RESEARCH ARTICLE OPEN ACCESS

AN EFFICIENT HHUARL ALGORITHM FOR HIDING ON INTERSECTION LATTICE

RAGIPANI SOWMYA*1, J S RADHIKA 2

[12]Assistent Professor, Dept of CSE, Sri Indu College of Engineering and Technology, Hyderabad, Telangana, India

Abstract— Hiding high utility sensitive association rule is a fundamental issue for protecting security knowledge from being uncovered while sharing information outside the gatherings. In any case, this issue has not been thought about attentively. This paper expects to propose a novel procedure for hiding high utility sensitive association rules in view of intersection lattice. The methodology incorporates two steps:(i) The exchanges containing the sensitive rule and having the minimum utility are chosen as casualty exchanges; (ii) The casualty things are indicated in view of a heuristic such that altering them causes minimal effect on a lattice of high exchange weighted utility itemsets. Depending on those means, the algorithm named HHUARL for hiding high utility sensitive association rules is proposed. The analysis demonstrates that symptoms caused by HHUARL algorithm are worthy.

Keywords—High utility association rules, High utility sensitive association rule, Intersection lattice

I INTRODUCTION

Mining High Utility Association Rule is a developing pattern in Data Mining. It goes for finding the relationship among itemsets depended on their utility. In 2015, Jayakrushna Sahoo et al. [13] proposed an approach for mining high utility association rules. In this paper, two measures of the rule were characterized as utility esteem and utility certainty. These attributes have been utilizing to assess the convenience of high utility association rules.

By and large, high utility association rule mining method incorporates two phases: (1) finding all high utility itemsets (their utility isn't not as much as least utility limit) [2, 7, 9, 10, 14, 15, 16], and (2) mining high utility association rules from those itemsets (their utility certainty isn't not as much as least utility certainty edge). Thang Mai et al. (2017) [17] proposed an algorithm for mining high utility association rule in light of the lattice of itemsets and clarified that their algorithm is more proficient than the one proposed by J. Sahoo et al. [13].

In today, worldwide exchanging coordinated effort and global business are new patterns of the world for maintainable advancement. In the organization together, endeavors need to share information each other with a specific end goal to enhance their business data. By sharing information, organizations can apply information mining into and improve their aggressive capacity. In any case, sharing information likewise may cause the sensitive knowledge certain inside database be uncovered to contenders. Hiding sensitive association rules is an observational procedure that permits safeguarding sensitive knowledge (which can be derived from sensitive association rules) from being uncovered before sharing information.

Hiding sensitive association rules causes symptoms [19]: Hiding Failures, Missing Cost, Appearing Ghost. In this way, the effectiveness of association rule hiding algorithms is assessed by the symptoms. The lower the symptoms, the better the algorithm. M. Atallah (1999) [3] was the primary scientist proposed association rule display with a specific end goal to safeguard sensitive knowledge in information mining. More effective algorithms were then proposed in view of Atallah's thought [6, 8].

The procedures for hiding sensitive association rules can't be connected straightforwardly into the issue of hiding high utility sensitive association rules in light of the fact that the prominent association rule is estimated by things show up in the database while the high utility association rule is estimated by utility. In 2010, Jieh-Shan Yeh et al. [12] proposed algorithms entitled HHUIF and MSICF for hiding sensitive high utility itemsets. The objective of HHUIF is to supplant the amount estimation of a thing with the highest utility incentive in a few exchanges containing the sensitive itemset. MSICF algorithm diminishes the quantity of altered things from the first database. The chose thing has most extreme clash include among things the sensitive itemsets.

Chun-Wei Lin et al. [4] proposed a GA-based strategy for hiding sensitive high utility itemsets. Keeping in mind the end goal to conceal the itemsets, the creators connected GA method to indicate fitting exchanges, at that point embed them into the database. In the proposed algorithm, the descending conclusion property and the substantial idea are received to diminish the cost of rescanning database, subsequently accelerating the valuation procedure of chromosomes. In 2017, Chun-Wei Lin et al. [5] proposed an algorithm for hiding sensitive high utility itemsets. By receiving a GA-based approach, the creators indicated precisely exchanges and after that expelled them from unique database to lessen the utility of sensitive itemsets under the edge.

