Defending the Sensitive Data using Lattice Structure in Privacy Preserving Association Rule Mining

Bonam Janakiramaia$h$[1] , Dr A.RamaMohan Redd$y$[2] , G Kalyan$i$[3]

[1,3]*Department of computer science and Engineering, DVR and Dr HS MIC College of Technology, Vijayawada.*

[2]*Department of Computer Science and Engineering, S.V.University, Tirupathi.*

Abstract Innovation of association rules from enormous databases ensures benefits for the enterprises since such rules can be very operative in enlightening the knowledge that leads to tactical decisions. Association rule mining has acknowledged a proportion of attention in the collaborative business community and several algorithms were proposed to improve the performance of association rules or frequent itemset mining. The man-made data generators have been generally used for performance estimation. Latest works shows that the data generated is not worthy sufficient for standardizing as it has very dissimilar characteristics from real-world data sets. Hence forth there is an abundant need to use real-world data sets as standard. But, organizations hesitate to provide their data due to privacy concerns.Privacy preserving association rule mining addresses this problem by transforming the real data sets to hide sensitive or secretive rules. Though, transforming sensitive data in real data may influence other non-sensitive rules. One essential feature of privacy preserving association rule mining is the fact that the mining process deals with a trade-off between privacy and accuracy, which are typically conflicting, and improving one typically incurs a cost in the other. In this paper, we present a novel algorithm for balancing privacy and knowledge discovery in association rule mining. We use the concepts of sensitivity of the transaction and itemset lattice, to identify the transactions that are to be transformed and the item that is to be transformed respectively.The algorithm is experimentally assessed with a real data set and a synthetic data set. The analysis illustrate that our methodology is effective and efficient for restructuring real world data sets for a given set of sensitive association rules while preserving non-sensitive association rules.

Keywords Lattice Structure, Privacy Preserving, Accuracy, Sensitive Data, Impact-Factor

1 Introduction

Modern computers can usually collect, examine, and store millions of data in enormous transactional data warehouses. In several cases, the analysis of these masses of data using data mining tools may be evidenced to be beneficial for the data holder and, perhaps, for a large community of people. An area of data mining, named association rule mining mines innovative, hidden and advantageous patterns from massive sources of data. These patterns are useful for effective study and decision making in telecommunication network, marketing, business, medical analysis, website linkages, financial transactions, advertising and other applications. The sharing of frequent rules can bring lot of advantages in industry, research and business collaboration. At the same time, a massive repository of data comprises secretive data and sensitive rules that must be secured before sharing. On demand to various uneven requirements of data sharing, privacy preserving and knowledge discovery, Privacy Preserving Data Mining (PPDM) has become a research hotspot in data mining. Simply, the association rule hiding problem is to hide secret or sensitive rules in data from being exposed, while without losing non-sensitive rules at the same time. The problem of frequent association rules hiding motivated many authors [2], [5], [6], and they proposed different approaches. The majority of the proposed approaches can be classified into two principal research directions: (i) Methodologies based on Data and (ii) Methodologies based on knowledge.

1.1 Methodologies based on Data

These methods [7], [8] accumulate the approaches that discover how the privacy of raw data, or information, can be preserved in advance to the course of mining the data. The approaches of this kind targets at the elimination of sensitive or private information from the original data prior to its discloser and functions by relating the techniques such as perturbation, transformation, sampling and generalization etc.

1.2 Methodologies based on knowledge.

These methods encompass approaches that aim to protect the sensitive data mining results rather than the raw data itself, which were generated by the application of data mining tools on the original database. These can be further categorized into two sub groups: Data Distortion and Data Blocking techniques. Data Distortion [11], [3] is applied by removing or adding items to reduce the support or confidence of the sensitive rule, while Data Blocking [6] is employed by substituting certain items with a question mark (?) to make the support of the sensitive rule ambiguous.

2 Review of Related Work

Distortion based methods operate by selecting specific items to include(or exclude from) to the selected transactions of the original database in order to facilitate the hiding of the sensitive frequent itemsets. Two of the most commonly employed strategies for data distortion involve the swapping of values between transactions [3], as well as the deletion of specific items from the database [6].

Atallah [3] were the first to propose an algorithm for the hiding of sensitive association rules through the reduction in the support of their generating itemsets.

Dasseni [2] generalize the hiding problem in the sense that they consider the hiding of both sensitive frequent itemsets and sensitive association rules. The authors proposed three single rule heuristic hiding algorithms that are based on the reduction of either the support or the confidence of the sensitive rules, but not both. In all three approaches, the goal is to hide the sensitive rules while minimally affecting the support of the non-sensitive itemsets.

