A Survey on High Utility Itemsets Mining

Shalini Zanzote Ninoria
Department of Mathematics and Computer Science Rani Durgavati Vishwavidyalaya
Jabalpur (M.P.)

S. S. Thakur
Department of Applied Mathematics
Jabalpur Engineering College
Jabalpur (M.P.)

ABSTRACT
Data Mining can be defined as a process that extracts nontrivial information contained in huge databases. Association rule mining is one of the important techniques of data mining in which relationships among the items present in the transactions are discovered. Traditional data mining techniques have focused largely on detecting the correlation between the items that are more frequent in the databases. Also termed as frequent itemset mining, these techniques were based on the grounds that itemsets which appear more frequently must be more significant to the user .High utility itemset mining is an extension to the problem of frequent pattern mining. In this paper we emphasis on an emerging area called High Utility Mining which not only considers the frequency of the itemsets but also considers the utility associated with the itemsets. The term utility refers to the usefulness of the itemset in transactions, like profit, sales or any other user preferences. In High Utility Itemset Mining the target is to identify itemsets that have utility value greater than the threshold utility value. In this paper a study of literature of the various techniques and current scenario of research in mining high utility itemset have presented also advantages and limitations of various techniques for HUIM have been presented.

General Terms
Data Mining

Keywords
Frequent Itemset, Association Rule Mining, High Utility Itemsets.

1. INTRODUCTION
We are in an age often referred to as the information age. In this information age, because we believe that information leads to power and success, and thanks to sophisticated technologies such as computers, satellites, etc., we have been collecting tremendous amounts of information. This data gives large unexploited opportunities for knowledge discovery. Here the discussion is on a very important and most popular area of research i.e. Data Mining for finding High Utility Itemsets. The goal of data mining is to predict the future or to understand the past [21]. Knowledge Discovery in Database (KDD) aims at finding meaningful and useful information in immense amounts of data.Two fundamental issues in KDD, having numerous applications in various domains, are frequent itemset mining (FIM) and association rule mining (ARM) [1],[2]. Before starting the focus should be on some basic preliminaries.

2. PRELIMINARIES
2.1 Data Mining
Data mining emerged in 1990s and has a big impact in business, industry, and science. Only the information extraction is not sufficient to help in decision making. It is essential, to build up a powerful way for analysis of such data for the extraction of interesting knowledge that could help in decision-making. Data Mining, popularly known as Knowledge Discovery in Databases (KDD), refers to the nontrivial extraction of implicit, previously unknown and potentially useful information from data in databases [24]. Data Mining is a collection of techniques for efficient automated discovery of previously unknown, valid, novel and understandable patterns in large databases. The patterns must be actionable so that they may be used in an enterprise's decision making process [15]. The goal of data mining is to extract higher-level hidden information from an abundance of raw data [6]. Data mining has been used in various data domains. Data mining can be regarded as an algorithmic process that takes data as input and yields patterns, such as classification rules, itemsets, association rules, or summaries, as output [17]. Data mining tasks can be classified into two categories, Descriptive Mining and Predictive Mining. The Descriptive Mining techniques such as Clustering, Association Rule Discovery, Sequential Pattern Discovery is used to find human-interpretable patterns that describe the data. The Predictive Mining techniques like Classification, Regression, Deviation Detection use some variables to predict unknown or future values of other variables [42].

2.2 Association Rule Mining (ARM)
Association rule mining (ARM) is a admired procedure for discovering co-occurrences, relationships, frequent patterns among items in a set of transactions or a database. The concept of Association Rules for discovering regularities between products in large databases has introduced by Rakesh Agrawal et al, 1994[1],[2]. Mining association rules can be divided into two steps: the first is generating frequent itemsets and the second is generating association rules. The major challenge in association rule mining is to recognize frequent itemsets. Finding frequent itemset is one of the significant steps in association rule mining. Since the solution of second sub-problem is straight forward, most of the researchers had focus on how to generate frequent itemsets. ARM is widely used in market-basket analysis. For example, frequent itemsets can be found out by analyzing market basket data and then association rules can be generated by predicting the purchase of other items by conditional probability [1],[2]. In example of an association rule would be "If a customer buys a computer, he is 80% likely to also purchase pen drive." So Association rule mining is the most important and well explored data mining technique, which is used by most of the organizations for decision making so that they improve the profit and enhance their performance in terms of sales and good product quality.

2.3 Frequent Itemset Mining (FIM)
Frequent itemset mining is a motivating branch of data mining. In frequent itemset mining, the base data takes the form of sets of instances (also called transactions) that each has a number of features (also called items). The original algorithm for mining frequent itemsets, which was published in 1993 by Agrawal and is still frequently used. This

algorithm functions by first scanning the database to find all frequent 1-itemsets, then proceeding to find all frequent 2-itemsets, then 3-itemsets etc. At each iteration, candidate itemsets of length n are generated by joining frequent itemsets of length n –1; the frequency of each candidate itemset is evaluated before being added to the set of frequent itemsets. The objective of Frequent Itemset Mining is to identify all the frequent itemsets in a transaction database.

