Volume 8, Issue 4
August 2019

ITEE Journal
Information Technology & Electrical Engineering

ISSN: - 2306-708X

A Hybrid Approach for Mining High Utility Rare Itemsets Over Uncertain Database

[1]Shalini Zanzote Ninoria and [2]S. S. Thakur

[1]Department of Mathematics & Computer Science, R. D. University, Jabalpur, India

[2]Department of Applied Mathematics, Jabalpur Engineering College, Jabalpur, India

E-mail: [1]shalini.ninoria@gmail.com, [2]samajh_singh@rediffmail.com

ABSTRACT

In present era, as there are many applications of uncertain data hence more emphasis has paying focus on mining itemsets over uncertain databases. Data mining is a technique which is useful in the extraction of interesting relationships between data in huge databases. Association rule mining is one of the most vital techniques of data mining in which association among the items present in the transactions are discovered. High-utility itemset mining (HUIM) has come up as a most significant research topic in data mining. High utility rare itemsets in a database can be used by retail stores to adapt their marketing strategies in order to increase their profits. Even though the itemsets mined are infrequent, since they generate a high profit for the store, marketing strategies can be used to increase the sales of these items. In this paper, a novel approach named A Hybrid Approach for Mining High Utility Rare Itemsts over Uncertain Database ((Improved High Utility Rare Itemset over Uncertain Database (IHURIU) algorithm) is proposed to generate all high utility rare itemsets while keeping the algorithm time-efficient as well as memory-efficient. This proficient approach improved the concept of apriori inverse over uncertain database and it will give blend of Improved Apriori[1],apriori-inverse[2] and UHUI-apriori [3] algorithm approaches in the form of hybrid proposed approach. This paper will also give the new version or extension of the algorithm HURI proposed by Jyothi et al as it will give the improvement in the basic apriori algorithm for item generation as well as implementation of HURI over uncertain database. The implementation of an algorithm for the analysis is done on JDK 6.1 and referred the sample dataset presented by Lan Y.et al,2015[3] for uncertain database.

Keywords: *Data Mining,Association Rule Mining,High Utility Itemset Mining,Rare Itemset Mining,Uncertain Database*

1. INTRODUCTION

In the current decade the amount of database has increased tremendously. This rapid development has led major interest to develop various tools which can be used to handle the data as well as extract the desired information moreover knowledge. The term data mining or knowledge discovery in database has been implemented for a field of research dealing with the automatic discovery of hidden information or knowledge within the databases. The hidden information within databases, mainly the interesting association relationships among sets of objects that lead to association rules may reveal useful patterns for decision support, financial predict, marketing policies, even medical diagnosis and many other applications [4].Data mining is the method of determining interesting, meaningful, and understandable patterns hidden in large data sets [5]. Now a day's organization collects numerous data and this data is stored in form of transaction [6]. In the area of business, corporate and customer data are becoming recognized as a tactical asset. The ability to extract valuable knowledge unseen in these data and to act on that knowledge is becoming increasingly important in today's competitive globe. The entire process of application of a computer based methodology, including new techniques, to find out knowledge from data is called data mining [1].Knowledge Discovery in Database (KDD) aims at finding meaningful and useful information in substantial amounts of data [7]. Data mining emerged in 1990s and has a big impact in business, industry, and science. Data mining has been used for many years by many fields such as businesses, scientists and governments, etc. Han and Kamber [8] presented, data mining functionalities include data characterization, data discrimination, association analysis, classification, clustering, outlier analysis, and data evolution analysis. Data mining is the procedure of applying these methods to data with the intention of discovering hidden patterns [9].

Two primary issues in KDD, having many applications in a variety of domains, are frequent itemset mining (FIM) and association rule mining (ARM) [10],[5],[7]. Agrawal and Srikant in 1993. Discovering associations between items is helpful to understand customer behavior. For example, a retail store manager can use this facts to take strategic marketing decisions such as co-promoting products or putting them closer on the shelves[11].The first algorithm has been proposed named Apriori for finding the Frequent Itemsets occurred in the database [12]. Apriori algorithm is useful for searching the association rules among items in market-basket data [13],[14]. Association rules use two main constraints, i.e. minimum support and minimum confidence. The task of frequent itemset mining has various applications, but it can also be viewed with limitations in terms of the assumptions that it makes. These limitations have generated the need of extension to FIM.Some work has done on sampling techniques also which can be one of the solution for the limitation of FIM[15]. One of the remarkable limitations of traditional FIM is that it assumes that all items are equal. But in real-life applications, items are often different from each other [12].

