International Journal of Scientific Engineering and Technology (ISSN : 2277-1581)
Volume No.3 Issue No.11, pp : 1339-1344 01 Nov. 2014

Mining High Utility Itemsets – A Recent Survey

U Kanimozhi[1], J K Kavitha[2], D Manjula[3]

[1,2]Research Scholar, Department of Computer science and Engineering, Anna University, Chennai, India
[3] Professor, Department of Computer science and Engineering, Anna University, Chennai, India
[1]kanimozhiu.03@gmail.com, [2]jeyakumar.kavitha@gmail.com, [3]manju@annauniv.edu.

Abstract: Association rule mining (ARM) plays a vital role in data mining. It aims at searching for interesting pattern among items in a dense data set or database and discovers association rules among the large number of itemsets. The importance of ARM is increasing with the demand of finding frequent patterns from large data sources. Researchers developed a lot of algorithms and techniques for generating association rules. Since the traditional ARM approaches consider the utility of the items by its presence in the transaction set. An emerging topic in the field of data mining called Utility Mining has evolved. Since the frequency of itemset is not sufficient to reflect the actual utility of an itemset. The main objective of Utility Mining is to identify the itemsets with highest utilities, by considering profit, quantity, cost or other user preferences. Mining High Utility itemsets from a transaction database is to find itemsets that have utility above a user-specified threshold. Mining Itemset Utility is an extension of Frequent Itemset mining, which discovers itemsets that occur frequently. Several researches about itemset utility mining were proposed. Here, a literature survey of various algorithms for high utility itemset mining has been presented.

Keywords: Data Mining, Association Rule Mining, Frequent Pattern Mining, Utility Mining and High Utility Itemsets

I. INTRODUCTION

Data Mining

Data mining and knowledge discovery is the process of discovering and extracting information or pattern, revealing potentially useful information from large databases. Among discovering many kinds of knowledge in database, Association rules mining was a form of data mining to extract interesting correlations, frequent patterns, associations or casual structures among sets of items in the databases. Discovering useful patterns hidden in a database plays an essential role in several data mining tasks, such as frequent pattern mining, weighted frequent pattern mining and high utility pattern mining. Among them, frequent pattern mining is a fundamental research topic that has been applied to different kinds of databases, such as transactional databases, streaming databases and time series databases, and various application domains, including decision support, market strategy, financial forecast, bioinformatics, web click-stream analysis, and mobile environments.

Association Rule Mining

Mining Association Rules is one of the most challenging fields of data mining which was introduced in Agarwal et al., (1993) to discover the interesting associations, correlations, co-occurrences, frequent patterns among the different item sets in a database. Mining frequent item sets is the fundamental and essential part in many data mining applications, such as the discovery of association rules, sequential patterns, and so on. Several algorithms like Apriori (Agrawal et. al., 1993), AprioriTID (Agrawal and Srikant, 1994), Partition Algorithm (Savasere et al.,1995), DIC (Brin et al., 1997), FP-tree (Han et al., 2000), CT-PRO (Y G Sucahyo et al., 2004), CTU-Mine (A Erwin et al., 2007), HUC-Prune (Ahmed C F et al., 2011) etc. have been developed to meet the requirements of this problem. Mining association rules can be decomposed into two steps: the first is generating frequent itemsets. The second is generating association rules. The main challenge in association rule is to identify frequent itemsets. Finding frequent itemset is one important step in association rule mining. Since the solution of second sub-problem is straightforward, most of the researchers had focus on how to generate frequent itemsets.