Let $I = \{i_1, i_2, \ldots . i_n\}$ be a set of n distinct literals called items. An itemset is a non-empty set of items. An itemset $X = (i_1, i_2, \ldots . i_k)$with k items is referred to as k-itemset, A transaction $T = \langle TID, \big((i_1, i_2, \ldots . . i_k)\big)\rangle$ consists of a transaction identifier (TID) and a set of items $(i_1, i_2, \ldots . i_k)$ where j = 1, 2, …, k.

For example, consider the transaction database shown in Table 1. This database contains five transactions, where the letters p,q,r,s,t represent items bought by customers. For example, the first transaction T1 represents a customer that has bought the items p,r and s.

Table 1: Transaction Database

TID	Transaction
T1	{p,r,s}
T2	{q,r,t}
T3	{p,q,r,t}
T4	{q,t}
T5	{p,q,r,t}

An itemset *X* is a set of items such that $X \subseteq I$. Let the notation |*X*| denotes the set cardinality or, in other words, the number of items in an itemset *X*. An itemset *X* is said to be of length *k* or a *k*-itemset if it contains k $items$ (|*X*| = *k*). The goal of itemset mining is to discover interesting itemsets in a transaction database, i.e., interesting associations between items. In general, in itemset mining, various measures can be used to assess the interestingness of patterns. In FIM, the interestingness of a given itemset is tradi tionally defined using a measure called the *support*. The support (or *absolute support*) of an itemset *X* in a database *D* is denoted as *sup*(*X*) and defined as the number of transactions containing *X*, i.e., *sup*(*X*) =|{*T*|*X* $T \wedge T \supseteq D$}|.

For example, the support of the itemset $\{p, q\}$ is 2 because this itemset appears in two transactions (T3 and T5). Note that some authors prefer to define the support of an itemset X as a ratio. This definition called the relative sup-port is *relSup(X) = sup(X)/|D|*. For example, the relative support of the itemset {p, q} is 0.4.

An itemset X is frequent if it has a support that is no less than a given minimum support threshold minsup set by the user (i.e., sup(X) ≥ minsup). For example, if we consider the database shown in Table 1 and that the user has set minsup = 3, the task of FIM is to discover all groups of items appearing in at least three transac- tions. In this case, there are exactly nine frequent itemsets:

$\{p\}$: 3, $\{q\}$: 4, $\{r\}$: 4, $\{t\}$: 4, $\{p, r\}$: 3, $\{q, r\}$: 3,$\{q, t\}$: 4, $\{r, t\}$: 3, and $\{q, r, t\}$: 3, where the number besides each itemset indicates its support.[41]

3. HIGH UTILITY MINING (HUM)

In the previous section introduction to the basic concepts of Data Mining, Association Rule Mining, Frequent Itemset Mining .In this section a brief overview of the various algorithms, concepts and approaches of High Utility Mining that have been defined in various research publications will be presented.

The Utility Mining concept is the extension to the traditional frequent itemset mining. The traditional ARM approaches consider the utility of the items by its occurrences in the transaction set. The frequency of itemset is not sufficient to reflect the actual utility of an itemset. For example, the sales manager may not be interested in frequent itemsets that do not generate significant profit. Recently, 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 computer system may be more profitable than a telephone in terms of profit [42].In view of this utility mining emerges as an important topic in data mining field. Mining high utility itemsets from databases refers to finding the itemsets with high profits. Here, the meaning of itemset utility is interestingness, importance, or profitability of an item to users. Utility of items in a transaction database consists of two aspects: 1) the importance of distinct items, which is called external utility, and 2) the importance of items in transactions, which is called internal utility. Utility of an itemset is defined as the product of its external utility and its internal utility. An itemset is called a high utility itemset if its utility is no less than a user-specified minimum utility threshold; otherwise, it is called a low-utility itemset [55].

Utility of Itemset (U) =

*Internal Utility (i) * External Utility (e)*

Example

Let Table 2 be a database containing five transactions. Each row in Table 2 represents a transaction, in which each letter represents an item and has a purchase quantity (internal utility).Table 3 represents the unit profits associated with each itemset.

Table 2: Transaction Database

Trans_id	Transaction	Transaction Utility
T1	A(1),B(1),E(1),W(1)	5
T2	A(1),B(1),E(3)	8
T3	A(1),B(1),F(2)	8
T4	E(2),G(1)	5
T5	A(1),B(1),F(3)	11

Table 3: Unit Profits associated with items

Item Name	A	B	E	F	G	W
Unit Profit	1	1	2	3	1	1

Most of the researchers has highlighted the importance of constraint based itemset mining in which the user has the privilege to specify his or her preferences by defining constraints that capture the semantic significance of the itemset in the intended application domain [8],[9],[37].The two utility measures for any itemset are transaction utility and external utility[18]. The Transaction utility of an item in a transaction is defined according to the information stored in the transaction. The external utility of an itemset is based on the information provided by the user and is not available in the transactions [7].The concept of Mining the itemsets has the extensive well recognized literature from frequent itemsets mining to high utility itemset mining , which can be seen ahead in the paper.

Agarwal et al in [1],[2] studied the mining of association rules for finding the relationships between data items in large databases. Association rule mining techniques uses a two step process. The first step is to apply Apriori Algorithm to identify all the frequent itemsets based on the support value of the itemsets,it is a Horizontal Breadth-First Search Algorithm. Apriori uses the downward closure property of itemsets to prune off itemsets. The second step is the generation of association rules from frequent itemsets using the support – confidence model.