Various algorithms for hiding high utility sensitive itemsets were proposed however there is no algorithm for hiding high utility sensitive association rule has been distributed. In this paper, we propose a strategy for hiding high utility sensitive association rule in view of intersection lattice named HHUARL (Hiding High Utility Sensitive Association Rules Based on Intersection Lattice).

II RELATED WORKS

High utility itemset mining is a procedure that finds all itemsets, of which the utility esteem isn't not as much as least utility edge given by the client. The itemset mining methods can't be connected into a high utility itemset mining problem[7] on the grounds that qualities of incessant itemsets are not quite the same as high utility itemsets. In 2004, Hong Yao, Howard J. Hamilton [9] proposed a strategy for mining high utility itemsets. Basing on this model, Hong Yao, Howard J. Hamilton [10] proposed algorithms entitled Using and UmingH which decrease seeking space by falling applicant itemsets. In 2005, Liu. Y, Liao. W, A. Choudhary [14] proposed a Two-Phase algorithm for mining high utility itemsets in view of two stages process. The creators displayed ideas about exchange utility(TU) esteem and exchange weighted use itemsets(TWU) to prune the looking space of high utility itemsets. An essential property of TWU is that TWU of itemsets fulfills the descending conclusion property. In view of this property, Two-Phase algorithm quickly lessens seeking space. In any case, the Two-Phase algorithm creates a high number of applicants.

Mengchi Liu, Junfeng Qu [15] proposed HUI-Miner algorithm and a novel structure (called utility-list) for mining high utility itemset. Rather than producing a possibility for high utility itemsets, HUI

- Miner makes an underlying utility-rundown and after that bases on this rundown to figure high utility itemsets.

Vincent S. Tseng et al. [18] proposed UP-Growth and UP-Growth+ algorithms for mining high utility itemsets in view of an arrangement of powerful procedures for pruning hopeful itemsets. An information structure called UP-Tree is made to keep up data of high utility itemsets, so competitor itemsets can be produced proficiently utilizing just two sweeps of the database.

Philippe Fournier-Viger[7] proposed an expanded form of the Hui-Miner algorithm (named FHM) that utilized a novel pruning system named EUCP (Estimated Utility Cooccurrence Pruning) to prune itemsets without performing joins.

Souleymane Zida et al. [16] proposed a novel algorithm for high-utility itemset mining named EFIM. It depends on two upper-limits called the sub-tree utility and nearby utility to prune the hunt space adequately. It likewise presents a novel cluster based utility tallying approach named Fast Utility Counting to ascertain these upper-limits in straight time and space.

B. HIGH UTILITY ASSOCIATION RULE MINING

Mining high utility association rule goes for discovering all association rules from the arrangement of high utility itemsets.

Jayakrushna Sahoo et al. [13] characterized the issue of discovering association rules utilizing the utility-certainty system. They proposed an algorithm to create the utility based non-excess association rules and a strategy for recreating all high utility association rules. This approach incorporates three stages: (1) mining high utility shut itemsets(HUCI) and generators; (2) creating high utility nonexclusive fundamental association rules (HGB); and (3) mining all high utility association rules in light of HGB. Depending on this base, Thang Mai et al.[17] proposed an algorithm for mining high utility association rules utilizing a lattice of itemsets. This approach in light of two stages:

(1) building a high utility itemset lattice (HUIL); and

(2) mining all high utility association rules from the HUIL. The demonstrate that their algorithm was superior to anything the algorithm proposed in [13].