Frequent Pattern Mining

Frequent itemsets are the itemsets that occur frequently in the transaction database. The objective of Frequent Itemset Mining is to identify all the frequent itemsets in a transaction database. The initial solution of frequent pattern mining, candidate set generation-and-test paradigm of Apriori (Agrawal et. al., 1993) has revealed many drawbacks including that it requires multiple database scans and generates many candidate itemsets. FP-Growth (Han et al., 2000) which is based on pattern growth method was afterward proposed to achieve a better performance than Apriori-based methods. In the FP-Growth algorithm, a tree structure, called FP-Tree, is used. FP-Growth solved this problem by introducing a prefix-tree (FP-tree)-based algorithm without candidate set generation and-testing. Although frequent pattern mining plays an important role in data mining applications, its two limitations are, first, it treats all items with the same importance/weight/price and, second, in one transaction, each item appears in a binary (0/1) form, i.e., either present or absent. However, in the real world, each item in the supermarket has a different importance/price and one customer can buy multiple copies of an item.

Moreover, items having high and low selling frequencies may have low and high profit values, respectively. For example, some frequently sold items such as bread, milk and pen may have lower profit values compared to that of infrequently sold higher profit value items such as gold, platinum and diamond. Therefore, finding only traditional frequent patterns in a database cannot fulfill the requirement of finding the most valuable itemsets/customers that contribute to the major part of the total profits in the real world retail database. This gives the motivation to develop a mining model to discover the itemsets/customers contributing to the majority of the profit. A frequent itemset is the itemset having frequency support greater a minimum user specified threshold.

Among the five rule constraints, the anti-monotone and monotone constraints are satisfied in MFS_DoubleCons (Duonga et al., 2014) was proposed for mining frequent patterns. It generates all frequent patterns quickly and distinctly with the constraints. In the framework of frequent pattern mining, meanwhile, various kinds of databases are also considered such as sequential databases (Chang et al., 2009), incremental databases (Leung et al., 2007), and stream databases (Yun et al., 2014). Max- Freq-Miner (Caldersa et al., 2014) finds frequent itemsets in a data stream by maintaining a very compact summary of the stream.

Weighted Frequent Pattern Mining

Although frequent pattern mining has played an important role in pattern mining field, the importance of items and item quantities in transactions are not considered in contrast to real world retail databases. In the framework of frequent itemset mining, the relative importance of items to users is not considered. Thus, weighted association rule mining came to address this problem (Cai et al., (1998), Sun and Bai (2008), Tao et al., (2003), Wang et al., (2000), Yun (2008), Yun and Leggett (2005), Yun and Leggett (2006)). Cai et al. (1998) first proposed the concept of weighted items and weighted association rules. However, since the framework of weighted association rules does not have downward closure property, hence mining performance cannot be improved. To address this problem, Tao et al., (2003) proposed the concept of weighted downward closure property. By using transaction weight, weighted support cannot only reflect the importance of an itemset but also maintain the downward closure property during the mining process. Also Chang (2011) have developed weighting functions for weighted pattern mining.

Weighted frequent pattern mining (Yun and Ryu, 2013 and Yun et al., 2014) emerged to reflect the relative importance of items in databases. In the framework of weighted frequent pattern mining, weight of a pattern is the ratio of the sum of weight values of items in the pattern to its length. Also, WIT-FWI (Vo et al., 2013) a tree-based algorithm for discovering frequent weighted itemsets with a tree structure, called WIT-Tree, based on the concept of weight, where relative importance of items are considered. In addition, to satisfy downward closure property (Agrawal & Srikant, 1994), overestimated weights are applied in the framework. Nevertheless, the non-binary occurrence of items in transactions is not considered. Although weighted association rule mining considers the importance of items, in some applications, such as transaction databases, items quantities in transactions are not taken into considerations yet.

Utility Mining

The traditional ARM approaches consider the utility of the items by its presence in the transaction set. The frequency of itemset and weighted association rule mining are not sufficient to reflect the actual utility of an itemset. Now, one of the most challenging data mining tasks is the mining of high utility itemsets efficiently. Identification of the itemsets with high utilities is called as Utility Mining. The utility can be measured in terms of cost, profit or other expressions of user preferences. For example, a smart phone may be more profitable than a mobile phone in terms of profit. The main objective of high-utility itemset mining is to find all those itemsets having utility greater or equal to user-defined minimum utility threshold.