Oliveira [6] were the first to introduce multiple rule hiding approaches. The proposed algorithms are efficient and require two scans of the database, regardless of the number of sensitive itemsets to hide. During the first scan, an index file is created to speed up the process of finding the sensitive transactions and to allow for an efficient retrieval of the data. In the second scan, the algorithms sanitize the database by selectively removing the smallest amount of specific items that accommodate the hiding of the sensitive knowledge. Three item restriction-based algorithms (known as MinFIA, MaxFIA, and IGA) are proposed that selectively remove items from transactions that support the sensitive rules.

A more efficient approach than that of [6] and the work of [2] , [13] was introduced by [10]. The projected algorithm, called SWA, is a well-organized, scalable, one-scan heuristic which targets at providing a balance between the requirements of privacy and knowledge discovery in association rule hiding. It attains to hide multiple rules in only one pass through the dataset, regardless of its size or the number of sensitive rules that need to be protected.

Amiri [1] proposes three effective, multiple association rule hiding heuristics that outperform SWA by offering higher data utility and lower distortion, at the expense of increased computational speed. The first approach, called Aggregate, computes the union of the supporting transactions for all sensitive itemsets. Among them, the transaction that supports the most sensitive and the least non-sensitive itemsets is selected and ejected from the database. The same process is repeated until all the sensitive itemsets are hidden. Similar to this the second approach called Disaggregate approach intentions at eliminating specific items from transactions, rather than removing the entire transaction. The third approach, called Hybrid, is a combination of the two previous algorithms.

Wang [14] projected two data modification algorithms that aim at the hiding of predictive association rules, i.e. rules containing the sensitive items on their left hand side (rule antecedent). The first strategy, called ISL, decreases the confidence of a sensitive rule by increasing the support of the itemset in its left hand side. The second approach, called DSR, reduces the confidence of the rule by decreasing the support of the itemset in its right hand side (rule consequent). George V, et al. [12] proposed a new algorithmic method for sanitizing raw data from sensitive knowledge in the situation of mining association rules. The new approach (MaxMin2) (a) depends on the maxmin principle which is a method in decision theory for maximizing the minimum gain and (b) builds upon the border theory of frequent itemsets.

T.-P. Hong, et al. [9] proposed a novel greedy-based approach called Sensitive Items Frequency-Inverse Database Frequency (SIF-IDF) to evaluate the grade of transactions associated with given sensitive itemsets. It uses concept of TF-IDF for decreasing the frequencies of sensitive itemsets in data sanitization. Based on the greedy SIF-IDF algorithm, the user-specific sensitive itemsets can be completely hidden with reduced side effects.

3 Problem Formulation

We concentrated on the knowledge hiding thread of PPDM and revision on specific category of approaches which are collectively known as association rule hiding approaches. In the perspective of privacy preserving association rule mining, we do not focused on privacy of individuals; rather, we focused on the problem of defending sensitive knowledge mined from databases. The sensitive knowledge is represented by a special group of association rules called sensitive association rules. These rules are most important for tactical decision and must remain private (i.e., the frequent rules are private to the owner of the data). The problem of protecting sensitive knowledge in transactional databases draw the hypothesis that data owners have to know in advance some knowledge (frequent itemsets and/or rules) that they want to defend. Such rules are essential in decision making, so they must not be revealed. The problem of defending sensitive knowledge in association rule mining can be stated as, given a data set D to be released, a set of association rules R mined from D, and a set of sensitive itemsets or rules,$R_s \subseteq R$ to be hidden, how can we get a new data set D^1 such that the rules in R_s cannot be mined from D^1, while the rules in $R - R_s$ can still be mined as many as possible. In this case, D^1 becomes the released database.

4 Proposed Framework

The projected framework initially aims at identifying the association rules R by using any association rule mining algorithm (AR) from the given data set D. Subsequently the data owner will identify the sensitive rules R_s , which need to be concealed from mining. By taking into consideration the sensitive rules and original dataset as input, our projected algorithm releases a sanitized dataset D^1. After that by applying any association rule mining algorithm on the sanitized dataset D^1, all association rules which are mined from original dataset D except the sensitive rules can be mined. The projected framework is shown in Fig. 1.

5 The Proposed Algorithm

The algorithm uses itemset lattice and impact factor of items in the sensitive association rules to decide the victim item to hide the sensitive rules.