Following is the Apriori algorithm:

$I_1 = \{large\ 1\text{-}itemsets\};$
$For\ (k = 2; I_{k-1} \neq \phi; k++)\ do\ begin$
$C_K = apriori\text{-}gen(I_k\text{-}1); //New\ Candidates$
$For\ all\ transactions\ T \in D\ do\ begin$
$C_T = subset(C_k, T); //Candidates\ contained\ in\ T$
$for\ all\ candidates\ c \in C_T\ do$
$c.count++;$
end
end
$I_k = \{c \in C_k, c.count \geq minsup\}$
end
$Answer = U_k I_k;$

Apriori uses an iterative approach i.e. level-wise search, where k-itemsets are used to explore k+1 itemsets.The Advantage of the Apriori algorithm is perfect pruning of infrequent candidate item sets (with infrequent subsets). While on the other hand, the disadvantage of Apriori algorithm is that can require a lot of memory (since all frequent itemsets are represented) and support counting takes very long for large transactions [54].

Hence overall the Apriori alogorithm holds three major disadvantages:

1. Huge amount of time spending of processing candidates which are generated by combining the itemsets without looking at the databases and even some patterns generated that do not appear in the database.
2. Repeatedly scan the database to count the support of candidates, which is very costly.
3. It follows breadth-first search approach which is quite costly in terms of memory utilization [41].

Zaki MJ in 2000[62] has given an improvement in Apriori through Eclat Algorithm which is a Vertical Depth First Search Algorithm so that can avoid keeping many itemsets in memory. It has given the vertical database representation which can be obtained by only scanning the original horizontal database only once and vice versa.

For example the vertical representation of the database presented in Table 1 is being seen in Table 4.

Table 4: Vertical Representation of the Database of Table 1.

Item(x)	TID-Set(tid(x))
p	$\{T_1,T_3,T_5\}$
q	$\{T_2,T_3,T_4,T_5\}$
r	$\{T_1,T_2,T_3,T_5\}$
s	$\{T_1\}$
t	$\{T_2,T_3,T_4,T_5\}$

Here the search space can be explored by scanning the database only once for creating the initial TID-lists. Candidate generation and support counting are carried out directly without scanning the database.

The Eclat generates all frequent itemsets according to the depth-first search order hence it is said to be a depth-first search algorithm. Eclat is faster than Apriori but still has some drawbacks like, It generates candidates without scanning the database, hence it spend time considering itemsets that do not exist in the database, the TID-lists consumes a lot of memory too. However, there has been some work to reduce the size of TID-lists using an improved structures [61],[38].

To address the main limitation of algorithms such as Apriori and Eclat, a major progress has been presented in the form of pattern-growth algorithms. The main idea of pattern-growth algorithms is to scan a database to find itemsets, and thus avoid generating candidates that do not appear in the database. Furthermore, to reduce the cost of scanning the database, pattern-growth algorithms have introduced the concept of projected database to reduce the size of databases as an algorithm explore larger itemsets with its depth-first search[20],[53].A major advantage of pattern-growth algorithms is that they only explore the frequent itemsets in the search space thus avoiding considering many itemsets not appearing in the database, or infrequent itemsets.

In latest years, a bundle of research has been carried on further improving the performance of algorithms for FIM because it is a computationally expensive task. The improvements have been proposed in terms of various novel algorithms with additional optimization [12],[14],[43],[56]some FIM algorithms which can be run on GPU[63],on multicore processor[45],on cloud platforms[39],[44].

Although the FIM has improved in various ways but still having limitations. One of the most important limitations of FIM is that an algorithm may find a huge amount of itemsets, depending on how the minimum support threshold is set. Discovering too many patterns makes it difficult for a human to analyze the patterns found. To overcome from this many researchers have designed algorithms to extract concise representation of frequent itemsets [53],[38],[60],[49],[5].

Another limitation of traditional FIM is that it assumes that all items are equal, but in real-life applications, items are generally different from each other [33].Some items have obviously more chances of being frequent than others. This

gives the concept of rare items problem, on this issue also many researchers worked in [23],[26].Many of the researchers have worked on finding rare itemsets[50].Also the perfectly rare itemsets and minimal rare itemsets finding algorithms have also been devised[50],[27].

Likewise one more important limitation of traditional FIM algorithm is the database format for which the extension have been proposed by the researchers in terms of Weighted itemsets mining where weights are associated to each items to indicate their relative importance[52],[58],[59].

The most popular extension of weighted itemsets mining is the High Utility Itemset Mining (HUIM) in which not only the weights are considered but also the purchase quantities are considered in transactions [11],[18],[7],[19],[17],[16],[3].In HUIM weights can be considered as the unit profit of items [46],[34],[4],[47],[35],[40]. The objective of HUIM is to discover all itemsets that have a utility higher than a given threshold in a database. The journey of HUIM has embarked in 2003.

Chan et al in [11] has presented the extension to the Apriori algorithms in terms of the Novel algorithm OOA mining with the top-K utility frequent closed patterns, he also observes that the candidate set pruning strategy exploring the anti-monotone property used in apriori algorithm do not hold for utility mining.