In actual situations some items naturally have more chance of being frequent than others. This show the way to generate the rare item problem [12], which means that some items are much less likely to appear in frequent itemsets than others. Only frequency consideration in traditional techniques is not sufficient. Mining rare patterns from Databases has always been overlooked and giving more emphasis on frequent one. The unknown and unusual patterns are proficient in discovering hidden useful information from databases in various domains of applications. The first attempt towards rare association rule mining was made by Liu et al. [12] in their algorithm called MS-Apriori that employs an Apriori like strategy to incorporate some rare items during itemset generation. The authors argued that a single support threshold cannot be used for extracting the rare patterns effectively and ended up proposing a "multiple support framework" for the same. The framework assigns each item their individual support values instead of relying on a single one. The algorithm is efficient in finding rare patterns but it employed an additional parameter β that adds to the computational complexity of the algorithm. Kiran et al. [16] in their algorithm IMS-Apriori, improved the initial MS-Apriori algorithm by incorporating another parameter of support difference. Even though it succeeded in generating more number of rare items it increases the burden of assigning two extra parameters:β and support difference. Lee at al. [17] extended the concept of multiple minimum supports using a model called maximum constraints model. The minimum support considered in this case is the maximum value among the minimum support values assigned to each item. The algorithm is faster due to granular bit string computation but fails to generate the complete set of rare items. Some algorithms extend the Apriori algorithm and use only a single minimum support threshold to find the rare itemsets. The most significant effort in this regard was made in [18]. The algorithm called ARIMA is capable of finding the complete set of rare items but spends a lot of time looking for the rare and frequent itemsets. ARIMA is further extended by Hoque et al. [19] in their algorithm FRIMA that generates both the frequent and rare itemsets. The algorithm maintains the rare, frequent and zero itemsets in three different candidate lists and later on merges the lists containing frequent and rare itemsets into a single list removing the zero itemsets. The algorithm handled the generation of whole set of rare items in less execution time than ARIMA but it increases memory utilizations as it also retain zero itemsets along with frequent and rare itemsets. Adda et al. [20] employed a strategy different from the previous approaches. Their algorithm AfRIM, carried out the level-wise search in top-down manner contrasting the traditional bottom up search approach. The algorithm initially generates the largest candidate itemset combining all rare items and then proceeds to generate the smaller candidate itemsets. Similar to FRIMA, it also suffers from the drawback of generating zero support itemsets. Pillai and Vyas [21],[22] identified the need for high-utility rare itemsets and proceed to generate the same in their algorithm HURI. To measure the significance of rare itemsets, HURI consider the utility values of the itemsets along with their frequencies. The itemsets satisfying the predefined minimum utility value are considered to be rare, discarding other itemsets. Despite generating the user interested rare itemsets, the algorithm proves to be tedious due to the pre-assignment of utility values to each individual item. Instead of generating the complete set of rare itemsets, Haglin and Manning [23] developed the MINIT algorithm to generate only a subset of the rare itemsets called minimal infrequent itemsets. The algorithm assigns individual ranks to the items based on their support values and further considers only the higher rank items for itemset generation. The algorithm spends lesser execution time due to the generation of only minimal infrequent itemsets but still misses out some significant rare itemsets. Rarity algorithm proposed by Troiano et al. [24] considers the longest transaction in the database for rare itemset generation and performs a levelwise top-down search like AfRIM. The algorithm maintains a Candidate list for retaining the rare itemsets and a Veto list for retaining the frequent itemsets. The rare itemsets generated are finally stored in another list. Despite generating the complete set of rare items, the algorithm undergoes memory overhead. In addition to the usage of a single minimum support threshold or multiple minimum support thresholds, some rare pattern mining techniques employ dynamic thresholds or more than one threshold. RSAA algorithm proposed by Yun et al. [25] employed two thresholds, one for generating the rare itemsets and another for generating the frequent itemsets. The advantage of this algorithm is that it is independent of the parameter β employed by MS-Apriori but fails to outperform in terms of execution time. Tao et al. [14] in their algorithm WARM employed weighted support instead of minimum support threshold. Based on the significance of items, a weight is assigned to each item and only those items are considered further that satisfy the predefined weight threshold. However, assigning proper weights to each item adds to complexity of the algorithm. Wang et al. [26] in their algorithm Adaptive Apriori, pushed some support constraints on the itemsets. The lowest minimum support is considered, in case two or more constraints are applied on the itemsets. Maintaining the ordering of items even at run time becomes a tedious affair for the algorithm. DCS Apriori developed by Selvi and Tamilarasi [27] uses two support thresholds: Dynamic and Collective. Using the Dynamic support count, significant rare items are retained and the items that do not satisfy the Collective support are removed. Even though the algorithm is free of the user-defined threshold still it fails to generate the whole set of rare items. Sadhasivam and Tamilarasi [28] proposed Automated Apriori Rare that automatically assigns the support thresholds to items to derive the frequent as well as rare itemsets. The algorithm employs the strategy of MS-Apriori to extract the rare itemsets and Apriori to derive the frequent itemsets. The algorithm has the advantage of operating in parallel but misses out some significant rare itemsets [29]. Practically, the FIM is limited by the intendment of the discovered itemsets and quantity is not considered (all items are viewed as having the same importance), high-utility itemsets introduced utility [30], a measure which is a weight/profit associated with each itemset. In [10],[31],[32]. Examples explain that all FIMs are not profitable and all high profits need not be frequent. Due to this fact, many itemsets emerged like [33] rare itemsets, [34] infrequent itemsets, and [5] closed itemsets, [35] lattice-based mining [36]. In real world, for decision making the user wants to know the total profit earned by an item or itemset. For this it needs to take into account the quantity of the item purchased. The profit of an item considers the profit of single item and the number of item purchased. To address these, utility mining has been launched in which utility considers both the profit and the number of items purchased. The utility of an itemset can be evaluated as the product of profit of an items and number of items purchased. Utility mining considers both the importance of an item in the database (i.e.) profit or external utility and the importance of an item in the transaction (i.e.) quantity or internal utility of an item. An item is said to be high utility item if the sum of the utility of the item in the database is greater than the user specified minimum threshold, otherwise it is low utility item. High utility pattern mining is a major research area in recent years and many researches are carried in different areas of high utility mining (HUI) which includes sequence high utility mining, lossless representation of HUI, incremental HUI etc [37]. Weighted itemset mining is an extension of frequent itemset mining where weights are coupled to each item to indicate their relative importance[38],[39],[40] with the objective to find itemsets that have a minimum weight. The infrequent weighted itemsets is a popular variation to this problem [34]. The major extension of weighted itemset mining is arised in terms of High-utility itemset mining (HUIM) where not only weights are considered but also purchase quantities or utility in transactions [41],[42],[43],[44,[45],[46],[47]. In traditional FIM, either an itemset appears in a transaction or not. In HUIM, the utility or quantity is also indicated in transactions. For example, a transaction could specify that a customer has bought two desktops and one pen drive. In HUIM, weights are used to indicate how much profit is generated by each unit sold of a product. The objective of HUIM is to find all itemsets that have a utility higher than a given threshold in a database (i.e. itemsets generating a high profit). Plenty of work has been done on HUIM so far like Two Phase algorithm[45] where an upper-bound, called the TWU is used to reduce the search space , tighter upper-bounds on the utility is introduced so that the algorithm be able to prune a larger part of the search space and improve the performance of HUIM algorithms [41],[48][49],[46],[47]current fastest HUIM algorithm is EFIM[19], shelf-time periods of items[50], discover the top-k most profitable itemsets [51],[52] etc.The rest of the paper is organized as follows: in Section 2 Comprehensively discusses Preliminaries. In Section 3 A Hybrid Approach for Mining High Utility Rare Itemsets over Uncertain Databases (IHURIU-Algorithm) is proposed. In Section 4 Experimental Assessment on the uncertain dataset example given by Lan Y et al, (2015) are reported. Finally, we conclude the paper in Section 5.