5.1 Itemset lattice.

We adopt lattice theory that is presented in [4]. Let I be a predetermined non-empty itemset. It is clear that the power set of I, denoted by Power-set(I) is an ordered set under the relation $\subseteq$ .It can be proved that (Power-set(I); $\subseteq$) forms a lattice, where sup(a, b)=$a \cup b$ and inf(a,b)=$a \cap b$. If X $\subseteq$ I and (X; $\subseteq$) is a lattice satisfying the properties that sup (a,b)= $a \cup b$ and inf(a,b)=$a \cap b$ for all a and b, then(X; $\subseteq$) is called a set lattice. Likewise if (Y; $\subseteq$) is a semi-lattice satisfying inf (a, b) = $a \cap b$, for all a and b, then (X; $\subseteq$) is said to be intersection lattice. It is obvious that intersection of elements in an intersection lattice (X; $\subseteq$) belongs to X. In other words, an intersection lattice (X; $\subseteq$) is closed under the intersection operator. Let FIS be a set of frequent item sets. By the Apriori property, if P, Q $\in$ FIS, then P $\cap$Q $\in$FIS. It can be inferred that FIS is an intersection lattice.

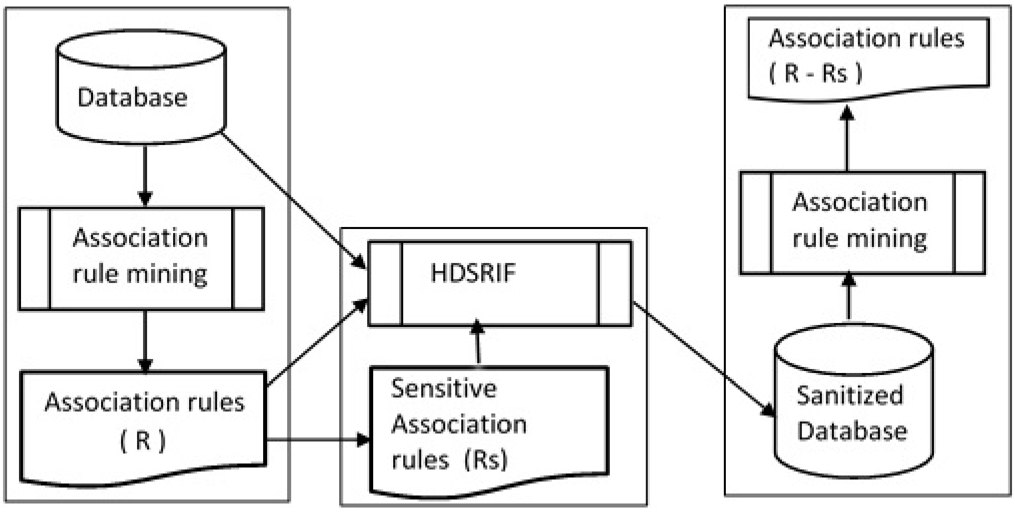

Fig. 1. Proposed Framework.

5.2 HDSRIF (Heuristic for Defending Sensitive Rules using Impact Factor) Algorithm.

Let R_s be the sensitive association rules. Suppose that the sensitive rule that wants to be concealed each time is denoted by A ⇒ B. Let γ(A ⇒ B) be the support of A ⇒ B and δ $(A \Rightarrow B)$ be the confidence of $A \Rightarrow B$. Our scheme intention is hiding $A \Rightarrow B$ by altering an item in B from a number of transactions until γ $(A \Rightarrow B)$ $<\theta$ and δ $(A \Rightarrow B)$ $<\rho$ where θ be the Minimum Support Threshold (MST) and ρ be the Minimum Confidence Threshold (MCT). The algorithm states the victim item by using the concept of impact factor of an item in consequent i.e RHS of the sensitive rule. The impact factor of an item is equal to the number of non-sensitive association rules that are effected by eliminating that item from the essential number of transactions. Lessen the impact factor of an item, lesser is its effect on the non-sensitive association rules.

Step 1: Selection of a rule

All the rules in the set R_s are considered one after the other for hiding process until R_s is empty. To select a rule the process is as follows: Calculate the frequencies of items (number of times each item occurs) in the sensitive rules. Select a rule which contains more number of highest frequency items because a rule with highest frequency items may affect the other rules in the set of sensitive association rules which may also contains the same items.

Step 2: Recognizing the essential number of transactions

This step objective is to compute the essential minimum number of transactions that are to be modified to hide the given sensitive rule. Let this number be denoted by T_n. Then to hide the rule $A \Rightarrow B$, our requirement is

$\gamma(AB) - T_n \; \theta$ or $(\gamma(AB) - T_n \; / \; \gamma(A)\rho$

$\Rightarrow T_n > \gamma(AB) - \theta$ or $T_n > \gamma(AB) - [\gamma(A) * \rho]$

Thus $T_n = min\{\gamma(AB) - \theta + 1, \gamma(AB) - [\gamma(A) * \rho] + 1\}$.