C. HIDING HIGH UTILITY SENSITIVE ASSOCIATION RULE

Hiding high utility sensitive association rule intends to make the sensitive association rules can't be mined by cleaning database while limiting the reactions. As a result of contrast between attributes of well known association rule and high utility association rule, the systems for hiding prominent sensitive association rules can't be connected straightforwardly into hiding high utility association rule.

III THE ALGORITHM FOR HIDING HIGH UTILITY SENSITIVE ASSOCIATION RULES

A. Lattice-based approaches for hiding high utility sensitive association rules

Basing on lattice theory [11], Hai Quoc Le et al.[8] proposed a method to build the lattice of frequent itemsets. The authors defined theory of intersection lattice and based on properties of such lattice to propose sensitive association rule hiding algorithms. Their method aims to protect the intersection lattice, by reducing affects on the generation itemsets (the GEN), while modifying data in order to decrease side effects of the rule hiding process.

We can construct an intersection lattice of frequent itemsets relying on Apriori property but we cannot do the same with high utility itemsets. However, the set of itemsets having high transaction weighted utility is an intersection lattice because it satisfies Apriori property (Property 3). Figure 1 demonstrates the intersection lattice of high transaction weighted utility itemsets presented in Table IV.

he lattice of $HTWU^D$ always contains the set of high utility itemsets (Property 4). This shows that if the lattice of $HTWU^D$ is protected from the affects of high utility sensitive association rule hiding process, then the high utility itemsets and high utility non-sensitive association rules will be protected.

This paper applies association rule hiding based on intersection lattice [8] into hiding high utility sensitive association rule problem.

The following contents reminds the basic of lattice theory and intersection lattice [8]. Based on these theories, we construct the intersection lattice of $HTWU^D$.

Definition1: An ordered set $(L; <)$ is said to be a lattice if $inf\{a, b\}$ and $sup\{a, b\}$ exist for all $a, b \in L$, and are denoted by $a \vee b$ and $a \wedge b$.

Definition 2: Let $\mathbf{L} = (L;\subseteq)$, if $\mathbf{L}$ satisfies $\inf(X, Y) = X \cap Y$ for all X, Y then $\mathbf{L}$ is said to be an intersection lattice.

We have, for all $X, Y \in \mathbf{L}$ then $X \cap Y \in \mathbf{L}$. In the other words, intersection lattice $(L;\subseteq)$ is closed to intersection operation.

Theorem 1. $HTWU^D$ formed an intersection lattice.

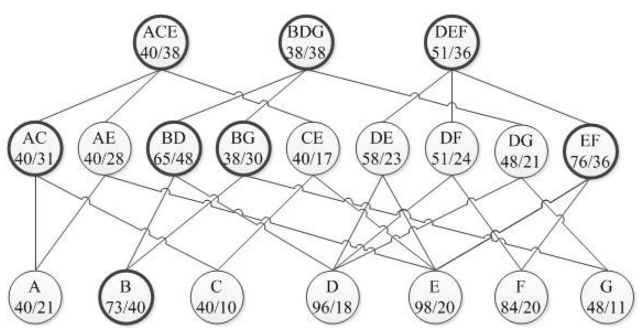

Fig. 1. The intersection lattice of $HTWU^D$ presented in Table III (the bold border nodes are high utility itemsets, the value inside each node shows the *HTWU* itemset and its *TWU*/Utility)

Definition 3: The generated set of $HTWU^D$, denoted by $Gen(HTWU^D)$, is a smallest subset of $HTWU^D$ such that for every itemset in $HTWU^D$ can be generated by intersection of some itemsets in $Gen(HTWU_D)$. In the other words,

$HTWU^D=\{X \backslash X=\cap_{K \in N*} Y^K, Y^K \in Gen(HTWU^D)\}$

The Set Gen ($HTWU^D$) Can Be Calculated AS FOLLOWS :