2. PRELIMINARIES

In this section the basic definitions have discussed which are important to understand the problem statement. Let I = {i_1, i_2, ..., i_n}signifies a set of distinct items. A unit value u(i_p) is associated with each items i_p. A itemset is a non-empty subset X. X = {x_1, x_2, ...x_n} denotes the set of itemset. X is a l-item set if it has l items. In given example D is a uncertain transaction database in which Given an uncertain transaction database D, each transaction is denoted as < tid, Y >, where tid is the transaction identifier, and Y = {$y_1(p_1)$, $y_2(p_2)$,..., $y_m(p_m)$}. Y also contains m units. Every unit has an item yi.Also every unit has a probability pi, which denotes the possibility of item yi appearing in the tid tuple. The minimum utility threshold is also defined as µ [3].

Let us take an example of the uncertain database in Table 1.The utility table is available in Table 2 shown below [3]. µ = 25% i.e. minimum utility threshold.

TID	Transactions
T1	A (0.2) C (0.3) E (0.2)
T2	B (0.2) D (0.3)
T3	A (0.1) B (0.2) C (0.1) E (0.3)
T4	C (0.2)
T5	B (0.3) D (0.2) E (0.1)
T6	A (0.2) C (0.2) D (0.5)
T7	A (0.1) B (0.1) D (0.4) E (0.1)
T8	B (0.4) E (0.1)
T9	A (0.3) C (0.3) D (0.2)
T10	B (0.2) C (0.3) E (0.1)

Table 1: An Uncertain Database

TID	A	B	C	D	E
Utility	4	1	12	6	15

Table 2: Utility Table

Definition 1:

Utility of an Item: The Utility of an item is the quantity of each item from the itemset multiplied by their unit profit [3]. i.e utility of an item i_j in T_d can be defined as :

$$U(i_j, T_d) = p(i_j, T_d) \times u \qquad (1)$$

For example, the utility f an items (C) in T_3 is,

$$U(C, T_3) = p(C, T_3) \times u(C) = 0.1 \times 12 = 1.2$$

Definition 2 :

Utility of an itemset X in transaction T_d: The Utility of an itemset X in transaction T_d is denoted by U(X,T_d) [3],which can be defined as :

$$U(X, T\mathrm{d}) = \sum_{i_j \in X \wedge X \subseteq T_d} U(i_j, T_d) \qquad (2)$$

For example, the utility of (BC) in T_3 is calculated as,

$$\begin{aligned} U(BC, T_3) &= U(B, T_3) + U(C, T_3) \\ &= p(B, T_3) \times U(B) + p(C, T_3) \times U(C) \\ &= (0.2 \times 1) + (0.1 \times 12) = 1.4 \end{aligned}$$

Definition 3:

Utility of an itemset X in Uncertain database D: The Utility of an itemset X in Uncertain database D is denoted as U(X) [3], it can be defined as:

$$U(X) = \sum_{X \subseteq T_d \wedge T_d \in \mathrm{D}} U(X, T_d) \qquad (3)$$

For example,

U(C)=U(C,T$_1$)+U(C,T$_3$)+U(C,T$_4$)+U(C,T$_6$)+
U(C,T$_9$)+U(C,T$_{10}$)
= 3.6+1.2+2.4+2.4+3.6+3.6)=16.8

U (BC) = U (BC, T$_3$) + U (BC, T$_{10}$) = (1.4+3.8) = 5.2

Definition 4:

Transaction utility of transaction Tq : The transaction utility of transaction Tq is denoted as tu(Tq) [3], which can be defined as:

$$tu(T_q) = \sum_{j=1}^{m} U(i_{j,} T_d) \qquad (4)$$

in which m is the number of items in T$_d$.

For example,

TU(T$_3$)=U(A,T$_3$)+U(B,T$_3$)+U(C,T$_3$)+U(E,T$_3$)
= 0.4+0.2+1.2+4.5 = 6.3

Definition 5:

Total utility in D: The total utility in D is the sum of all transaction utilities in D and is denoted as TU [3], which can be defined as:

$$TU = \sum_{T_d \in D} TU(T_d) \qquad (5)$$

For example, the transaction utilities for T$_1$ to T$_{10}$ are respectively calculated as TU(T$_1$) = 7.4, TU(T$_2$) = 1.3, TU(T$_3$) = 6.3, TU(T$_4$) = 2.4, TU(T$_5$) = 3, TU(T$_6$) = 6.2, TU(T$_7$) = 4.4, TU(T$_8$) = 1.9, TU(T$_9$) = 6.0, TU(T$_{10}$) = 5.3.

The total utility in D is the sum of all transaction utilities in D, which is calculated as:

TU = (7.4+1.3+6.3+2.4+3+6.2+4.4+1.9+6+5.3)=44.2

Definition 6:

High Utility Itemset (HUI) : An itemset X is defined as a high utility itemset (HUI) if its utility value U(X) is not less than the minimum utility count [3] as:

$$\sum_{X \subseteq T_d \wedge T_d \in \mathrm{D}} U(X, T_d) = U(X) \geq \mu \times TU \qquad (6)$$

For example, suppose that the minimum utility threshold μ is set at 25%. An itemset(C) is considered as a HUI in D since U(C) = 16.8, which is larger than the minimum utility count = (0.25 × 44.2) = 11.05.

Definition 7:

Transaction-weighted utility (TWU): The transaction-weighted utility (TWU) of an itemset X is the sum of all transaction utilities TU(T$_d$) containing an itemset X [3], which is defined as:

$$TWU(X) = \sum_{X \subseteq T_d \wedge T_d \in \mathrm{D}} TU(T_d) \qquad (7)$$

Definition 8:

High transaction-weighted utilization itemset (HTWUI): An itemset X is considered as a high transaction-weighted utilization itemset (HTWUI) [3] if its

$$TWU(X) \geq TU \times \mu$$

Theorem 1 (Downward Closure Property of HTWUI): Let X^k and X^{k-1} be the HTWUI from uncertain databases, and $X^{k-1} \subseteq X^k$. The TWU (X^{k-1}) ≥ TWU (X^k) [3].