Moreover identifying the order of transactions for altering the selected item is an important step in reducing the side effects. To achieve this, for every transaction in the transactional data base compute Sensitive Item Frequency (SIF) with respect to all the items in the sensitive rules. For a transaction the sensitive item frequency can be calculated as number of sensitive items in the rule divided with length of the transaction. Thus to achieve the slightest impact on the non-sensitive association rules, data base is to be sorted in descending order of SIF.

Step 3: Victim Item Selection.

The victim item is the item that is to be removed to hide a rule such that removing this item minimizes the effect on non-sensitive association rules. The victim item will be selected from right hand side of the rule $A \Rightarrow B$ i.e B. For every item in B, calculate the Impact-Factor. Impact-Factor of an item can be defined as the number of non-sensitive association rules that gets effected when the item is removed from essential number of transactions. The item with minimum Impact-Factor will be selected as victim item because it effects minimum number of non-sensitive association rules.

Step 4: Updating the transactional data set and updating the set of association rules R and sensitive rules R_s.

The victim item is removed from T_n transactions which are supporting $A \Rightarrow B$ by using Modify function. After modifying the dataset, update the support counts of association rules and sensitive rules by using Modify-Rules function.

5.3 Algorithm HDSRIF

Input `The dataset D ; Minimum support threshold (MST),` θ `; Minimum confidence threshold (MCT),`ρ `; Frequent association rules, R; Set of association rules to be hidden,` R_s.

Output `Sanitized Data Set` D^1

Method:

```
Step 1. Repeat
For each rule r in $R_s$
    For each item i in r
        IF(i $\in$ X[ ] ) then
            count[i]++;
        else
            insert i to X[];
        End IF
    End For
End For
Select the rule from $R_s$ with maximum number of highest frequency items;
Step 2: For each transaction i in D
    SIF[i] = number of sensitive items / length of i.
End For
$D^1$= Sort (D) //in descending order of SIF
$T_n = min\{\gamma(AB) - \theta + 1, \gamma(AB) - [\gamma(A) * \rho] + 1\}$.
N [] = First $T_n$ transactions from $D^1$.
Step 3: For each item p ∈ B
    For each rule q ∈ R
        IF p ⊆ q
            Add q to TR;
        End IF
    End For
    For each rule q ∈ TR
        $\gamma(q) = \gamma(q) - \gamma 1(q)$ where $\gamma 1$(q)=support of r with respect to N
    End For
    count=0;
    For each rule j in TR
        IF ( $\gamma((j) < \theta)$ or $(\delta(j) < \rho)$)
            count=count+1;
        End IF
    End For
    Impact-Factor(p)=count.
End For
Victim-item=min(Impact-Factor[])
Step 4: Modify(victim,$T_n$,D);
    Modify-rules(R);
    Modify-rules($R_s$);
Until($R_s$ is Empty)
```

6 ILLUSTRATIVE EXAMPLE

Consider the data set shown in Table 1. The minimum support threshold, MST=10 and minimum confidence threshold, MCT=70%. Let the set of sensitive association rules to be hidden $R_s = \{5 \Rightarrow 1, 2; 1, 10 \Rightarrow 2, 5; 10 \Rightarrow 1, 2\}$. We apply HDSRIF algorithm to hide R_s and to release a sanitized dataset.

Tid	List of items	Tid	List of items
1	1 5 10	11	1 2 4 5 6 10
2	1 3 4 8 9 10	12	1 2 5 8
3	1 2 3 4 5 6 10	13	1 4
4	1 2 3 5 6 8 9 10	14	2 3 5 6 9 10
5	1 2 3 5 10	15	1 2 5 10
6	1 2 4 5 8 10	16	1 2 4 4 6 8 9 10
7	1 3 5 10	17	1 2 4 5 8 9 10
8	1 2 4 5 6 10	18	2 3 4 5 6 9 10
9	1 5 9	19	1 2 3 4 8 9 10
10	1 2 3 5 6 9 10	20	1 2 4 5 8 9 10

Table 1. Dataset

Step 1: Selection of a Sensitive Rule from R_s. Frequencies of the items in the rules of R_s are shown in Table 2:

S.No	Item	Frequency
1	1	3
2	2	3
3	5	2
4	10	2

Table 2. Frequencies of the items in R_s :

The items with highest frequency are 1 and 2. The rule which is having highest frequency items and shortest length was $\{5 \Rightarrow 1, 2\}$

Step 2: Recognizing the Essential Number of Transactions

This step aims at identifying the required number of transactions for modification. For this first all the transactions of the dataset are to be sorted in descending order of their SIF. Sensitive Item Frequency (SIF) of a transaction is calculated as number of sensitive items divided with length of the transaction. The sorted order of all the transactions based on their SIF are shown in Table 3. The number of transactions that are required for hiding the rule 5⇒1, 2 are $T_n = min\{\gamma(5, 1, 2) - 10 + 1, \gamma(5, 1, 2) - \lceil\gamma(5) * 0.7\rceil + 1\}$.