Utility mining (H. Yao, H.J. Hamilton, and C.J. Butz 2004) model was defined to discover more important knowledge from a database. We can measure the importance of an itemset by the concept of utility (profit). We can handle the dataset with non-binary frequency values of each item in transactions, and also with different profit values of each item. Therefore, utility mining represents real world market data. By utility mining, several important business area decisions like maximizing revenue or minimizing marketing or inventory costs can be considered and knowledge about itemsets/customers contributing to the majority of the profit can be discovered. In addition to our real world retail market, if we consider the biological gene database and Web click streams, then the importance of each gene or Web site is different and their occurrences are not limited to a 0/1 value. Other application areas, such as stock tickers, network traffic measurements, Web server logs, data feeds from sensor networks, and telecom call records can have similar solutions.

MEU (Yao et al., 2004), Two-Phase (Y. Liu et al., 2005), UMining and Umining_H (Yao et al., 2006), IIDS (Y.C. Li et al., 2008), IHUP (C. F. Ahmed et al., 2009), TWU-Mining (B.Vo et al., 2009), UP-Growth (V. S. Tseng et al., 2010), UP-Growth$^+$ (V. S. Tseng et al., 2013) etc. are some of the existing algorithms which attempt to discover the high utility itemset from large datasets. Several other works are also found in Chan et al. (2003), Y G Sucahyo et al., (2004), Erwin A et al. (2007), Yen S J et al.,(2007), Chithra R et al., (2011), Ahmed C F et al.,(2011), that has given some attention to the high utility item set mining problem. The goal of Frequent Itemset Mining is to find items that co-occur above a user given value of frequency, in the transactional database. In the Frequent Itemset Mining problem, the occurrence of each item in a transaction is represented by a binary value without considering its quantity or an associated weight such as price or profit. However, quantity and weight are significant for addressing real world decision problems that require maximizing the utility in an organization.

Interestingness measures can play an important role in knowledge discovery. These measures are intended for selecting and ranking patterns according to their potential interest to the user. In practice, the frequency of occurrence may not express the semantics of applications, because the user's interest may be related to other factors, such as cost, profit, or aesthetic value. Utility based data mining refers to allowing a user to conveniently express his or her perspectives concerning the usefulness of patterns as utility values and then finding patterns with utility values higher than a threshold. The solution is to survey different aspects which are discussed in the several research papers and after analyzing those research papers, conclude a solution which is best in efficiency and performance. This paper provides an overview of some techniques that are used to improvise the efficiency of mining high utility itemsets over a transactional database with their strength and weakness.

II. IMPROVING THE EFFICIENCY OF HIGH UTILITY ITEMSET MINING

The main goal of Association Rule Mining (ARM) is to discover the interesting associations or relations among the different itemsets in database. Interestingness measures can play an important role in knowledge discovery. These measures are intended for selecting and ranking patterns according to their potential interest to the user. Hence the factors to be considered while improving the efficiency of High Utility Itemset Mining are as follows:

- Reducing the number of scans in the original database.
- Minimize memory utilization (Reducing the search space).
- Reducing the total execution and computation time.
- Reducing the resource utilization.
- Increase the performance in terms of time and space complexity.

III. HIGH UTILITY ITEMSET MINING APPROACHES

Utility mining over dense dataset is a challenging area of research for the data mining researchers. Several recent works can be found in the survey to meet this challenge. Some of the extensive studies of utility mining are (Erwin et al., 2008; Hong et al., 2011; Lin et al., 2011; Lin et al., 2012; Shie et al., 2011; Shie et al., 2013; Weiss et al., 2008; Wu, Lin, Yu, and Tseng, 2013; Wu, Shie, Tseng, and Yu, 2012; Yeh et al., 2007; Yin, Zheng, and Cao, 2012). In the framework of utility mining, maintaining downward closure property is a difficult task, and TWU (Transaction Weighted Utilization) model has defined (Liu & Choudhary 2005), which is an overestimated method.