Proof: Let X^{k-1} be a (k-1)-itemset and its superset k itemset is denoted as X^k. Since $X^{k-1} \subseteq X^k$, thus,

$$\begin{aligned} TWU(X^k) &= \sum_{X^k \subseteq T_d \wedge T_d \in D} TU(T_d) \\ &\leq \sum_{X^{k-1} \subseteq T_d \wedge T_d \in D} TU(T_d) \\ &= TWU(X^{k-1}) \end{aligned} \qquad (8)$$

Corollary 1: If an itemset X^k is a HTWUI, every subset X^{k-1} of X^k is a HTWUI [3].

Corollary 2: If an itemset X^k is not a HTWUI, no superset X^{k+1} of X^k is a HTWUI [3].

Theorem 2 (HUIs ⊆ HTWUIs): The transaction-weighted downward closure (TWDC) property ensures that HUIs ⊆ HTWUIs, which indicates that if an itemset is not a HTWUI, then none of its supersets will be HUIs [3].

Proof: ∀X ⊆ D, X is an itemset; thus,

$$\begin{aligned} U(X) &= \sum_{X \subseteq T_d \wedge T_d \in D} U(X, T_d) \\ &\leq \sum_{X \subseteq T_d \wedge T_d \in} TU(T_d) \\ &= TWU(X) \end{aligned} \qquad (9)$$

Considering the above definitions as base ,the statement of problem of mining HUIs over uncertain databases can be stated as, given an uncertain database D with total utility TU, μ the minimum utility threshold the problem of MHUI over uncertain databases is: To mine HUIs whose utilities are not less than (μ × TU) [3].

Let us consider transaction database in Table 3 with total utility which have been calculated with the help of Table 4 profit table shown below .

TID	Transaction	TU
T1	(C : 5) (D : 20)	70
T2	(C : 1) (F : 40)	42
T3	(A : 1) (B : 1) (C : 2) (G : 10)	20
T4	(A : 1) (B : 1) (C : 2)	10
T5	(A : 5) (C : 10)	45
T6	(B : 1) (C : 1) (E : 1)	5
T7	(B : 1) (C : 1) (E : 1) (G : 10)	15
T8	(B : 1) (C : 1) (E : 1) (H : 1)	6
T9	(C : 10) (E : 10)	40
T10	(A : 1) (B : 1) (C : 1)	8

Table 3 :Transaction Table with utility

Item	A	B	C	D	E	F	G	H
Profit	5	1	2	3	2	1	1	1

Table 4 : Profit Table

Definition 9:

An itemset X is called a rare itemset, if sup(X) < max_sup_threshold [22].

If max_sup_threshold=2, Table 5 shows the Rare Itemsets of above example database.

Itemsets	List of Rare Itemsets
1-itemset	{D},{F},{H}
2-itemset	{AG},{BH},{CD},{CF},{CH},{EG},{EH}

3-itemset	{ABG},{ACG},{BCH},{BEH},{BEG},{CEG}
	{CEH}

4-itemset	{ABCG},{BCEG},{BCEH}

Table 5: Rare Itemset Table

Based on the above definitions, the problem statement of mining IHURIU over uncertain databases had formulated as, for a given uncertain database D with total utility is TU, the minimum utility threshold which is set as µ. The problem of mining HURI over uncertain databases is to mine Rare Itemset by considering those itemsets which have support value less than the max support threshold using improved Apriori Inverse Concept and the itemsets which are rare but having utility value greater than the min utility threshold value.

3. A HYBRID APPROACH FOR MINING HIGH UTILITY RARE ITEMSETS OVER UNCERTAIN DATABASE

This section will address the proposed innovative approach for mining high utility Rare itemsets over uncertain databases using apriori-inverse [2],Improved Apriori [1] and Improved UHUI-Apriori [53] using idea of improved apriori algorithm .The major limitations in apriori algorithm has been focussed and updated the process of item generation method. Extensive experiments have been performed to test the performance of these approaches over our sample example dataset. Mohammad et al in [1] have proposed an approach, here the Apriori algorithm enhanced to reduce the time consuming for candidates itemset generation. Firstly it scans all transactions to generate L_1 which contains the items, their support count and Transaction ID where the items are found. Then L_1 is used later as an assistant to produce L_2, L_3 ... L_k. At the time when C_2 get generated, a self-join L_1 *L_1 will be generated to construct 2-itemset C(x, y), where x and y are the items of C_2. Before scanning all transaction records to count the support count of each candidate, use L_1 to get the transaction IDs of the minimum support count between x and y, and thus scan for C_2 only in these specific transactions. The same thing for C_3, construct 3-itemset C(x, y, z), where x, y and z are the items of C_3 and use L1 to get the transaction IDs,with their minimum support count between x, y and z, then scan for C_3 only in these specific transactions and repeat these steps until no new frequent itemsets are identified. The process can be clearer through Figure1 given below:

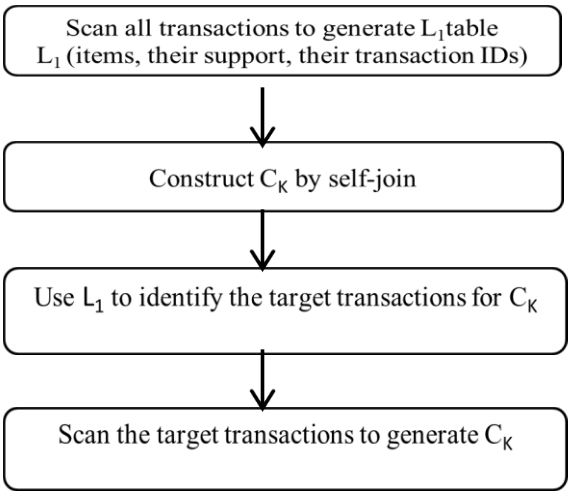

Scan all transactions to generate L_1table
L_1 (items, their support, their transaction IDs)
Construct C_K by self-join
Use L_1 to identify the target transactions for C_K
Scan the target transactions to generate C_K

Figure 1: Steps for C_k Generation [1]

On the basis of the approach proposed in [22], shalini et al has proposed a novel efficient algorithm HURIU which reduced the extraction time of high utility rare itemsets over uncertain database drastically [53].The explanation of the working process of complete phases have also been explained with the help of experiment on a sample dataset. The proposed approach is a two phase method.