=min$\{12 - 10 + 1, 12 - \lceil 17 * 0.7\rceil + 1\}$

$= min\{3, 1\}$ =1

So only one transaction modification is sufficient to hide the rule $5 \Rightarrow 1, 2$. Select the transaction from Table 3 which supports $\{5 \Rightarrow 1, 2\}$ and with highest SIF value. So the selected transaction is with TID 15. In this process we can remove either item 1 or item 2 to hide the rule.

Step 3: Selecting Victim Item

The victim item was selected based on the concept of Impact-Factor. So we calculate the Impact-Factor of items 1 and 2. The item which is having minimum Impact-Factor will be selected as victim item. To calculate the Impact-Factor of an item 1, consider the rules from R which are having item 1 as one of the item either on the left hand side or right hand side of the rule. Update the support of those rules by considering the selected transaction in step 2. Calculate the updated confidence based on the updated support values. Then count the number of rules which are having support less than the minimum support threshold or confidence less than the minimum confidence threshold. Store the count as Impact-Factor of the item 1. In the similar manner calculate the Impact-Factor of item 2 also. The Impact-Factor(1) is 5 and Impact-Factor(2) is 6 . The calculations of the Impact-Factor are shown in Table 4. Item 1 is having minimum Impact-Factor. So the selected victim item is item 1.

Step 4: Updating the transactional data set and Updating the set of association rules R and sensitive rules R_s

Update the data set by removing item 1 from

Tid	List of Items	SIF	Tid	List of Items	SIF
1	1 5 10	1	10	1 2 3 5 6 9 10	0.5714
15	1 2 5 10	1	17	1 2 4 5 8 9 10	0.5714
5	1 2 3 5 10	0.8	20	1 2 4 5 8 9 10	0.5714
7	1 3 5 10	0.75	4	1 2 3 5 6 8 9 10	0.5
12	1 2 5 8	0.75	13	1 4	0.5
6	1 2 4 5 8 10	0.6667	14	2 3 5 6 9 10	0.5
8	1 2 4 5 6 10	0.6667	16	1 2 4 5 6 8 9 10	0.5
9	1 5 9	0.6667	18	2 3 4 5 6 9 10	0.4286
11	1 2 4 5 6 10	0.6667	19	1 2 3 4 8 9 10	0.4286
3	1 2 3 4 5 6 10	0.57	2	1 3 4 8 9 10	0.3333

Table 3. Sorted order of dataset based on SIF

transaction with TID 15. The modified dataset shown in Table 5.

Tid	List of items	Tid	List of items
1	1 5 10	11	1 2 4 5 6 10
2	1 3 4 8 9 10	12	1 2 5 8
3	1 2 3 4 5 6 10	13	1 4
4	1 2 3 5 6 8 9 10	14	2 3 5 6 9 10
5	1 2 3 5 10	15	2 5 10
6	1 2 4 5 8 10	16	1 2 4 5 6 8 9 10
7	1 3 5 10	17	1 2 4 5 8 9 10
8	1 2 4 5 6 10	18	2 3 4 5 6 9 10
9	1 5 9	19	1 2 3 4 8 9 10
10	1 2 3 5 6 9 10	20	1 2 4 5 8 9 10

Table 5. Modified Dataset.

Along with this,update the sensitive association rules and complete set of association rules also. In updating sensitive association rules, along with rule $\{5 \Rightarrow 1, 2\}$, another sensitive rule $\{10 \Rightarrow 1, 2\}$ also be hidden so now R_s contains only one rule i.e $\{1, 10 \Rightarrow 2, 5\}$ with support 10. By repeating step 1 to step 3 again, we will get T_n as 1 i.e only one transaction is required for modification to hide the rule, the selected transaction was TID 5, Impact-Factor(2)=10 and Impact-Factor(5)= 13. So the victim item was item 2 which is to be removed from transaction with TID 5. By this modification the rule $\{1, 10 \Rightarrow 2, 5\}$ will be hidden and R_s becomes empty. The final sanitized data set was released which was shown in Table 6.