A. MEU (Mining with Expected Utility)

In the precedent, Han, Pei, and Yin (2000) proposed the Frequent-Pattern tree (FP-tree) structure for efficiently mining association rules without generation of candidate itemsets. The FP-tree (Han et al., 2000) was used to compress a database into a tree structure which stored only large items. They must process all the transactions in a batch way. In frequent pattern mining field, past researches consider the importance of items uniformly. Thus, a new topic is raised for conquering this problem, that is, utility mining. In utility mining, each item may have different importance, such as profits and degree of user interest. The importance is generally called utility. Chan et al., (2003) first proposed the problem of utility mining. High utility pattern mining finds all itemsets in a transaction database with utility value greater or equal to the user specified minimum utility threshold. It also discovers the semantic significance among items in the mining process. Hence, Yao et al. (2004) proposed a framework for high utility itemset mining and theoretical model called Mining with Expected Utility (MEU). This model cannot maintain downward closure property, and heuristic technique was used to determine candidate set.

B. Two-Phase Algorithm

Liu et al. (2005) proposed the Two-Phase algorithm, based on Apriori algorithm, discovers high utility itemsets and uses the transaction-weighted downward closure property to maintain downward closure property in utility mining. Although Two-Phase algorithm can reduce the search space of utility mining, it still generates too many candidates. Transaction-weighted utilization (TWU) is defined and then proved that, it is possible to maintain the downward closure property and also items' quantities in transactions are taken into considerations. For the first database scan, the algorithm finds all the one-element transaction-weighted utilization itemsets, and based on that result, it generates the candidates for two element transaction-weighted utilization itemsets. In phase I, it employs an Apriori-based level-wise method to enumerate HTWUIs. Candidate itemsets with length k are generated from length k-1 HTWUIs and their TWUs are computed by scanning the database once in each pass. After the above steps, the complete set of HTWUIs is collected in phase I. In the second database scan, it finds all the two-element transaction weighted utilization itemsets, and based on that result; it generates the candidates for three-element transaction weighted utilization itemsets, and so on. i.e., in phase II, HTWUIs that are high utility itemsets are identified with an additional database scan. At the last scan, the Two-Phase algorithm determines the actual high utility itemsets from the high transaction-weighted utilization itemsets. This algorithm suffers from the same problem of the level-wise candidate generation-and-test methodology. However, the Two-Phase demands multiple database scans and generates a huge number of candidate itemsets because of a level-wise method.

C. UMining and Umining_H

Yao et al. (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.

D. IIDS - Isolated Items Discarding Strategy

To reduce the number of candidates, FUM and DCG + algorithms based on IIDS (Isolated Items Discarding Strategy) were proposed by Li et al., (2008). Isolated Items Discarding Strategy (IIDS) discovers high utility itemsets and reduces the number of candidates in every database scan. IIDS shows that itemset share mining (Barber et al., 2003) problem can be directly converted to the utility mining problem by replacing the frequency value of each item in a transaction by its total profit, i.e., multiplying the frequency value by its unit profit. However, this algorithm still suffer level-wise candidate set generation-and-test problem of Apriori and require multiple database scans. Here, Transaction weighted utility model is efficient in terms of (1) Fewer candidates set (2) Accuracy and

(3) Less arithmetic complexity compared to Yao et al's., (2006) UMining and Umining_H. Although they can decrease the number of candidates, they still need multiple database scans and apply a candidate generation-and-test approach.

E. IHUP- Information of High Utility Pattern Mining Algorithm

To efficiently generate high utility itemsets and to void multiple database scans, IHUP (Ahmed et al., 2009) was proposed. It uses three tree structures, IHUPL-Tree, IHUPTF Tree, and IHUPTWU-Tree, which are based on FP-Tree. Each node in the trees is composed of an item name, a support count, and a TWU value. IHUP generates all high utility itemsets from the IHUP-Tree through three steps.

Step 1:
Items in transactions are sorted according to lexicographic order, and the transactions are inserted into IHUPL-Tree with a single database scan. For a single-pass tree construction, the constructed tree can be restructured without additional database scan by arranging nodes by support descending order (IHUPTF-Tree) or TWU descending order (IHUPTWU-Tree).