Yao et al in [18] defines the problem of utility mining formally. The work defines the terms transaction utility and external utility of an itemset. The mathematical model of utility mining was then defined based on the two properties of utility bound and support bound. The utility bound property of any itemset provides an upper bound on the utility value of any itemset. This utility bound property can be used as a heuristic measure for pruning itemsets at early stages that are not expected to qualify as high utility itemsets.

Liu et al in [36] proposed a Two-phase algorithm for finding high utility itemsets that can discover high utility itemsets more efficiently.It works in two phases, in Phase I, a term transaction-weighted utilization is defined, and proposed the transaction-weighted utilization mining model that holds Transaction-weighted Downward Closure Property. That is, if a k-itemset is a low transaction-weighted utilization itemset, none of its supersets can be a high transaction-weighted utilization itemset. The transaction-weighted utilization mining not only effectively restricts the search space, but also covers all the high utility itemsets. Although Phase I may overestimate some itemsets due to the different definitions, only one extra database scan is needed in Phase II to filter out the overestimated itemsets.

Yao et al in [19] defines the utility mining problem as one of the cases of constraint mining. This work shows that the downward closure property used in the standard Apriori algorithm and the convertible constraint property are not directly applicable to the utility mining problem. The authors also present two pruning strategies. By exploiting these pruning strategies, the UMining and UMining_H algorithms were developed to provide efficient solutions to the utility based itemset mining problem. The pruning strategies facilitates to reduce the cost of finding high utility itemsets. The mathematical properties of the utility value of an itemset were analyzed. With these pruning strategies, a k-itemset with a utility upper bound less than minutil can be pruned immediately without accessing the database to calculate its actual utility value.

Yao et al in [17] classifies the utility-measures into three categories namely, item level, transaction level and cell level. The unified utility function was defined to represent all existing utility based measures, the mathematical properties of the utility based measures were identified. These properties can facilitate the design of efficient pruning strategies for utility based itemset mining Li et al in [16] 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.

Ahmed et al in [3] proposed three novel tree structures with the property "build once mine many", which are highly suitable for interactive mining.$IHUP_L$,IHU_{TF},&$IHUP_{TWU}$ theses three novel structures efficiently perform incremental and interactive high utility pattern mining. All three tree structures require maximum two database scans.

Shankar et al [46] have presented a novel algorithm Fast Utility Mining (FUM) which finds all high utility itemsets within the given utility constraint threshold. The authors also suggested a technique to generate different types of itemsets such as High Utility and High Frequency (HUHF), High Utility and Low Frequency (HULF), Low Utility and High Frequency (LUHF) and Low Utility and Low Frequency (LULF).

Liu Jian-ping, Wang Ying, Yang Fanding et al[34] proposed a calculation called tree based incremental affiliation manage mining calculation (Pre-Fp). It depends on a FUFP (quick redesign frequent example) mining technique. The significant objective of FUFP is the re-utilization of beforehand mined frequent things while moving onto incremental mining. The benefit of FUFP is that it decreases the quantity of hopeful set in the overhauling strategy. In FUFP, all connections are bidirectional while in FP-tree, connections are just unidirectional. The benefit of bidirectional is that it is anything but difficult to include, evacuate the youngster hub without much recreation. The FUFP structure is utilized as a contribution to the pre-extensive tree which gives positive check contrast at whatever point little information is added to unique database. It manages few changes in database if there should arise an occurrence of embedding new transaction. In this paper the calculation arranges the things into three classifications: frequent, infrequent and pre-expansive. Pre-vast itemsets has two backings limit esteem i.e. upper and lower edge. The downside of this approach is that it is tedious [34].

Ahmed CF, Tanbeer SK, Jeong BS et al[3] created HUC-Prune. They proposed a novel tree based applicant pruning strategy called HUC-tree, (high utility competitor tree) which catches the critical utility data of transaction database. HUC-Prune is totally free of high utility applicant example and it requires three database sweeps to compute the outcome for utility example. The downside of this approach is that it is extremely hard to keep up the calculation for bigger database check locales [4]. Shih-Sheng Chen et al (2011) proposed a strategy for frequent intermittent example utilizing different least backings. This is a proficient way to find frequent example since it depends on numerous base limit bolster in light of ongoing occasion. It doesn't hold download conclusion property; rather it utilizes sorted conclusion property. At that point PFP (intermittent frequent example) calculation is connected which is same as that of FP development where restrictive example base is utilized to find frequent examples. This calculation is more proficient as far as memory space, subsequently diminishing the quantity of database outputs [47].

Liu & Qu (2012) proposed HUI-MINER algorithm. In this paper Utility List is created. 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 itmesets, each utility list for an itemset keeps the information of TIDs for all of transactions containing the itemset, utility values of the item set in the transactions, and the sum of utilities of the remaining items that can be included to super itemsets of the itemset in the transactions. The distinct advantage of HUI-Miner is that it avoids the costly candidates generation and utility computation [35].

Vincent S.Tseng et al (2013) have proposed two algorithms, namely Utility Pattern Growth (UP-Growth) and UP-Growth+, for mining high utility itemsets with a set of effective strategies for pruning candidate itemsets. The information of high utility itemsets is maintained in a tree-based data structure named utility pattern tree (UP-Tree) such that candidate itemsets can be generated efficiently with only two scans of database. The performance of UP-Growth and UP-Growth+ is also compared with the state-of-the-art algorithms on many types of real and synthetic data sets. Experimental results shows that the proposed algorithms, especially UP Growth+, reduces the number of candidates effectively and also outperforms substantially in terms of runtime, especially when databases contain lots of long transactions [55].