$Gen(HTWU^D)=\{X \in HTWU^D | d(X)\}<1\}$ Where

$d(X)=|\{Y \in HTWU^D | X \subseteq Y\}|$

Exaple 1

Example 1. Consider the set $HTWU^D$ from Table IV, we have:

$$Gen(HTWU^D) = \begin{Bmatrix} ACE, BDG, DEF, AC, AE, BD, BG, CE, \\ DE, DF, DG, EF \end{Bmatrix}$$

Definition 4: For each set $HTWU^D$, the set containing maximal subset of $HTWU^D$ is said to be *Coatom*($HTWU^D$), and is defined as follow:

$$Coatom(HTWU^D) = MAX(Gen(HTWU^D))$$

Example 2: Consider the set $HTWU^D$ from Table IV, we have:

$$Coatom(HTWU^D) = \{ACE, BDG, DEF\}.$$

Each itemset

$X \in HTWU^D, Gen(X) = \{X \backslash I_k, I_k \in X, k = 1 ... |X|\}$ is a

generated set of $Poset(X) \backslash \{X\}$

By the property 3, the itemsets contained in *Gen*($HTWU^D$) have the least transaction weighted utility in $HTWU^D$. These itemsets are easier impacted when reducing utility value of itemsets in the intersection lattice. Moreover, if *Gen*($HTWU^D$) is maintained during hiding process then all itemsets in $HTWU^{D'}$ will be high transaction weighted utility itemsets. Therefore, maintaining itemsets in *Gen*($HTWU^D$) from theintersection lattice while hiding process will help to minimize the side effects.

B. Proposed methodology for hiding utility sensitive association rule

C. Proposed methodology for hiding utility sensitive association rule

In order to hide a sensitive rule R_S: $X \rightarrow Y$, our method is to

modify some transactions in original database to reduce utility confidence of the rule R_S to below utility confidence threshold. The best way is to reduce numerator (means to reduce

luv(X,XY)) in such away that $\frac{luv(X, XY)}{u(X)} < \beta$. To reduce *luv(X,XY)*, we need to reduce number of transactions containing *XY*. This means we need to remove item in *Y* to let *luv(X,XY)* to be reduced but *u(X)* not be reduced.

Side effects caused by data modification depend on the victim transaction and item selection process. Our method, therefore, based on two steps:

Step 1: Victim transaction selection

T_{victim} denotes the set of transactions containing a sensitive high utility association rule. For $K = luv(X, XY)$, when an

item I_i, $I_i \in Y \wedge XY \in T_{victim}$, is removed from the victim

transaction T_{victim} then K will be reduced, $K = K - u(X, T_{victim})$ and utility confidence of the sensitive associaiton rule R_S will

be $uconf(R_S) = \frac{K - u(X, T_{victim})}{u(X)}$. In order to achieve

$(dr = uconf(R_S) - \beta) < 0$ and reduce the side effects,

transactions having the least $u(X, T_{victim})$ among transactions containing *XY* and having $\frac{u(X, T_{victim})}{u(X)} > dr$ will be selected as

victim transactions. If such transactions do not exist, transactions having maximal $u(X, T_{victim})$ among transactions containing *XY* will be selected as victim transactions.

Step 2: Victim item selection based on intersection lattice Removing item $I_i \in Y$ from a victim transaction selected at

step 1 will impact on itemsets containing I_i. To minimize the side effects, we propose a method to select victim items based on intersection lattice of high transaction weighted utility itemsets as follows:

Input: $HTWU^D$

Output: *Victim item*

Step 2.1: Select candidate victim item

```
1. compute Gen(HTWU^D);
2. compute Coatom(HTWU^D);
```

3. foreach $(C \in Coatom(HTWU^{D})$ and $XY \subseteq C)$

4. foreach $(I_i \in Y)\{$

5. $tuple_{\min}(I_i, C) = (I_i, C, G, \lambda)$,

$\lambda = \min(TWU(G)) \mid I_i \subseteq Y \cap G, G \in Gen(C)$;

6. $CAN_{\min}(X \rightarrow Y) \leftarrow tuple_{\min}(I_i, C)$;