Let us take an example of a sample uncertain dataset shown in Table 1 and the profit value associated with each item in Table 2.Let the minimum utility threshold µ = 25 % and the maximum frequency support is 2. Improved Apriori concept with Apriori-inverse concept is used in HURIU (High Utility Rare Itemset over Uncertain Dataset) algorithm presented and generates HURI over Uncertain Dataset. In first phase, rare itemsets are extracted by considering those itemsets which have support value less than the maximum support threshold using apriori-inverse [2]concept.In second

Volume 8, Issue 4
August 2019

ITEE Journal
Information Technology & Electrical Engineering

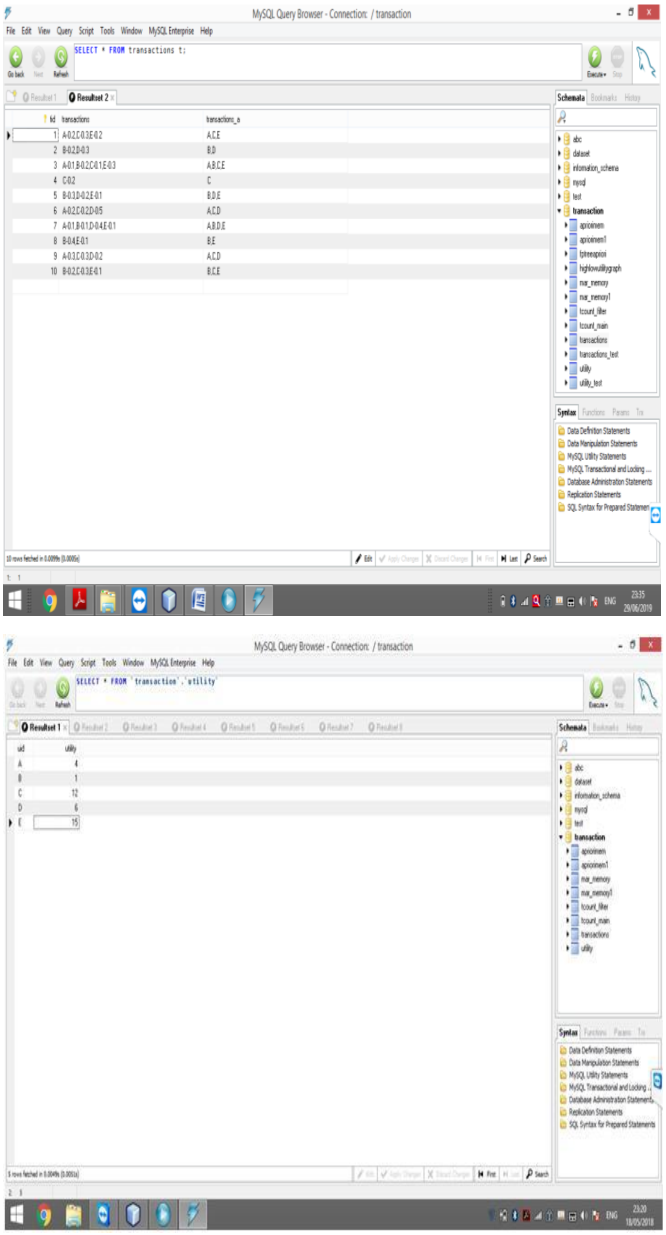
ISSN: - 2306-708X

phase the UHUI-apriori algorithm is extended using as an input the utility threshold value according to user's interest, rare itemsets having utility value greater than the minimum utility threshold are generated using the concept of UHUI-apriori algorithm. The proposed efficient algorithm HURIU Algorithm to mine HURI over uncertain databases is given below.

```
Algorithm : IHURIU algorithm
//D is uncertain database; utable is table of utilities for itemsets, min. threshold given by µ
Input: D, an uncerain database, utable, utility table, µ minimum utility threshold
Output: the setof high utility rare itemsets(HURIs)
//Loop for calculating Transaction Weighted Utility
// Scanning database and probabilities
    For each $T_d$ Present in D $\wedge$ $i_j$ in $T_d$ Do
    calculate $TWU(i_j)$
    End for loop
//Condition Check for High Utility and generate set of HTWUI$^k$
      For each $T_d$ in D $\wedge$ $i_j$ in $T_d$ do check
         If $TWU(i_j) \geq TU \times \mu$ Then
            Set $HTWUI^1 \leftarrow i_j$
         End If
      End For loop
      set $k \leftarrow 2$
      set X as (k) – itemset
//start candidate generation and check condition for HURI.
      while $HTWUI^{k-1} \neq null$ Do
      $C_k \leftarrow$ Improved – apriori
                        – inverse$(HTWUI^{k-1})$
         For each k – itemset X in $C_k$Do
            Scan D to Find TWU(X)
               If $TWU(X) \geq TU \times \mu$ Then
                  $HTWURI^k \leftarrow X$
               End If
         End For
      Do $k \leftarrow k+1$
      End While Loop
      $HTWURIs \leftarrow HTWUI^k$
//Rescanning of D
   For each k – itemset X in HTWURIs Do
   Scan D to Find u(X)
   If $u(X) \geq TU \times \mu$ Then
   $HURI^k \leftarrow X$
   End If
   End For
   $HURI_s \leftarrow HURI^k$
```

4. EXPERIMENTAL ASSESSMENT

This section illustrates the experimental analysis and results of proposed IHURIU algorithm. The experiment is implemented in Java (jdk 1.6), NetBeans IDE 7.0 & MySQL.This implementations have been done with sample example dataset with varying thresholds. The screen of sample dataset for implementation using MySQL can be seen in Figure 2.



MySQL Query Browser - Connection: / transaction

```
SELECT * FROM transactions t;
```

tid	transactions	transactions_a
1	A-0.2,C-0.3,E-0.2	A,C,E
2	B-0.2,D-0.3	B,D
3	A-0.1,B-0.2,C-0.1,E-0.3	A,B,C,E
4	C-0.2	C
5	B-0.3,D-0.2,E-0.1	B,D,E
6	A-0.2,C-0.2,D-0.5	A,C,D
7	A-0.1,B-0.1,D-0.4,E-0.1	A,B,D,E
8	B-0.4,E-0.1	B,E
9	A-0.3,C-0.3,D-0.2	A,C,D
10	B-0.2,C-0.3,E-0.1	B,C,E

10 rows fetched in 0.0099s (0.0005s)

MySQL Query Browser - Connection: / transaction

```
SELECT * FROM `transaction`.`utility`
```

uid	utility
A	4
B	1
C	12
D	6
E	15

Figure 2: Sample Dataset

For the experimental assessment consider the sample uncertain database is given in Table 1 and Table 2 where each transaction is assigned a unique probability value in the range of 0.1 to 0.5.The experiment have performed for different higher and lower threshold values viz 25%, 35%, 40%, 50%, and 60% respectively. This implementation has done for the performance analysis of the proposed algorithm IHURIU,HURIU[53],UHUI-apriori[3].The performance has been evaluated on the basis of three parameters i) Run Time

ii) Memory Utilization iii) Total Number of Itemsets Generated.