Philippe Fournier-Viger (2014) proposed FHM algorithm [40]. It extends the Hui-Miner Algorithm. It is a Depth-first search Algorithm. It relies on utility-lists to calculate the exact utility of itemsets. This algorithm integrates a novel strategy named EUCP (Estimated Utility Co-occurrence Pruning) to reduce the number of joins operations when mining high utility itemsets using the utility list data structure. Estimated Utility Co-Occurrence Structure (EUCS) stores the transaction weighted utility (TWU) of all 2-itemsets. It built during the initial database scans. FHM is up to 6 times faster than HUI - Miner.

Next most remarkable progress in High Utility Itemset mining was proposed in terms of Efficient High Utility Itemset in 2015. Here several new ideas have introduced to more efficiently discovers high utility itemsets both in terms of execution time and memory [48]. EFIM relies on two upper-bounds named sub-tree utility and local utility to more effectively prune the search space. It also introduces a novel array-based utility counting technique called Fast Utility Counting to calculate these upper-bounds in linear time and space. Transaction merging is obviously desirable. However, a key problem is to implement it efficiently. To find identical transactions in O(n) time, sort the original database according to a new total order T on transactions. Sorting is achieved in time, and is performed only once. Projected databases generated by EFIM are often very small due to transaction merging.

The most modern recent year's attraction in the area of research in the field of FIM is Uncertainty. Uncertainty plays a role in several real-life applications because data collected is often imperfect, inaccurate, or may be collected through noisy sensors. Two main models have been proposed for uncertain FIM [51]. The first model is the expected-support model [13],[28],[29]. It considers that each item i appearing in a transaction Tq is associated to an expected support value e(i, Tq) representing the certainty that this item appeared in the transaction (a value in the [0,1] interval). The job of uncertain itemset mining in the expected support model is to find out all itemsets that are expected to be frequent. The second model is the probabilistic itemset model [13]. It considers two thresholds: the minimum support threshold minsup, as well as a second threshold called the minimum probability threshold minprob. An itemset is considered a probabilistic frequent itemset if the calculated probability that it appears in more than minsup transactions by considering possible worlds is greater than minprob.

Fuzzy itemset mining [10],[22],[11],[30] is also a well-studied extension of itemset mining. In fuzzy itemset mining, quantitative values are assigned to each item in transactions and fuzzy membership functions are defined for each attribute(item) to map these values to nominal values. For example, an attribute of a transaction could be the 'height' of a person and a corresponding fuzzy membership function could be defined to map a height to nominal values such as short, average, or tall. Fuzzy FIM algorithms discover itemsets where each item is associated with a nominal value and a membership percentage. For example, a fuzzy FIM algorithm can discover itemsets such as 'height (tall = 80%) and age (young = 60%).' Some of the most important applications of fuzzy item- set mining are in text mining, including text clustering [11].

Jerry Chun-Wei Lin et al(2015) has proposed a novel framework for mining potential high-utility itemsets (PHUIs) over uncertain databases. This is the first paper to address the issue of mining potential high-utility itemsets from uncertain databases. The upper-bound-based algorithm (PHUI-UP) and the list-based algorithm (PHUI-List) are respectively proposed to consider the mining of not only high-utility but also high probability itemsets from uncertain databases. The designed PHUI-UP algorithm is based on the proposed downward closure property to level-wisely generate-and-test candidates for mining PHUIs. The second PHUI-List algorithm is further developed to improve the performance based on the designed vertical PU-list structure for directly mining PHUIs without candidate generation [32]. In 2016 & 2017 the recent MUHUI and PHUIMUS have also proposed [31],[25].

Overall the most extensive and remarkable contributions in HUIM can be seen in the below Table 5.

4. CONCLUSION

A Utility mining is an apparent topic in data mining. The main focus in the field of Utility Mining is not only FIM but also the consideration of utility .Practically it has been found that the utility is of great interest in industry if considers with high utility itemsets. Different decision making domains such as business transactions, medical, security, fraudulent transaction, retail etc. make use of high item sets to get useful information. Survey on different high utility item set mining algorithms which were proposed is presented in this paper. This survey will be helpful for developing new efficient and optimize techniques for high utility item set mining. The open research opportunities in this field can be in the form of Novel Applications development by applying existing pattern mining algorithms in new ways, the performance can be enhanced in terms of memory and time utilization and can discover more complex and meaningful type of patterns. As the concept of High Utility Itemset Mining has a vast opportunities to be researched, the future work will incorporate soft computing methodologies for high utility itmesets mining such as the intuitionistic fuzzy logic can be explored in the field of High Utility Itemset Mining.