7. }

Step 2.2: Specify victim item

8. $tuple_{\max}(X \rightarrow Y) = (I_i, C, G, \mu) \mid (I_i, C, G, \mu) \in CAN_{\min}(X \rightarrow Y) \wedge \mu = \max(\lambda)$;

9. $I_{victim} = \{I_i, I_i \in tuple_{\max}(X \rightarrow Y)\}$;

10. Return I_{victim};

Example 3. Suppose that a sensitive rule is $F \rightarrow DE$, we need to modify item *D* or *E* to hide the rule. Running the proposed method gives results as follows:

$Coatom(HTWU^{D}) = \{ACE, BDG, DEF\} \Rightarrow C = DEF$,

$Gen(DEF) = \{DE, DF, EF\}$

$I_i = D \Rightarrow tuple_{\min}(D, DEF) = \min\{(D, DEF, DE, 58), (D, DEF, DF, 51)\} = (D, DEF, DF, 51)$

$\Rightarrow CAN_{\min}(F \rightarrow DE) = \{(D, DEF, DF, 51)\}$

$I_i = E \Rightarrow tuple_{\min}(E, DEF) = \min\{(E, DEF, DE, 58), (E, DEF, EF, 76)\} = (E, DEF, DE, 58)$

$\Rightarrow CAN_{\min}(F \rightarrow DE) = \{(D, DEF, DF, 51), (E, DEF, DE, 58)\}$

$\Rightarrow tulpe_{\max}(F \rightarrow DE) = (E, DEF, DE, 58) \Rightarrow I_{victim} = E$

Consequently, victim item is *E*.

C. Algorithm pseudocode

The HHUARL algorithm

Input: D (transaction database), HRS^{D} (The set of high utility sensitive association rules), β (Minimum confidence utility threshold)

Output: D' (The modified database)

foreach ($R_S : X \rightarrow Y \in HRS^D$){

  $dr = uconf(R_S) - \beta$;

  project $D_{RS} \leftarrow (D, R_S)$;

  while($dr \geq 0$){

    $T_{victim} \leftarrow \min\{u(X, T_c) \mid XY \subseteq T_c \wedge T_c \in D_{R_s} \wedge \frac{u(X, T_c)}{u(X)} > dr\}$;

    if(! $\exists T_{victim}$) $T_{victim} \leftarrow \max\{u(X, T_c) \mid XY \subseteq T_c \wedge T_c \in D_{Rs}\}$;

    i_{victim} = Victim item selection based on lattice ;

    delete i_{victim} from T_{victim} ;

    $dr = dr - \frac{u(X, T_{victim})}{u(X)}$;

    update D_{RS} ;

  }

  Remove R_S from HRS^D

}

D. Algorithm pseudocode

D. Experimental results

1) Dataset description

Experiment was executed with Foodmart database[20] described as follows:

Foodmart dataset contains: 4141 transactions, 1559 items. The maximal transaction contains 11 items, average transaction length is 4.

The set of 5 sensitive high utility association rules was randomly selected. The experiment result is presented in Table VI.

2) Performance analysis

For Privacy preserving utility mining (PPUM), Jieh-Shan Yeh, Po -Chiang Hsu[12] used three standard side effects in order to evaluate the performance of a sanitisation method, namely Hiding Failures (HF), Missing Cost (MC), Appearing Ghost (AG) and ***Difference between the Original and Sanitized Databases (DIFF)***. To measure the effectiveness of our proposed hide high utility sensitive association rule algorithms, we use four units of measure as follows:

a) *Hiding Failures (HF): The ratio of high utility sensitive association rules that are disclosed after and before the sanitization process. The hiding failure is calculated as follows:*

$$HF = \frac{\left|HRS^{D'}\right|}{\left|HRS^{D}\right|} \quad (1)$$

Where HRS_D *and* $HRS^{D'}$ *denote the set of high utility*

sensitive association rule discovered from the original database D and the sanitized database D' respectively.