Step 2:
Candidate itemsets are extracted from IHUP-Tree by FP-Growth (Han et al., 2005) algorithm.

Step 3:
Actual high utility itemsets are identified with an additional database scan.

Although IHUP can construct a tree and discover high utility itemsets with two database scans, it generates a large number of candidates by applying the TWU model.

F. UP-Growth Algorithm

To address issue of generating a large number of candidates, UP-Growth (V.S Tseng et al., 2010) has recently been proposed and it uses PHU (Potential High Utility) model. For reducing the number of candidate itemsets, the UP-Growth applies four strategies, DGU (Discarding Global Unpromising items), DGN (Decreasing Global Node utilities), DLU (Discarding Local Unpromising items), and DLN (Decreasing Local Node utilities). Besides, it constructs a tree structure, named UPTree, with two database scans and conducts mining high utility itemsets. In other words, it demands three database scans for discovering high utility itemsets. In the first database scan, TWU values of each item are accumulated. In the second database scan, items having less TWU values than the user-specified minimum utility threshold are removed from each transaction. In addition, items in transactions are arranged according to TWU descending order and the transactions are inserted into the UP-Tree. In this stage, DGU and DGN are applied for reducing overestimated utilities. After that, high utility itemsets are generated from the UP-Tree with DLU and DLN.

G. CHUD and DAHU

An algorithm called CHUD and DAHU (Wu & Yu, 2011) based on the concept of closed pattern were proposed. CHUD conducts mining Closed+ high utility itemsets based on closed pattern and DAHU recovers all high utility itemsets from the Closed+ high utility itemsets. Although it can perform mining high utility itemsets faster than the UP Growth by using a method for reducing the number of candidates, it needs much more amount of memory usage since it uses a database converted into a vertical database.

H. List-Based Algorithms

Recently, list-based algorithms (Liu & Qu, 2012; Liu et al., 2012) have been proposed for mining high utility itemsets. HUI-Miner (Liu & Qu 2012) discovers high utility itemsets with a list data structure, called utility list. It first creates an initial utility list for itemsets of the length 1 for promising items. Then, HUI-Miner constructs recursively a utility list for each itemset of the length k using a pair of utility lists for itemsets of the length k-1. For mining high utility itemsets, each utility list for an itemset keeps the information of TIDs for all of transactions containing the itemset, utility values of the itemset in the transactions, and the sum of utilities of remaining items that can be included to super itemsets of the itemset in the transactions.

I. UP-Growth+ Algorithm

UP-Growth+ algorithm proposed by V.S. Tseng et al., (2013) for reducing overestimated utilities is more effective than UP-Growth. Since more decreased overestimated utilities are in local UP-Trees of UP-Growth, hence UP-Growth+ (V.S Tseng et al., 2013) was proposed to address this problem. It stores a minimal utility value to each corresponding node. Moreover, UP-Growth+ reduces the overestimated utilities with the strategies DNU (Discarding local unpromising items and their estimated Node Utilities) and DNN (Decreasing local Node utilities for local UP-Tree by estimated utilities of descendant Nodes). Hence DNU and DNN are the improved strategies of DLU and DLN respectively. Moreover, this algorithm generates fewer candidates.

IV. FLOW DIAGRAM

The diagram represents the complete process of finding and displaying the high utility itemsets. Here, comparing with threshold value gives the frequent utility itemsets (promising itemsets) as the results.