Table 5: Summary of remarkable contributions in HUIM

Sr. No.	Studied By	Algorithms	Year of Publication	Outcomes
1	Chan Q.,YangY., and Shen D.	OOA Algorithm	2003	Mining the top-K utility frequent closed patterns. Anti-monotone property used in apriori algorithm do not hold for utility mining.
2	Liu Y., Liao W. And Choudhary A.	Two Phase Algorithm	2005	In two phases, Phase I-transaction-weighted utilization is defined. Phase II - one extra database scan to filter out the overestimated itemsets
3	Liu M. and Qu J.	HUI-MINER Algorithm	2012	It avoids the costly candidates generation and utility computation and generate high utility itemsets.
4	Philippe Fournier Viger ,Cheng-WeiWu,Souleymane Zida and Vincent S. Tseng	FHM Algorithm	2014	It is a Depth-first search Algorithm. Reduces the number of join operations using the utility list data structure. It is up to 6 times faster than HUI-Miner.
5	Souleymane Zida, Philippe Fournier-Viger, Jerry Chun-Wei Lin,Cheng Wei Wu,and Vincent S. Tseng	EFIM Algorithm	2015	Two upper-bounds named sub-tree utility and local utility to more effectively prune the search space is used. Array-based utility counting technique is proposed.
6	Lin JCW,Gan W,Fournier-Viger P, Hong TP and Tseng VS	PHUI Algorithm	2015	Addressed the issue of mining potential high-utility itemsets from uncertain databases. Upper-bound-based algorithm (PHUI-UP) and the list-based algorithm (PHUI-List) are proposed.
7	Lin JCW,Gan W,Fournier-Viger P, Hong TP and Tseng VS	MUHUI Algorithm	2016	Based on the probability-utility-list (PU-list) structure,It directly mine PHUIs without candidate generation and can reduce the construction
8	Ju Wang, Fuxian Liu, and Chunjie Jin	PHUIMUS Algorithm	2017	Represents the itemsets with high utilities and high existential probabilities over uncertain data stream based on sliding windows.

5. REFERENCES

[1] Agarwal R., and Srikant R., "Fast algorithms for mining association rules", In the Proceedings of 20th Internaional Conf. Very large Data Bases, pp.487-499, 1994.

[2] Agrawal R., Imieliński T., Swami A.,"Mining association rules between sets of items in large databases", Proceedings of the 1993 ACM SIGMOD international conference on Management of data - SIGMOD '93. p. 207, 1993.

[3] Ahmed C.F. , Tanbeer S.K., Jeong Byeong-Soo, Lee Young-Koo, "Efficient tree structures for high utility pattern mining in incremental databases", in: IEEE Transactions on Knowledge and Data Engineering 21(12) ,2009.

[4] Ahmed CF,Tanbeer SK,Jeong BS, Lee YK, "HUC-Prune: An Efficient Candidate Pruning Technique to mine high utility patterns", Appl Intell PP: 181–198, 2011.

[5] Aliberti G, Colantonio A, Di Pietro R, Mariani R, " EXPEDITE: EXPress closED ITemset enumeration", Expert Syst Appl, 42:3933–3944, 2015.

[6] Attila Gyenesei, "Mining Weighted Association Rules for Fuzzy Quantitative Items", Lecture notes in Computer Science, Springer, Vol. 1910/2000, pages 187-219, TUCS Technical Report No.346, ISBN 952-12-659-4,ISSN 1239- 1891, May 2000.

[7] Bhattacharya S. and Dubey D., "High Utility Itemset Mining",International Journal of Emerging Technology and Advanced Engineering,Vol 2,8 August 2012.

[8] Cai C.H. , Fu A.W.C, Cheng C.H. , Kwong W.W., "Mining association rules with weighted items",in:Proceedings of IEEE International Database Engineering and Applications Symposium, Cardiff, United kingdom, pp.68-77, 1998.

[9] Chan Q.,YangY., Shen D., "Mining high utility itemsets", in:Proceedings of the 3rd IEEE International Conference on Data Mining , Melbourne , Florida, pp.19-26, 2003.

[10] Chen C.H., Li A.F., Lee Y.C. "Actionable high-coherent- utility fuzzy itemset mining",Soft Comput,18:2413–2424, 2014.

[11] Chen C.L., Tseng F.S., Liang T., "Mining fuzzy frequent itemsets for hierarchical document clustering", Inf Proc- ess Manage,46:193–211,2010.

[12] Chen J, Xiao K. BISC, "A bitmap itemset support counting approach for efficient frequent itemset mining", ACM Trans Knowl Discov Data, 4:12, 2010.

[13] Chui C.K., Kao B., Hung E., " Mining frequent itemsets from uncertain data", In: Pacific-Asia Conference on Knowledge Discovery and Data Mining, Nanjing, China, 22–25 May,47–58, 2007.

[14] Deng Z.H., Lv S.L., " PrePost+:An efficient N-lists-based algorithm for mining frequent itemsets via children parent equivalence pruning", Expert Syst Appl, 42:5424–5423, 2015.

[15] G.K.Gupta, "Introduction to Data Mining with Case Studies"Prentice-Hall of India Pvt.Ltd.New Delhi,India(2006).

[16] H.F.Li, H.Y. Huang , Y.Cheng Chen, y. Liu, S.Lee, "Fast and memory efficient mining of high utility itemsets in data streams", in :Eigth International Conference of Data Mining 2008.

[17] H.Yao, H.J Hamilton, L.Geng, "A unified framework for utility based measures for mining itemsets", in: proceedings of the ACM international conference on utility-based Data Mining Workshop (UBDM), pp. 28-37, 2007.