b) *Missing Cost (MC): The ratio of high utility non-sensitive association rules that are disclosed before and after the sanitization process. The MC is measured as follows:*

$$MC = \frac{\left|HNSR^{D}\right| - \left|HNSR^{D'}\right|}{\left|HNSR^{D}\right|} \quad (2)$$

Where $HNSR_D$ *and* $HNSR^{D'}$ *denote the set of high utility non-sensitive association rule discovered from the original database D and sanitized database* D' *respectively*

c) Appearing Ghost(AG): The ratio of high utility non-sensitive association rules can not discovered from original database D but discovered form sanitized database D' *and high utility association rules discovered from original database D. The AG is measured as follows:*

$$AG = \frac{\left|HR^{D'}\right| - \left|HR^{D} \cap HR^{D'}\right|}{\left|HR^{D'}\right|} \quad (3)$$

Where HR^D *and* HR_D' *denote the set of high utility association rule discovered from the original database D and the sanitized database D'.*

d) ***Difference (DIFF)****: The difference ratio between the original database D and sanitized database D', is given by:*

$$DIFF(D, D') = \frac{1}{\sum_{i=1}^{n} f_D(i)} \left(\sum_{i=1}^{n} \left[f_D(i) - f_{D'}(i) \right] \right) \quad (4)$$

here $f_D(i)$ and $f_{D'}(i)$ represent the frequency of i^{th} item in

original database D and sanitized database D', n is the number of distinct items in the original database D.

TABLE VI. PERFORMANCE OF HHUARL ALGORITHM FOR DIFFERENT MINIMUM THRESHOLD VALUES

MinUtility=7,500; MinUtilityConfidence=0.6 $\|HUI^D\|$=1,098; $\|HR^D\|$=1,042,625; $\|HRS\|$=5			
HF	MCs	AG	DIFF
0%	2.735115694%	0%	0,027%
MinUtility=7,500; MinUtilityConfidence=0.7 $\|HUI^D\|$=1,098; $\|HR^D\|$=1,041,506; $\|HRS\|$=5			
HF	MC	AG	DIFF
0%	2.736038007%	0%	0,0220%
MinUtility=7,500; MinUtilityConfidence=0.8 $\|HUI^D\|$=1,098; $\|HR^D\|$=1,041,471; $\|HRS\|$=5			
HF	MC	AG	DIFF
0%	2,811679100%	0%	0,0220%

The results shows that HHUARL algorithm achieves good performance in hiding the sensitve rules with minimal size effects. In the most cases, HF and AG are at 0% indicates that all sensitive rules were hidden and no ghost rule are generated in the sanitized database. The MCs are lower than 3% while the DIFFs are lower than 0.03% giving a high confidence to the senitized database. The sharing database therefore still gain high quality in mining high utility association rules.

CONCLUSION

This paper proposes the first run through a strategy for hiding high utility sensitive association rule keeping in mind the end goal to secure protection knowledge in information mining when sharing database outside gatherings. A heuristic algorithm entitled HHUARL, which depends on the intersection lattice of itemsets having high exchange weighted utility for determining a casualty thing is proposed. The heuristic is to movement on the lattice to determine a casualty thing such that adjusting it makes the slightest effect the lattice. This technique goes for limiting reactions of hiding process. The exploratory outcomes show that HHUARL is productive in limiting reactions.

REFERENCES

[1] Agrawal, T. Imielinski, A. Swami, Mining association rules between sets of items in large databases, In Proc of the 1993 ACM SIGMOD International Conference on Management of Data, pp. 207-216,1993.

[2] Erwin A., Gopalan R., Achutan, Efficient Mining of High Utility Itemsets from Large Datasets, T. Washio: PAKDD 2008, LNAI 5012, pp. 554-561, 2008.