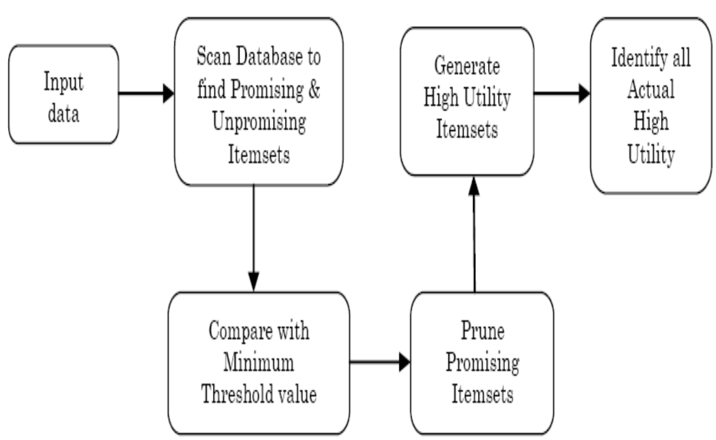

Figure 1: Data Flow in Utility Mining Process

V. COMPARATIVE STUDY

Following table shows the comparative study between some of the methods for mining high utility itemsets that are mentioned above. The performance of MEU and UMining and Umining_H is poor because they need multiple scanning over the original datasets and they cannot work efficiently if they applied on dense datasets. The algorithms Two Phase and IIDS have average performance since it use the transaction-weighted downward closure property to maintain downward closure property in utility mining. Although these algorithms can reduce the search space of utility mining, it still generates too many candidates and it works efficiently with sparse datasets as compared to MEU. The performance of IHUP, UP-Growth and UP-Growth+ is good while the performance of Two Phase and IIDS is average because it is based on Apriori algorithm and suffers from the problem of level-wise candidate generation-and-test strategy.

TABLE I
COMPARISON BETWEEN VARIOUS METHODS

Name of Method	Scanning Method/ Concept Used	Nature of Dataset	Performance
MEU	Multiple Scanning/ FP-Growth tree based	Sparse data	Poor
UMining and Umining_H	Multiple Scanning/ Apriori based	Sparse data	Poor
Two Phase	Multiple Scanning/ Apriori based	Sparse data	Average
IIDS	Multiple Scanning/ Apriori based	Sparse data	Average
IHUP	Scans the original database once/ FP-Growth tree based	Dense data	Good
UP-Growth	2 Scan/ FP-Growth tree based	Dense data	Good
CHUD and DAHU	Based on Closed Pattern Mining Concept.	Sparse data	Good
UP-Growth+	2 Scan/ FP-Growth tree based	Dense data	Good

VI. 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. Utility Mining covers all aspects of economic utility in data mining and helps in detection of itemset having high utility. High Utility itemset mining is very beneficial in several real-life applications. From the above discussion and analysis of the existing techniques of mining High utility itemsets, it can be observed that each algorithm can extract HUIs from database with their pros and cons. The best technique for mining high utility itemsets depends on factors like reducing the search space, reducing the number of scans on original database, improving performance etc., but some of the existing algorithms works well only when the number of transactions are limited. Those techniques failed to handle huge number of transactions or dense datasets. Hence it is important in developing techniques in such a way that interesting rules are mined effectively from dense datasets. 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

i. *R Agrawal and R Srikant 1994. Fast algorithms for mining association rules. In Proc. of the 20th Int'l Conf. on Very Large Data Bases, pp. 487-499.*

ii. *C F Ahmed, S K Tanbeer, B S Jeong and Y K Lee 2009. Efficient tree structures for high utility pattern mining in incremental databases. In IEEE Transactions on Knowledge and Data Engineering, Vol. 21, Issue 12, pp. 1708-1721.*

iii. *R Chan, Q Yang, and Y Shen 2003. Mining high utility itemsets. In Proc. of Third IEEE Int'l Conf. on Data Mining, pp. 19-26.*

iv. *A. Erwin, R P Gopalan and N R Achuthan 2008). Efficient mining of high utility itemsets from large datasets. In Proc. of PAKDD, LNAI 5012, pp. 554-561.*

v. *Jiawei Han, Jian Pei and Y Yin 2000. Mining frequent patterns without candidate generation. In Proc. of the ACM-SIGMOD Int'l Conf. on Management of Data, pp. 1-12.*