[18] H.Yao, H.J.Hamilton, C.J.Butz, "A foundation approach to mining itemset utilities from databases", in: Proceedings of the Third SIAM International Conference on Data Mining, Orlando, Florida , pp.482-486, 2004.

[19] H.Yao, H.J.Hamilton, "Mining itemset utilities from transaction databases" , in Data and Knowledge Engineering 59,pp.603-626 , 2006.

[20] Han J, Pei J, Ying Y, Mao R., "Mining frequent patterns without candidate generation: a frequent-pattern tree approach", Data Min Knowl Discov, 8:53–87, 2004.

[21] Han J., Pei J., Kamber M., "Data Mining: Concepts and Techniques", Amsterdam: Elsevier; 326-335, 2011.

[22] Hong T.P., Kuo C.S., Wang S.L., "A fuzzy AprioriTid mining algorithm with reduced computational time" Appl Soft Comput, x 5:1–10, 2004.

[23] Hu Y.H, Chen Y.L., "Mining association rules with multiple minimum supports: a new mining algorithm and a support tuning mechanism", Decis Support Syst, 42:1–24, 2006.

[24] Introduction to Data Mining and Knowledge Discovery, Third Edition ISBN: 1-892095-02-5, Two Crows Corporation, 10500 Falls Road, Potomac, MD 20854 (U.S.A.), 1999.

[25] Ju Wang, Fuxian Liu, and Chunjie Jin, "PHUIMUS: A Potential High Utility Itemsets Mining Algorithm Based on Stream Data with Uncertainty",Hindawi Mathematical Problems in Engineering Volume, Article ID 8576829, 13 pages 2017.

[26] Kiran R.U., Reddy P.K., " Novel techniques to reduce search space in multiple minimum supports-based frequent pattern mining algorithms", In: Proceedings of the 14th International Conference on Extending Data- base Technology, Uppsala, Sweden, 21–24,11–20, March,2011.

[27] Koh Y.S., Rountree N., "Finding Sporadic Rules Using Apriori-Inverse", In: Proceedings of the 9th Pacific-Asia Conference, PAKDD 2005, Hanoi, Vietnam, 18–20May, 97–106 , 2005.

[28] Leung CKS, MacKinnon RK., "BLIMP: a compact tree structure for uncertain frequent pattern mining", In: Proceedings of the International Conference on Data Warehousing and Knowledge Discovery, Munich, Germany, 2–4 September,115–123, 2014.

[29] Lin JCW,Gan W,Fournier-Viger P,Hong TP,Tseng VS.," Weighted frequent itemset mining over uncertain databases", Appl Intell 2015, 44:232–250,2015.

[30] Lin JCW, Tin L, Fournier-Viger P, Hong TP., "A fast algorithm for mining fuzzy frequent itemsets", J Intell Fuzzy Syst, 9:2373–2379, 2015.

[31] Lin JCW,Gan W,Fournier-Viger P, Hong TP, Tseng VS, "Efficiently mining uncertain high-utility itemsets". Springer International Publishing Switzerland 2016,WAIM 2016, Part I, LNCS 9658, pp. 17–30, 2016.

[32] Lin JCW,Gan W,Fournier-Viger P, Hong TP, Tseng VS, "Mining Potential High-Utility Itemsets over Uncertain Databases". ASE BD&SI '15 Proceedings of the ASE BigData & SocialInformatics, Article No. 25 Kaohsiung, Taiwan -October 07 - 09, 2015ACM New York, NY, USA, 2015.

[33] Liu B., Hsu W., Ma Y., "Mining association rules with multiple minimum supports ", In: Proceedings of the ACM SIGKDD International Conference on Knowl- edge Discovery and Data Mining, San Diego, CA, USA, 15–18, 337–341, August 1999.

[34] Liu Jian-Ping,Wang Ying Fan-Ding, "Incremental Mining algorithm Pre-FP in Association Rule Based on FP-tree", Networking and Distributed Computing, International Conference, pp: 199-203, 2010.

[35] Liu M. and Qu J., "Mining High Utility Itemsets withoutCandidateGeneration",CIKM'12,Maui,HI,USA, ACM, October29-November 2,2012.

[36] Liu Y., Liao W., and Choudhary A., "A Fast High Utility Itemsets Mining Algorithm," Proc. Utility-Based Data Mining Workshop, 2005.

[37] Lu S., Hu H., Li F., "Mining weighted association rules", Intelligent Data Analysis 5(3) 211-225, 2001.

[38] Lucchese C., Orlando S., Perego R., " Fast and memory efficient mining of frequent closed itemsets",IEEE Trans Knowl Data Eng,18:21–36, 2006.

[39] Moens S., Aksehirli E., Goethals B., "Frequent itemset mining for big data", In: 2013 I.E. International Con- ference on Big Data, Santa Clara, CA, USA, 6–9 October, 111–118, 2013.

[40] Philippe Fournier Viger ,Cheng-Wei Wu,Souleymane Zida,Vincent S. Tseng, "FHM: Faster High-Utility Itemset Mining using Estimated Utility Co-occurrence Pruning", Proc. 21st International Symposiumon methodologies for Intelliegnet Systems (ISMIS 2014),Springer,LNAI,pp 83-92,2014.