4.1 Run Time

In computer science, the analysis of algorithms is for the determination of the amount of resources such as time which is necessary to execute them. Here, comparison analysis between the proposed algorithms IHURIU, HURIU and UHUI-apriori is done with various threshold values on the sample dataset. Results can be seen in the graph shown in the below Figure 3.

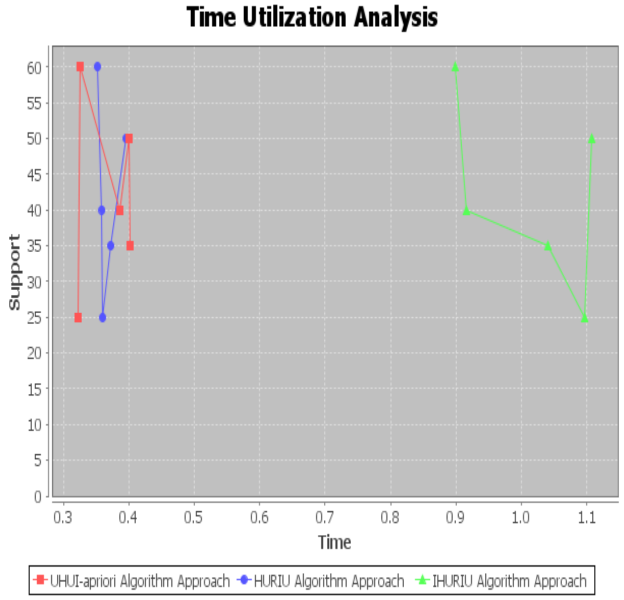


Figure 3: Time Utilization Analysis

4.2 Memory Utilization

Memory complexity can be given in terms of the size used by the algorithm for the work. The memory utilization is analyzed for both higher and lower threshold. We can observe easily that our proposed algorithm act differently for various threshold values. Results are shown in below Figure 4.

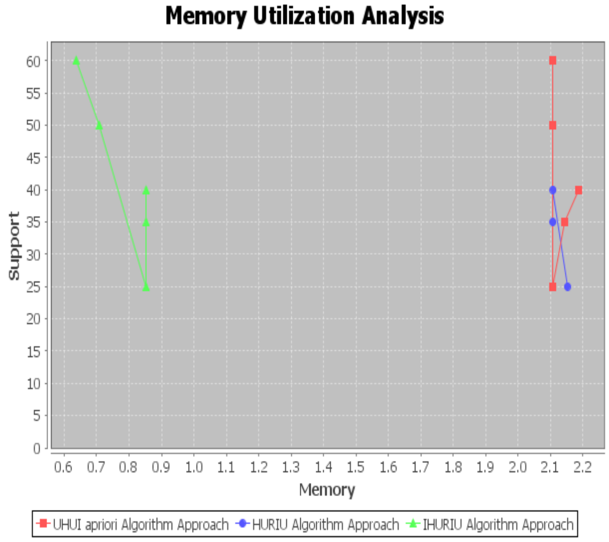


Figure 4: Memory Utilization Analyses

4.3 Total No. of Items Generation

It can be clearly observed in the below graph that the Number of high utility rare items (HURI) are more for the lower threshold and reduced when the threshold is high. The comparison can also be seen between high utility items, high utility rare items using HURIU approach and high utility rare items using proposed approach. The analysis can be seen in the below Figure 5.

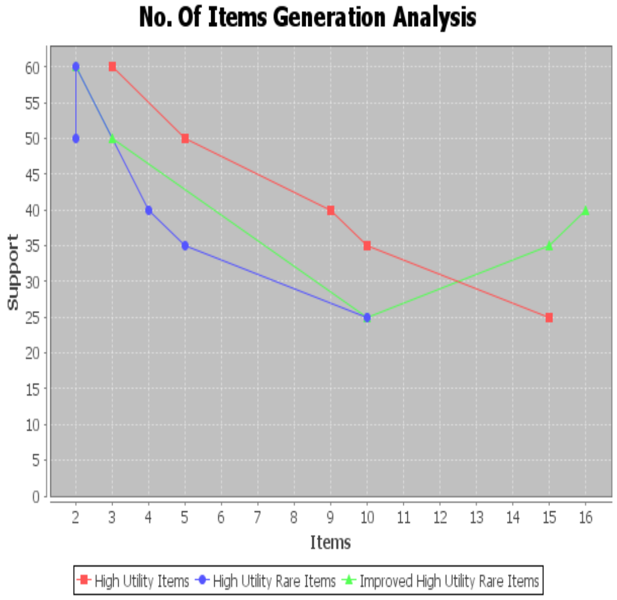


Figure 5: Items Generation Analysis

5 CONCLUSION

In this paper the working analysis has been done on the approaches to generate variety of high utility items. The high utility items(HUI) generated by UHUI-apriori algorithm ,high utility itemsets which are rare in nature(HURI) using HURIU algorithm and high utility rare itemsets(HURI) using proposed IHURIU algorithm over uncertain database. It can be concluded that proposed algorithm IHURIU can be suggested for better utilization of memory as it consumes less memory for the generation of High Utility Rare Itemsets.

Sr. No.	Algorithms	Support (Threshold %)	Run Time Utilization (in seconds)	Memory Utilizations (in bytes)
1	UHUI-apriori Algorithm	25%	0.322	2.107421875
		35%	0.402	2.143608093
		40%	0.386	2.187011719
		50%	0.4	2.10710144
		60%	0.326	2.106452942
2	Improved-UHUI-apriori Algorithm	25%	0.36	2.151176453
		35%	0.372	2.106758118
		40%	0.359	2.10773468
		50%	0.396	2.107284546

		60%	0.352	2.106361389
3	IHURIU Algortihm	25%	1.097	0.853797913
		35%	1.041	0.853805542
		40%	0.916	0.85382843
		50%	1.108	0.708869934
		60%	0.899	0.63634491

Table 6 : Conclusion Table of comparison of Time utilized and Memory utilized

It consumes more time hence not recommended for better time utilization. It is also concluded that as the threshold value is increases the generation of HURI is decreases and with lower threshold values it increases the number of HURI generated.