For 1:		Impact-Factor(1) =5			For 2:		Impact-Factor (2) = 6		
LHS	RHS	Support	Modified Support	Modified Confi	LHS	RHS	Support	Modified Support	Modified Confi
2	1	13	12	0.8	2	1	13	12	0.8571
1	2	13	12	0.7058	1	2	13	12	0.6666
2,5	1	12	11	0.7857	2,5	1	12	11	0.8461
1,5	2	12	11	0.7857	1,5	2	12	11	0.7333
1,2	5	12	11	0.9166	1,2	5	12	11	0.7857
5	1,2	12	11	0.6470	5	1,2	12	11	0.6470
2	1,5	12	11	0.7333	2	1,5	12	11	0.7857
2,5,10	1	11	10	0.7692	2,5,10	1	11	10	0.8333
1,5,10	2	11	10	0.8333	1,5,10	2	11	10	0.7692
1,2,10	5	11	10	0.9090	1,2,10	5	11	10	0.9090
1,2,5	10	11	10	0.9090	1,2,5	10	11	10	0.9090
5,10	1,2	11	10	0.6666	5,10	1,2	11	10	0.6666
2,10	1,5	11	10	0.7142	2,10	1,5	11	10	0.7692
2,5	1,10	11	10	0.7142	2,5	1,10	11	10	07692
1,10	2,5	11	10	0.7142	1,10	2,5	11	10	0.6666
1,5	2,10	11	10	0.7142	1,5	2,10	11	10	0.6666
1,2	5,10	11	10	0.8333	1,2	5,10	11	10	0.8333
2	1,5,10	11	10	0.6666	2	1,5,10	11	10	0.7692
2,10	1	12	11	0.7857	2,10	1	12	11	0.8461
1,10	2	12	11	0.7857	1,10	2	12	11	0.7333
1,2	10	12	11	0.9166	1,2	10	12	11	0.9166
10	1,2	12	11	0.6470	10	1,2	12	11	0.6470
2	1,10	12	11	0.7333	2	1,10	12	11	0.7857
4	1	10	9	0.8181	5	2	14	13	0.7647
5	1	15	14	0.8235	2	5	14	13	0.9285
1	5	15	14	0.8235	5,10	2	13	12	0.8
5,10	1	13	12	0.8	2,10	5	13	12	0.9230
1,10	5	13	12	0.8571	2,5	10	13	12	0.9230
1,5	10	13	12	0.8571	10	2,5	13	12	0.7058
10	1,5	13	12	0.7058	5	2,10	13	12	0.7058
5	1,10	13	12	0.7058	2	5,10	13	12	0.8571
1	5,10	13	12	0.7058	10	2	14	13	0.7647
10	1	15	14	0.8235	2	10	14	13	0.9285
1	10	15	14	0.8235					

Table 4. Calculating th mpact-Factors in step 3.

Tid	List of items	Tid	List of items
1	1 5 10	11	1 2 4 5 6 10
2	1 3 4 8 9 10	12	1 2 5 8
3	1 2 3 4 5 6 10	13	1 4
4	1 2 3 5 6 8 9 10	14	2 3 5 6 9 10
5	1 3 5 10	15	2 5 10
6	1 2 4 5 8 10	16	1 2 4 5 6 8 9 10
7	1 3 5 10	17	1 2 4 5 8 9 10
8	1 2 4 5 6 10	18	2 3 4 5 6 9 10
9	1 5 9	19	1 2 3 4 8 9 10
10	1 2 3 5 6 9 10	20	1 2 4 5 8 9 10

Table 6. Sanitized Dataset D^1.

7 Performance Measures

7.1 Hiding Failure:(HF)

When some sensitive association rules that cannot be hidden by sanitization process , we call this problem as Hiding Failure, and is measured in terms of the percentage of sensitive association rules that are discovered from sanitized database D^1. The hiding failure is calculated as follows $HF = \frac{\sharp R_S(D^1)}{\sharp R_S(D)}$ where$\sharp R_S(D^1)$ denotes the number of sensitive association rules discovered from sanitized database D^1, and $\sharp R_S(D)$ denotes the number of sensitive association rules discovered from original database D.

7.2 Misses Cost/Lost rules:(MC)

Misses Cost are some non-sensitive association rules that can be discovered from original database but cannot be mined from the sanitized database D^1. This happens when some non-sensitive association rules loose support or confidence below the minimum threshold values in the database due to the sanitization process. We call this problem as Misses Cost, and is measured in terms of the percentage of non-sensitive association rules that are not discovered from sanitized database D^1. The misses cost is calculated as $MC = \frac{\sharp\sim R_S(D)-\sharp\sim R_S(D^1)}{\sharp\sim R_S(D)}$ where$\sharp \sim R_S(D)$ denotes the number of non-sensitive association rules discovered from original database D, and $\sharp \sim R_S(D^1)$ denotes the number of non-sensitive association rules discovered from sanitized databaseD^1.

7.3 Ghost rules/False rules/Artifactual Patterns:(GR)

Ghost rules occurs when some artificial association rules are generated from D^1 as a product of the sanitization process. We call this problem as ghost rules, and is measured in terms of percentage of the discovered association rules that are ghost rules. This is measured as $GR = \frac{|R^1|-|R\cap R^1|}{|R^1|}$ where $|R|$ and $|R^1|$ represent respectively the set of association rules that can be generated from D and D^1.