[3] M. Atallah, E. Bertino, A. Elmagamind, M. Ibrahim, and V. S. Verykios: Disclosure limitation of sensitive rules, .In: Proc. of the 1999 IEEE Knowledge and Data Engineering Exchange Workshop, pp. 45-52, 1999

[4] Chun-Wei Lin, Tzung-Pei Hong, Jia-WeiWong, Guo-Cheng Lan, Wen-Yang Lin, A GA-Based Approach to Hide Sensitive High Utility Itemsets, In Hindawi publishing corporation, the scientific world journal volume 2014, 2014

[5] Chun-Wei Lin, Tzung-Pei Hong, Philippe Fournier-Viger, Qiankun Liu, Jia-Wei Wong & Justin Zhan, Efficient hiding of confidential high-utility itemsets with minimal side effects, In Journal of Experimental & Theoretical Artificial Intelligence, pp.1225-1245, 2017.

[6] C. Dong and L. Chen, A fast secure dot product protocol with application to privacy preserving association rule mining, In Pacific-Asia Conference on Knowledge Discovery and Data Mining, Springer International Publishing, pp. 606-617, 2014.

[7] Fournier-Viger, P., Wu, C.-W., Zida, S., Tseng, V. S., FHM: Faster high-utility itemset mining using estimated utility co-occurrence pruning, In Proc. 21st Intern. Symp. on Methodologies for Intell. Syst., pp. 83-92, 2014.

[8] Hai Quoc Le, Somjit Arch-int, and Ngamnij Arch-int, Association Rule Hiding Based on Intersection Lattice, In Hindawi Publishing Corporation, Mathematical Problems in Engineering Volume 2013, pp. 776-784, 2013.

[9] Hong Yao, Howard J. Hamilton: A foundational Approach to Mining

Itemset Utilities from Databases, In 4th SIAM International Conference on Data Mining, Florida USA, 2004.

[10] Hong Yao, Howard J. Hamilton, Mining Itemset Utilities from Transaction Databases, In Journal Data & Knowledge Engineering, Volume 59 Issue 3, pp. 603 – 626, 2016.

[11] George Grätzer: Lattice Theory: Foundation. 3nd edn. Springer Basel AG, 2011.

[12] Jieh-Shan Yeh, Po-Chiang Hsu, HHUIF and MSICF: Novel algorithms for privacy preserving utility mining, In Expert Systems with Applications, Vol 37, pp: 4779–4786, 2010.

[13] Jayakrushna Sahoo, Ashok Kumar Das, A. Goswami, An efficient approach for mining association rules from high utility itemsets, In Expert Systems with Applications, 2015

[14] Liu. Y, Liao. W,A. Choudhary, A fast high utility itemsets mining algorithm, In: Proceedings of the Utility-Based Data Mining Workshop, August 2005.

[15] Mengchi Liu, Junfeng Qu, Mining high utility itemsets without candidate generation, In Proceedings of the 21st ACM international conference on Information and knowledge management, pp:55-64, 2012

[16] Souleymane Zida , Philippe Fournier-Viger, Jerry Chun-Wei LinCheng-Wei WuVincent S. Tseng, EFIM: A Fast and Memory Efficient Algorithm for High-Utility Itemset Mining, In Knowledge and Information Systems, Volume 51, Issue 2, pp:595-625, 2017.

[17] Thang Mai, Bay Vo, Loan T.T.Nguyen, A lattice-based approach for mining high utility, In: Information Sciences, Volume 399, Issue C. Elsevier, pp. 81-97, 2017

[18] Vincent S. Tseng, Bai-En Shie, Cheng-Wei Wu,Philip S. Yu, Efficient Algorithms for Mining High Utility Itemsets from Transactional Databases, In IEEE Transactions on Knowledge and Data Engineering, Volume 25, Issue 8, pp. 1772–1786, 2013

[19] Wu, Y. H., Chen, A. L. P., Hiding sensitive association rules with limited side effects. IEEE Transactionson Knowledge and Data Engineering, 19, 29–42, 2007

[20] http://www.philippe-fournier-viger.com/spmf/datasets/foodmart.txt