Sr.No.	Algorithms	Support (Threshold %)	No. Of Itemsets Generated
1	HURI itemsets using proposed approach	25%	10
		35%	15
		40%	16
		50%	3
		60%	2
2	HUI itemsets using UHUI-apriori approach	25%	15
		35%	10
		40%	9
		50%	5
		60%	3
3	HURI itemsets using HURIU approach	25%	10
		35%	5
		40%	4
		50%	2
		60%	2

Table 7 : Conclusion Table of comparison of different types of itemsets generated using different approaches

Hence as per the above conclusion tables we can conclude that in the proposed hybrid approach using IHURIU algorithms the memory utilization is reduced drastically for both the higher and lower threshold values. It can also be concluded as per Table 6 & Table 7 that if the threshold values are low the number of high utility rare itemsets will be more and as the threshold values are moving higher the generation of high utility rare itemsets will be less. Hence we can recommend the Hybrid approach for the extraction of High Utility Rare Itemset (HURI) over Uncertain Database for better memory utilization.

REFERENCES

[1] M. Al-Maolegi and B.Arkok, “An improved apriori algorithm for association rules”, arXiv preprint arXiv: 1403.3948, 2014.

[2] Y.S. Koh and N. Rountree , “Finding sporadic rules using apriori-inverse”. In Pacific-Asia Conference on Knowledge Discovery and Data Mining Springer, Berlin, Heidelberg., May 2005.

[3] Y.Lan, Y.Wang, Y.Wang, S. Yi and D.Yu,“Mining high utility itemsets over uncertain databases”. International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery IEEE, September 2015.

[4] S.Pramod, and O.P. Vyas, “Survey on frequent item set mining algorithms”. International journal of computer applications, 1(15),2010.

[5] R.Agrawal and R. Srikant,“Fast algorithms for mining association rules in large databases”. In: Proceedings of the 20th International Conference on Very Large Data Bases (VLDB ‘94), 1994.

[6] A.M.Patel and D.Bhalodiya, “A Survey On Frequent Itemset Mining Techniques Using GPU”, Internatonal Journal Of Innovative Research In Technology,Vol.1, Issue5, 2014.

[7] S.Z. Ninoria and S.S.Thakur, “Review On Rare Itemset Mining”. International Journal of Computer Sciences and Engineering, NCRTI, Feb 2019.

[8] J. Han, J.Pei and M.Kamber, “Data mining: concepts and techniques", Elsevier, 2011.

[9] M. Kantardzic, “Data mining: concepts, models, methods, and algorithms”,John Wiley & Sons, 2011.

[10] R.Agrawal, T.Imielinski and A. Swami, “Mining Association rules between sets of items in large database”, In: ACM SIGMOID International Conference on Management of Data, 1993.

[11] P.Fournier Viger, J.C.W.Lin, B. Vo, T.T. Chi, J.Zhang, and H.B. Le, “A survey of itemset mining”. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 7(4), 2017.

[12] B.Liu, W.Hsu and Y.Ma, “Mining association rules with multiple minimum supports”. In Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining ,ACM, August 1999.

[13] S.A. Abaya, “Association rule mining based on Apriori algorithm in minimizing candidate generation”, International Journal of Scientific & Engineering Research,3(7), 2012.

[14] F. Tao, F.Murtagh and M.Farid,“Weighted association rule mining using weighted support and significance framework”,In Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, August 2003.

[15] S.S.Thakur and S.Z.Ninoria, “An Improved Progressive Sampling based Approach for Association Rule Mining”, International Journal of Computer Applications, 165(7), 2017.

[16] R.U. Kiran and P.K. Re, “An improved multiple minimum support based approach to mine rare

association rules",IEEE Symposium on Computational Intelligence and Data Mining ,IEEE, March 2009.

[17] Y.C.Lee, T.P.Hong and W.Y. Lin, "Mining association rules with multiple minimum supports using maximum constraints". International Journal of Approximate Reasoning, 40(1-2), 2005.

[18] L.Szathmary, A.Napoli and P.Valtchev,"Towards rare itemset mining", In 19th IEEE International Conference on Tools with Artificial Intelligence (ICTAI 2007) Vol. 1, IEEE, October 2007.

[19] N.Hoque, B. Nathand and D.K. Bhattacharyya, "An efficient approach on rare association rule mining",In Proceedings of Seventh International Conference on Bio-Inspired Computing: Theories and Applications (BIC-TA 2012) Springer, India,pp. 193-203,2013.

[20] M.Adda,L. Wu and Y.Feng, "Rare itemset mining", Sixth International Conference on Machine Learning and Applications (ICMLA 2007), IEEE,2007.

[21] J. Pillai and O.P. Vyas, "Overview of itemset utility mining and its applications". International Journal of Computer Applications, 5(11), pp.9-13, 2010.

[22] J. Pillai and O.P. Vyas, "High Utility Rare Itemset Mining (HURI): An approach for extracting highutility rare item sets". Journal on Future Engineering and Technology, 7(1). 45 Haglin DJ, Manning AM (2007) On minimal infrequent itemset mining. In: DMIN, 2011.

[23] D.J. Haglin and A.M. Manning, "On Minimal Infrequent Itemset Mining". In DMIN, pp. 141-147, June 2007.

[24] L.Troiano, G.Scibelli and C.Birtolo, "A fast algorithm for mining rare itemsets",In 2009 Ninth International Conference on Intelligent Systems Design and Applications,IEEE, November 2009.

[25] H.Yun, D.Ha, B.Hwang and K.H.Ryu,"Mining association rules on significant rare data using relative support". Journal of Systems and Software, 67(3), 2003.

[26] K.Wang, Y. He and J.Han,"Pushing support constraints into association rules mining". IEEE Transactions on Knowledge and Data Engineering, 15(3), 2003.

[27] C.K. Selvi and A.Tamilarasi,"Mining association rules with dynamic and collective support thresholds". International Journal of Engineering and Technology, 1(3), 2009.

