently are called frequent itemsets. The objective of frequent item set mining is identification of all the itemsets which occur frequently. An itemset can be defined as a non-empty set of items also an itemset with m different items is called as m-itemset. Taking an example {bread, egg, butter, cheese} may denote a 3-itemsets in a supermarket transaction. The concept of frequent database was initiated by Agrawal et al [2].

Over the past few years the task of *finding frequent patterns* in large databases has gained importance and also used in wide areas of application. This technique is computationally more expensive especially when large numbers of patterns are involved. It becomes difficult for user to choose most interesting patterns when multiple patterns are present. The aim of frequent itemsets is to find the most occurring itemset. Once frequent itemsets are identified formation of association rule are straight forward. In real time perhaps each item in a supermarket has a variable price/importance and each customer will be interested in buying multiple copies of the same item. Therefore, only finding traditional frequent patterns in not sufficient enough to measure the requirement of finding most valuable customers/itemsets that contribute to most of the profit in retail business.

II. LITERATURE SURVEY

R. Agrawal et al in [2] proposed Apriori algorithm which is used to obtain frequent itemsets from databases. Apriori is a classic algorithm for frequent item set mining and learning association rules over transactional databases. In mining association rules [1] a problem occurs that is to generate all association rules that have the support and confidence greater than stated by the user's minimum support and confidence specifically. The first pass simply counts the itemsets occurred to find the large 1-itemset .This is done by generating the candidate sequence first and then choosing the large sequences from the candidate ones. After that the database is scanned and support of the candidate is counted. Second pass involves generation of association rules from frequent itemsets. Candidate itemsets are stored in hash-tree. The hash-tree contains either hash table or a list of itemsets. After large set identification only those set which support greater than minimum are allowed. Apriori generates a lot of candidate itemsets and also scans the database every time a new transaction is made to the database.

J. Han et al in [11] proposed frequent pattern tree *(FP tree) structure*. It is an extended prefix tree structure used for storing important information about frequent patterns, which are compressed and used to develop an efficient FP-tree based mining method. The complete set of frequent pattern is mined by pattern fragment growth using the FP-growth. It constructs a highly compact tree (FP tree), which is substantially smaller than the original database, this helps in saving costly database scans in subsequent mining processes. It applies a mechanism which generates a pattern growth method that prevents costly candidate generation. FP-growth is in capable to find high utility itemsets.

W. Wang et al in [23] proposed weighted association rule (WAR). In WAR, first we discover all the itemsets and then the weighted association rule corresponding to each frequent itemset is generated. Hence, in WAR we use two fold approach. First it generates frequent itemsets ignoring the weight associated with each itemset in this transaction. Second, for each itemset WAR finds out the support and confidence. WAR first discovered weighted items. But WAR does not have downward closure property, so mining performance cannot be improved. By using transaction weight, weighted support gives the importance of itemset and also maintains the downward closure property while mining.

Liu et al in [15] proposed a two phase algorithm for finding high utility itemsets. It efficiently prunes down the number of candidate itemsets living with us high utility itemsets. In Phase I, only the combination of high transaction weighted utility itemsets are included into candidate set at each level of level wise searching. In Phase II only a single extra database scan is done to eliminate overestimated itemsets. Two phase requires fewer database scans,less memory and less computational cost. Two phase is best suited for traditional database and not for data streams.

Li et al in [13] proposed two efficient one pass algorithms MHUI-BIT and MHUI-TID for mining high utility itemsets from data streams within a transaction sensitive sliding window. To improve the efficiency of mining high utility itemsets two effective representations of an extended tree-based summary data structure and itemset information were developed.

V.S Tseng et al in [21] proposed a method THUI (Temporal High Utility Itemsets)-Mine. The THUI are identified for their contribution of THUI-Mine by generating fewer temporal high transaction weighted itemsets such that the execution time period will be reduced in mining all high utility itemsets in data stream. To generate high progressive itemsets THUI-Mine sets a filter in each partition. This results in finding all THUI under time windows of data stream can effectively achieved. Large memory requirement and lots of false candidate itemsets are two problem of THUI-Mine algorithm.

J. Hu et al in [12] proposed an algorithm which identifies high utility item combinations in frequent item set mining. This algorithm is to find segment of data, which is defined with combination of few items that is rules, a predefined objective function and satisfy certain conditions as a group. The approach of this algorithm is different from other pattern mining methods as it conducts rule discovery with respect to the overall criterion for the mined set as well as individual attributes.

Erwin et al in [8] observed that the conventional candidate-generate-and-test approach for identifying high utility itemsets is not suitable for dense date sets. The high utility itemsets are mined using the pattern growth approach in the novel algorithm called CTU-Mine.

Shankar [19] proposed a novel algorithm Fast Utility Mining (FUM) which finds out all high utility itemsets within the given limited utility constraint threshold. For generating different types of itemsets the author also suggested techniques 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).

Cheng-Wei Wu et al in [22] proposed a novel algorithm with a compact data structure for efficiently finding high utility itemsets from transactional databases. Depending on the construction of a global UP-tree the high utility itemsets are generated using UP-Growth which is one of the efficient algorithms. In phase-I three steps are followed by framework of UP-tree as: (i) Construction of UP-tree, (ii) Generation of PHUIs from the UP-Tree and (iii) The high utility itemsets should be identified using PHUI.

Global UP-tree construction involves the following:

(i)Eliminating the low utility items and their utilities from transaction utilities by discarding global unpromising items(i.e DGU technique),(ii) During construction of UP-tree global node utilities(i.e DGN technique) are discarded, the node nearer to UP-tree root node are effectively reduced by DGN strategy. The PHUI is similar to TWU, in which the itemsets utility is calculated with the help of approximated utility and from PHUIs value the high utility itemsets are identified. The Global UP-tree consists of many sub paths, from the bottom node of header table each path is considered. This path is called Conditional Pattern Base (CPB).

Table1 Review of different algorithms

AUTHOR	PROPOSED ALGORITHM	CHARACTERISTICS	ISSUES
R.Agrawal et. Al	Apriori	Frequent and candidate itemsets, association rules	Large candidate itemsets generation and rescan database every time
J. Han et. al	FP-growth	Frequent itemsets without candidate key generation and less time.	Incapable to find high utility itemsets
W. Wang et. Al	WAR	Items with support and confidence, weights for Items.	Downward Closure property and no high priority Data.
Liu et. Al	Two Phase	High utility itemset in Traditional database , less candidates	Rescan database, no Temporal itemsets
Tseng et. Al	THUI-Mine	Generates few candidate and high performance	Lots of false candidate itemsets
Li et. Al	MHUI-BIT & MHUTTID	Item information, HTU for data stream	More time and candidate test fails
J. Hu et. al	High yield partition Tree	Binary tree partition, iterations to prune item	Lots of low utility values
Erwin et. Al	CTU-Mine	High utility itemset for pattern growth and dense data	Overestimated real utility
V.S. Tseng et. Al	UP-Growth	Pruning candidate itemset with two scans	Performance

III. CONCLUSION

This paper demonstrates a survey on different High Utility Itemset mining algorithms that were proposed by researchers earlier for better development in the field of Data Mining. The multiple algorithms discussed above will be of great use for developing a new improved technique for mining high utility item sets which is efficient and effective. In future we will be developing an algorithm for Mining Maximal High Utility Itemsets from Transactional Databases.

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 11th 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*, Vol. 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: Amethod 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 timesensitive 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, "UP Growth: 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 24thc IEEE Annu. Int. Computer Software and Application Conf.*, pp. 99–104, Oct., 2000.