7.4 Difference between the original and sanitized datasets(Diff(D,D^1))

We could measure the dissimilarity between original and sanitized database by simply comparing their histograms.

$Diff(D, D^1) = \frac{1}{\sum_{i=1}^{n} fD(i)} \sum_{i=1}^{n} [fD(i) - fD^1(i)]$ where fx(i) represents the frequency of the i^{th} item in the dataset x,and n is the number of distinct items in the original dataset.

8 Experiment and Evaluation

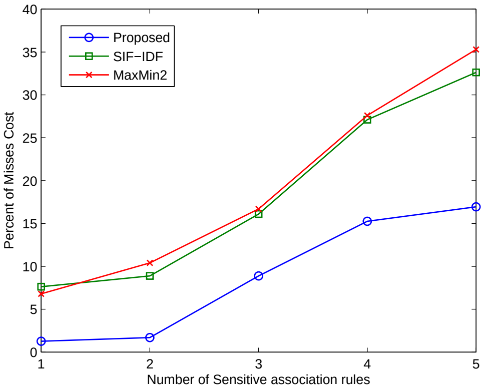

Fig. 2. Comparison of Misses cost .

The dataset for our testing has been posted in IEEE ICDM03 as the file name Retail.dat and has been available in On-line at http://mi.ua.ac.be/data/. The dataset comprises 88,162 transactions and 16,469 product IDs.

In this experimental evaluation, we compared our HDSRIF algorithm with the SIF-IDF algorithm presented in [12] and MaxMin2 algorithm presented in [9] to evaluate the side effects and computational complexity. The MaxMin2 algorithm was developed based on border approach and increases the efficiency in minimizing the side effects compared with the previous heuristic approaches. The SIF-IDF algorithm uses intersection lattice on frequent itemsets to decrease the side effects when compared with MaxMin2 algorithm. We compare the performance of these algorithms based on four metrics, including misses cost, artifacts, hiding failure and accuracy of the sanitized dataset.

Fig. 2 demonstrate that the effectiveness of the proposed algorithm in the misses cost minimization. The experimental evaluation specifies that when the number of sensitive association rules are increased, HDSRIF caused minimum number of lost rules than SIF-IDF and MaxMin2. In particular, MaxMin2 affected extremely high proportion non-sensitive rules.

Fig. 3 shows the performance of the proposed algorithm in hiding failure. The calculation specifies that the proposed algorithm hides all the sensitive association rules given by the data owner.

Fig. 4 illustrates the comparison of the proposed algorithm in artifacts (ghost rules) reduction. The evaluation specifies that even though the number of sensitive association rules are increased, HDSRIF caused no artifacts.

Fig. 5 indicates that the HDSRIF algorithm required reduced number of alterations than SIF-IDF and MaxMin2 algorithms. High accuracy (99%) means that the released sanitized dataset was slightly distorted.

Essentially, the evaluations shows that the proposed algorithm HDSRIF yields good results when compared to SIF-IDF and MaxMin2 in minimizing the side effects and data distortions.

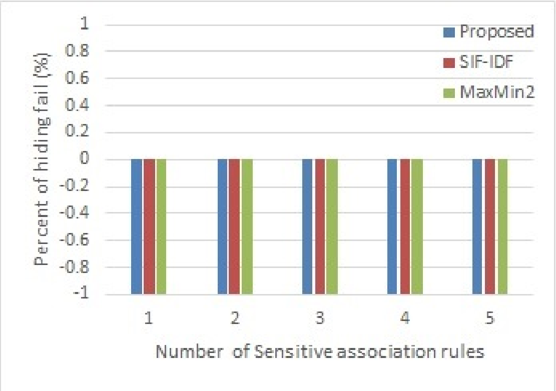

Fig. 3. Comparison of Hiding Failure.

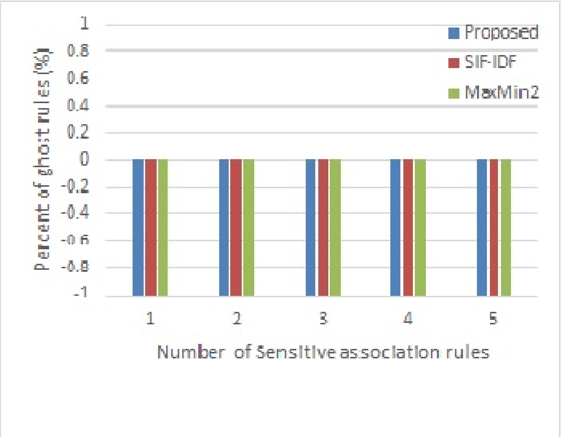

Fig. 4. Comparison of ghost rules.