[28] K.S. Sadhasivam and T. Angamuthu, "Mining rare itemset with automated support thresholds". Journal of Computer Science, 7(3),2011.

[29] A. Borah and B.Nath, "Rare pattern mining: challenges and future perspectives",Complex & Intelligent Systems", 5(1), 2014.

[30] R.Chan, Q.Yang and Y.D. Shen,"Mining high utility itemsets". In Third IEEE international conference on data mining,IEEE, pp. 19-26, November 2003.

[31] S.Bhattacharya and D.Dubey ,"High Utility Itemset Mining", Int. J. Emerg. Technol. Adv. Eng. ISSN 2(8), 2250–2459, 2012.

[32] H.Yao and H.J.Hamilton, "Mining itemset utilities from transaction databases",Data & Knowledge Engineering, 59(3), 2006.

[33] C.H.Weng, "Mining fuzzy specific rare itemsets for education data",Knowledge-Based Systems, 24(5), 2011.

[34] L. Cagliero and P. Garza, "Infrequent weighted itemset mining using frequent pattern growth",IEEE transactions on knowledge and data engineering, 26(4), 2013.

[35] J.Wang and J.Han, "BIDE: Efficient mining of frequent closed sequences", In Proceedings. 20th international conference on data engineering IEEE., March 2004.

[36] U.Suvarna and Y. Srinivas,"Efficient High-Utility Itemset Mining over Variety of Databases: A Survey". In Soft Computing in Data Analytics Springer, Singapore, 2019.

[37] V. Kavitha and B.G. Geetha , "Review on high utility itemset mining algorithms", World Conference on Futuristic Trends in Research and Innovation for Social Welfare (Startup Conclave) IEEE., February, 2016.

[38] U.Yun, "On pushing weight constraints deeply into frequent itemset mining", Intelligent Data Analysis 13(2):359-83., 2009.

[39] U.Yun,"Efficient mining of weighted interesting patterns with a strong weight and/or support affinity", Information Sciences,177:17:477-99,Sep 1 2007.

[40] U.Yun and J.J. Leggett,"WFIM: weighted frequent itemset mining with a weight range and a minimum weight",In Proceedings of the SIAM international conference on data mining Society for Industrial and Applied Mathematics., April 2005.

[41] P.Fournier-Viger, C.W.Wu, S. Zida and V.S.Tseng, "FHM: Faster high-utility itemset mining using estimated utility co-occurrence pruning", In International symposium on methodologies for intelligent systems Springer, Cham, pp. 83-92,2014.

[42] C.W.Lin, T.P. Hong and W.H. Lu, "An effective tree structure for mining high utility itemsets". Expert Systems with Applications, 38(6), 2011.

[43] Y.C.Lin, C.W. Wu and V.S.Tseng, "Mining high utility itemsets in big data". In Pacific-Asia Conference on Knowledge Discovery and Data Mining Springer, Cham., 2015.

[44] M.Liu and J.Qu,"Mining high utility itemsets without candidate generation". In Proceedings of the 21st ACM international conference on Information and knowledge management ,ACM., October 2012.

[45] Y.Liu, W.K. Liao and A.Choudhary,"A two-phase algorithm for fast discovery of high utility itemsets". In Pacific-Asia Conference on Knowledge Discovery and Data Mining Springer, Berlin, Heidelberg., May 2005.

[46] U.Yun, H.Ryang and K.H.Ryu, "High utility itemset mining with techniques for reducing overestimated utilities and pruning candidates", Expert Systems with Applications, 41(8), 2014.

[47] S.Zida, P.Fournier-Viger, J.C.W.Lin, C.W. Wu and V.S.Tseng, "EFIM: a highly efficient algorithm for high-utility itemset mining", In Mexican International Conference on Artificial Intelligence Springer, Cham.,October 2015.

[48] J.C.W.Lin, W.Gan, P.Fournier-Viger, T.P. Hong, V.S. Tseng ,“ Mining potential high-utility itemsets over uncertain databases” In Proceedings of the ASE Big Data & Social Informatics, ACM,October, 2015.

[49] C.W.Lin, T.P. Hong and W.H. Lu,“The Pre-FUFP algorithm for incremental mining”,Expert Systems with Applications, 36(5), 2009.

[50] P. Fournier-Viger and S.Zida,“FOSHU: faster on-shelf high utility itemset mining with or without negative unit profit”, In Proceedings of the 30th Annual ACM Symposium on Applied Computing , 857-864, 2015.

[51] Q.H.Duong, B.Liao, P.Fournier-Viger and T.L. Dam, “An efficient algorithm for mining the top-k high utility itemsets using novel threshold raising and pruning strategies”, Knowledge-Based Systems, 104, 2016.

[52] V.S.Tseng, C.W.Wu, P. Fournier-Viger and S.Y.Philip, “Efficient algorithms for mining top-k high utility itemsets”,IEEE Transactions on Knowledge and data engineering, 28(1), 2015.

[53] S.Z. Ninoria and S.S.Thakur, “An Efficient Algorithm For Mining High Utility Rare Itemsets Over Uncertain Databases”, International Journal of Computer Engineering & Technology (IJCET), Volume 10, Issue 2, March-April 2019.

AUTHOR PROFILES

I. Ms. Shalini Zanzote Ninoria pursed Master of Computer Application from Nagpur University, India in year 2005.She has also pursued MPhil(CS) from R.D.University, Jabalpur, India. She is currently pursuing Ph.D.(CS) at Department of Mathematics and Computer Science, R.D.University, India since 2016 under the guidence of Dr. S.S. Thakur. She has published research papers in reputed international journals. Her main research work focuses on Data Mining, Association Rule Mining, High Utility Mining, and Rare Itemset Mining. She has 9 years of teaching experience and 4 years of Research Experience.

II. Dr.S.S.Thakur is currently designated as Principal and Professor & Head at Jabalpur Engineering College, Jabalpur, India with 35 years of Teaching and Academic experience. He has pursued Ph.D. from Dr. Hari Singh Gour University, India in 1982 and is a member of various computer societies with more than 247 research publications with one book of his credit in reputed national and international journals.Dr.S.S.Thakur has also guided many Research Scholars for the successful award of Ph.D. His main research work focuses on field of fuzzy topology, intuitionistic fuzzy topology and Data Mining.