vi. *Y C Li, J S Yeh and C C Chang 2008. Isolated items discarding strategy for discovering high utility itemsets. In Data & Knowledge Engineering, Vol. 64, Issue 1, pp. 198-217.*

vii. *Y Liu, W Liao and A Choudhary 2005. A fast high utility itemsets mining algorithm. In Proc. of the Utility-Based Data Mining Workshop.*

viii. *H Yao, H J Hamilton and L Geng 2006. 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.*

ix. *Vincent S Tseng, C W Wu, B E Shie and P S Yu 2010. UP-Growth: An Efficient Algorithm for High Utility Itemset Mining. In Proc. of ACM-KDD, Washington, DC, USA, pp. 253-262.*

x. *H Yao and H J Hamilton 2006. Mining itemset utilities from Transactional databases", Data & Knowledge Engineering", 59, pp. 603-626.*

xi. *Han J, Pei J and Yin Y 2000. Mining frequent patterns without candidate generation. In Proc of the ACM-SIGMOD int'l conf on management of data, pp 1–12.*

xii. *Agrawal R and Srikant R 1995. Mining sequential patterns. In Proc of 11th int'l conf on data mining, pp 3–14.*

xiii. *Ahmed C F, Tanbeer S K, Jeong B S and Lee Y K 2011. HUC-Prune: An efficient candidate pruning technique to mine high utility patterns. Appl Intell 34(2):181–198.*

xiv. *Liu Y, Liao W K and Choudhary A 2005.A fast high utility itemsets mining algorithm. In Proc of utility-based data mining.*

xv. *qErwin, R P Gopalan, and N R Achuthan 2007. CTU-Mine: An Efficient High Utility Itemset Mining Algorithm Using the Pattern growth Approach. Proc. Seventh IEEE Int'l Conf. Computer and Information Technology (CIT '07), pp. 71- 76.*

xvi. *A Savasere, E Omiecinski and S Navathe 1995. An efficient algorithm for mining association rules in large databases. Proc. of the 21st Int'l Conf. on VLDB, pp. 432-444.*

xvii. *Y G Sucahyo and R.P. Gopalan 2004. CT-PRO: A bottom-up non recursive frequent itemset mining algorithm using compressed FP -tree data structure. Proc. of the IEEE ICDM Workshop on Frequent Itemset Mining Implementations (FIMI), Brighton, UK.*

xviii. *Yao H, Hamilton, H J, Buzz and C J 2004. A Foundational Approach to Mining Itemset Utilities from Databases. In 4th SIAM International Conference on Data Mining. Florida, USA.*

xix. *Alva Erwin, Raj P Gopalan and N R Achuthan 2007. A Bottom-Up Projection Based Algorithm for Mining High Utility Itemsets. 2nd Workshop on Integrating AI and Data Mining (AIDM 2007), Gold Coast, Australia, Conferences in Research and Practice in Information Technology (CRPIT), Vol. 84.*

xx. *Hong Yao and Horward J Hamilton 2006. Mining itemset utilities from transaction databases. Data & Knowledge Engineering, Volume 59, pp 603-626.*

xxi. *S Shankar, T P Purusothoman S Jayanthi and N Babu 2009. A fast agorithm for mining high utility itemsets. In Proceedings of IEEE International Advance Computing Conference, Patiala, India, pp.1459-1464.*

xxii. *Vincent S Tseng, Bai-En Shie, Cheng-Wei Wu and Philip S Yu 2013. Efficient Algorithms for Mining High Utility Itemsets from Transactional Databases. on IEEE Transactions on Knowledge and Data Engineering 25(8), 1772–1786.*

xxiii. *C. H. Cai, A. W. C. Fu, C. H. Cheng and W. W. Kwong 1998. Mining Association Rules with Weighted Items. In Proc. of the Int'l Database Engineering and Applications Symposium (IDEAS 1998), pp. 68-77.*

xxiv. *J. H. Chang. Mining weighted sequential patterns in a sequence database with a time-interval weight. Knowledge-Based Systems, Vol. 24, Issue 1, 2011.*