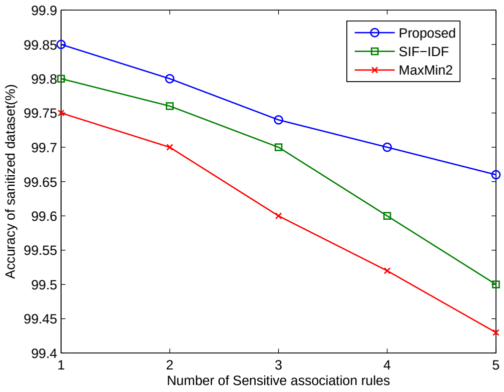

Fig. 5. Accuracy of sanitized dataset.

9 Conclusion

An amassed number of organizations under-way to share their transactional databases for their common aids. To associate the partnership with the other organizations for sharing databases, the organizations may possibly want to make balance among hiding sensitive association rules and enlightening non-sensitive association rules. Data sanitization has developed as a practice to encourage the sharing of data among the organizations while easing the concerns of specific members by preserving confidentiality of their sensitive knowledge in the form of sensitive association rules. This process is guided by the need to minimize the effect on the data effectiveness of the sanitized database by permitting mining non-sensitive knowledge in the form of non-sensitive association rules from the sanitized database. The problem of data sanitization is known to be very complex. In this paper we proposed a novel heuristic method for sanitization based on itemset lattice. The item called as victim item that is to be modified to hide a rule is selected from the consequent of the rule by minimizing the impact on the non-sensitive association rules. An empirical comparison study using a real dataset was been conducted. Results of the study shows that the proposed approach outperform the existing algorithms (MaxMin2, SIF-IDF) in terms of maximizing dataset accuracy, which is the primary objective of data sanitization, at the cost, unfortunately, of computational efficiency or speed.

References

[1] Ali Amiri. Dare to share: Protecting sensitive knowledge with data sanitization. *Decision Support Systems*, 43, 2007.

[2] E. Dasseni, V.S.Verykios, A.K. Elmagarmid, and E. Bertino. Hiding association rules by using confidence and support. In *In Proceedings of the 4th international workshop on Information Hiding*, pages 369–383, 2001.

[3] A. Elmagarmid, M. Ibrahim, M. Atallah, E. Bertino, and V. S. Verykios. Disclosure limitation of sensitive rules. In *In Proceedings of the 1999 IEEE Knowledge*, page 4552, 1999.

[4] G.Gratzer. Lattice theor. *Mathematics Subject Classification*, 2011.

[5] Aris Gkoulalas-Divanis and Vassilios S. Verykios. Association rule hiding for data mining. In *Advances IN DATABASE SYSTEMS, Springer New York*, volume 5012, pages 99–103, 2010.

[6] Oliveira Stanley R. M. and Osmar R. Privacy preserving frequent itemset mining. In *Proceedings of the IEEE international conference on Privacy, security and data mining*, pages 43–54, Darlinghurst, Australia, Australia, 2002. Australian Computer Society, Inc.

[7] Lindell; Pinkas. Non-perturbative masking. *In Encyclopedia of database systems US*, 2009.

[8] D. Ramakrishnan, R. LeFevre, and K. DeWitt. Efficient full-domain k-anonymity. *In SIGMOD*, 2005.

[9] Hong Tzung-Pei; Lin Chun-Wei; Yang Kuo-Tung; Wang Shyue-Liang. Using tf-idf to hide sensitive itemsets. *Applied Intelligence*, 38, 6 2013.

[10] Oliveira S.R.M. and Zaiane O.R. Algorithms for balancing privacy and knowl-

edge discovery in association rule mining. In *Database Engineering and Applications Symposium, 2003. Proceedings. Seventh International*, pages 54–63, 2003.

[11] A. A. Tsitsonis, E. D. Pontikakis, and V. S. Verykios. An experimental study of distortion based techniques for association rule hiding. In *In Proceedings of the 18th Conference on Database Security*, page 325339, 2004.

[12] George V. Moustakides; Vassilios S. Verykios. A maxmin approach for hiding frequent itemsets. *Data and Knowledge Engineering*, 65, 2008.

[13] V. S. Verykios, Y. Saygin, and A. K. Elmagarmid. Privacy preserving association rule mining. In *In Proceedings of the 2002 International Workshop on Research Issues in Data Engineering: Engineering ECommerce EBusiness Systems*, pages 151–163, 2002.

[14] Shyue Liang Wang and Jafari A. Using unknowns for hiding sensitive predictive association rules. In *Systems, Man and Cybernetics, 2005 IEEE International Conference*, volume 1, pages 164–169, 2005.