[41] Philippe Fournier Viger, Jerry Chun Wei Lin,Bay Vo,Tin Truong Chi,Ji Zhang,Hoai Bac Le, "A Survey of itemset mining" , WIREs Data Mining Knowl Discov 2017.

[42] Pillai J. and Vyas O.P. "Overview of itemset Utiltiy Mining and its Applications",August International Journal of Computer Applications (0975 - 8887) Volume 5 – No. 11, 2010.

[43] Pyun G., Yun U., Ryu KH., "Efficient frequent pattern mining based on linear prefix tree", Knowl-Based Syst, 55:125–139, 2014.

[44] Qui H.,Gu R.,Yuan C., Huang Y.,Yafim : "A parallel frequent itemset mining algorithm with spark" In proceedings of the 2014 I.E. International Parallel and Distributed Processing Symposium Workshops,Phoenix,AZ,USA,19-23,May 2014,1664-1671,2014.

[45] Schlegel B., Karnagel T., Kiefer T., Lehner W., "Scalable frequent itemset mining on many core processor", In: Proceedings of the 9th International Workshop Data Management on New Hardware, New York, USA,24 June, paper 3, 2013.

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

[47] Shih-Sheng Chen, Tony Cheng-Kui Huang, Zhe-Min Lin, "New and efficient knowledge discovery of partial periodic patterns with multiple minimum supports", The Journal of Systems and Software 84, pp. 1638–1651, ELSEVIER, 2011.

[48] Souleymane Zida, Philippe Fournier-Viger, Jerry Chun-Wei Lin,Cheng-Wei Wu,Vincent S. Tseng, "EFIM: A Highly Efficient Algorithm for High-Utility Itemset Mining", 30 December 2015,Mexican International Conference on Artificial Intelligence Advances in Artificial Intelligence and Soft Computing pp 530-546,2015.

[49] Szathmary L., Valtchev P., Napoli A., Godin R., Boc A, Makarenkov V. "A fast compound algorithm for mining generators, closed itemsets, and computing links between equivalence classes", Ann Math Artif Intell, 70:81–105, 2014.

[50] Szathmary L., Valtchev P., Napoli A., Godin R., "Efficient vertical mining of minimal rare itemsets", In: Proceedings of the 9th International Conference on Concept Lattices and Their Applications, Fuengirola, Spain, 11–14 October, 2012, 269–280,2012.

[51] Tong Y., Chen L., Cheng Y., Yu P.S., "Mining frequent itemsets over uncertain databases",VLDB Endowment,5:1650–1661 , 2012.

[52] Torres-VerdÃn C., Chiu K.Y., Vasudeva Murthy A.S., "WFIM: weighted frequent itemset mining with a weight range and a minimum weight." In: Proceedings of the 2005 SIAM International Conference on Data Mining, Newport Beach, CA, USA, 21–2 April ,636-640, 2005.

[53] Uno T, Kiyomi M, Arimura H. LCM ver. 2: "Efficient mining algorithms for frequent/closed/maximal itemsets", In: Proceedings of the ICDM'04 Workshop on Frequent Itemset Mining Implementations. Aachen, Germany: CEUR; 2004.

[54] Venkatesan, T.,Vinayaka, C, P. and Yogish, S., "Analysis of sampling techniques for Association Rule Mining.", In the Proceedings of the 12th International Conference on Database Theory, Vol.361, pp. 276-283, 2009.

[55] Vincent S.Tseng,Bai-En shie,Cheng-Wei Wu and Pjillip S.Yu, "Efficient Algorithms for Mining High Utility Itemset from Transactional Databases",8 August 2013,IEEE Transactions on Knowledge and Data Engineering ,Vol 25 pp 1172-1786,2013.

[56] Vo B, Le T, Coenen F, Hong TP., "Mining frequent itemsets using the N-list and subsume concepts", Int J Mach Learn Cybern, 7:253–265, 2016.

[57] Yao H.,Hamilton H. and Geng L., "A Unified Framework for Utilty-Based Measures for Mining Itemsets", In Proc. of the ACM Intel. Conf. on Utility-Based Data Mining Workshop (UBDM), pp. 28-37, 2006.

[58] Yun U. , " Efficient mining of weighted interesting patterns with a strong weight and/or support affinity",. Inform Sci. 177:3477–3499, 2007.

[59] Yun U. " On pushing weight constraints deeply into frequent itemset mining",Intell Data Anal 13:359-383,2009.

[60] Zaki M.J., Hsiao C.J., " CHARM: an efficient algorithm for closed itemset mining", In: Proceedings of the 12th SIAM International Conference on Data Mining, Ana- heim, CA, USA, 26–28 April, 457–473,2012.

[61] Zaki M.J., Gouda K., "Fast vertical mining using diffsets", In: Proceedings of the 9th ACM SIGKDD Interna- tional Conference Knowledge Discovery and Data Mining, Washington, DC, USA, 24–27 August, 2003.

[62] Zaki, M.J. "Scalable algorithms for association mining", IEEE Transactions on Knowledge and Data Engineering, 12(3), pp.372-390, 2000.

[63] Zhang F., Zhang Y., Bakos J.D., "Accelerating frequent itemset mining on graphics processing units", J Supercomput, 66:94–117,2013.