Decision Tree Based Data Reconstruction for Privacy Preserving Classification Rule Mining

G. Kalyani
Research Scholar, Acharya Nagarjuna University, India
E-mail: kalyanichandrak@gmail.com

M.V.P. Chandra Sekhara Rao
Professor, Dept of CSE, RVR & JC College of Engineering, Guntur, India

B. Janakiramaiah
Professor, Dept of CSE, DVR & Dr.HS MIC College of Technology, Vijayawada, India

Keywords: privacy, sensitive patterns, data reconstruction, decision tree, classification rules

Received: June 8, 2017

Data sharing among the organizations is a general activity in several areas like business promotion and marketing. Useful and interesting patterns can be identified with data collaboration. But, some of the sensitive patterns that are supposed to be kept private may be disclosed and such disclosure of sensitive patterns may effects the profits of the organizations that own the data. Hence the rules which are sensitive must be concealed prior to sharing the data. Concealing of sensitive patterns can be handled by modifying or reconstructing the database before sharing with others. However, to make the reconstructed database usable for data analysts the utility or usability of the database is to be maximized. Hence, both privacy and usability are to be balanced. A novel method is proposed to conceal the classification rules which are sensitive by reconstructing a new database. Initially, classification rules identified from the database are made accessible to the owner of the data to spot out the sensitive rules that are to be concealed. In the next, from the non-sensitive rules of the database, a decision tree will be constructed based on the classifying capability of the rules, from which a new database will be reconstructed. Finally, the released reconstructed database to the analysts reveals only non-sensitive classification rules. Empirical studies proved that the proposed algorithm preserves the privacy effectively. In addition to that utility of the classification model on the reconstructed database was also be preserved.

Povzetek: Predstavljena je metoda strojnega učenja, ki skrbi za privatnost podatkov.

1 Introduction

Significant improvements in data storage have led to rise in inexpensive data storage techniques for databases. Improvements in storing and analyzing enormous amounts of data present a challenge to people and organizations for transforming this data into valuable knowledge. Data mining, which involves extorting the patterns that are novel and valuable from mass repositories of data, is efficient in transforming the data into knowledge.

Various data mining algorithms are in usage for mining interesting patterns from the collected data. Patterns like classification rules, association rules and clusters can be discovered with mining techniques. On the other side, in order to get the mutual benefits data will be shared among the collaborated organizations. But, some sensitive information or patterns may exist with in the data which is to be maintained as private, since the revelation of sensitive information or pattern may affect the business deals of the data owner and violates the privacy issues of the data owner as an end user. Hence, along with the need of sharing and collaborative mining, the importance of protecting the information or patterns against disclosure is one of the most important point in the security issues of data mining [1, 2]. To preserve the sensitive information or patterns from unwanted disclosure, privacy preserving data mining (PPDM) has emerged as a security area in data mining and database field [1, 12].

1.1 Classification of approaches in PPDM

PPDM is an interesting research area in the data mining community. It concentrates on the privacy issues of individuals or organizations which are violated due to the disclosure of sensitive information or patterns. PPDM converts the original database into a transformed database in such way that no sensitive data or pattern can be mined from the transformed database. Various methodologies exists in the literature, for this transformation to protect sensitive information or knowledge. A taxonomy for the PPDM techniques based on a set of parameters is discussed and the taxonomy is shown in Figure 1.

Based on the parameter whether the data owner requires privacy for the data or knowledge, PPDM techniques were classified as:

- **Data Hiding Techniques (Protecting Sensitive Data)**
  Data hiding approaches[3, 6, 11] investigate about maintaining the privacy of data or information before applying the data mining techniques on the database. These approaches concentrate on the exclusion of private information from the database before sharing the data with others. Perturbation, sampling, suppression, transformation[17], etc. are the general techniques used to create a transformed database. The final aim of data hiding is, after sharing the transformed database receiver has to get valid data mining results without disclosing the private data of the data owner.

- **Knowledge Hiding Techniques (Protecting Sensitive Knowledge)**
  Knowledge hiding approaches [4, 13] investigate on the protection of sensitive knowledge inferred from the data(instead of the data), by applying the mining tools on the original database. The ultimate goal of knowledge hiding techniques is no sensitive knowledge is to be mined by applying the data mining techniques on the transformed database. Knowledge hiding approaches mainly deals with the following techniques.

  - **Data Distortion Technique**: This technique tries to protect the knowledge by changing the parameters associated with the sensitive knowledge. These techniques works by altering 0s to 1s or vice versa in the specified transactions of the database, which may generate unwanted side effects in the new database[16].
  - **Data Blocking Technique**: In this technique, 0's and 1's related to the data of the sensitive knowledge will be replaced by "?"(Unknown) in selected transactions instead of doing insertion and deletion of items [15].
  - **Reconstruction Based Technique**: This technique reconstructs a database from the sanitized knowledge, extracted from the original database. When compared to the heuristic methods side effects will be reduced in reconstructed database [8, 19, 21].

The paper concentrates on protecting the sensitive knowledge by reconstructing the database from the non-sensitive knowledge mined from the original database i.e. knowledge hiding based on reconstruction based technique.

1.2 Problem motivation

In business organizations, classification techniques reveals a set of classification rules.Among the rules mined, some are crucial for decision making and there by to increase their profits. In order to get some mutual benefits, organizations share their data with others also. By getting their data, others also can identify all the classification rules. In some cases the person who owns the data does not want to reveal some of the rules to others even though the data was shared with them. The set of rules which are crucial and important for gaining the profits must be kept confidential i.e. they ,must not be revealed to others even they have applied classification techniques on the shared data. The set of rules which are to be hidden from disclosure to others are called as sensitive classification rules.

The focus of this paper is on the privacy of classification rules mined from the databases. The need of privacy in classification rule mining was explored with an example scenario [21]. A credit card company agreed to share their credit card approvals to a new home loan company. When people have applied for the credit card, their data will be maintained as a separate record in the database of Credit Card Company. The attributes financial status, experience, gender, salary, age and address are maintained for every person. The class label is maintained as the approval result of their credit card application. After getting the data from the credit card company, the home loan company constructs a classification model to categorize the applicants of home loan. Based on the classification model and predicted results, the home loan company can decide the approval of the home loan to the applicants. The home loan company gets benefited by avoiding the approvals to the wrong applicants based on the data taken from the credit card company. The home loan company can also make use of the credit card company database in another manner to improve their business. By changing the class label to the address attribute, the home loan company can identify the appropriate group or individual customers to send advertising mails about their offers. Hence, to avoid such type of advertising to their customers, the credit card company should modify their database before sharing with the home loan company in such a way that classification rules which are useful for identifying a group of valued customers must not be revealed to the home loan company. The above scenario clearly indicates the need of preserving the sensitive classification rules before sharing the data with the others.

2 Literature review

In the perspective of privacy in classification rule mining, the major part of the work in research concentrates on the privacy of individual data. In [5], privacy of individual data can be achieved by data reduction. In the data reduction method, the effect of non-sensitive knowledge on the sensitive knowledge was analyzed. For preserving the privacy of individual data, a decision tree can be constructed by col-

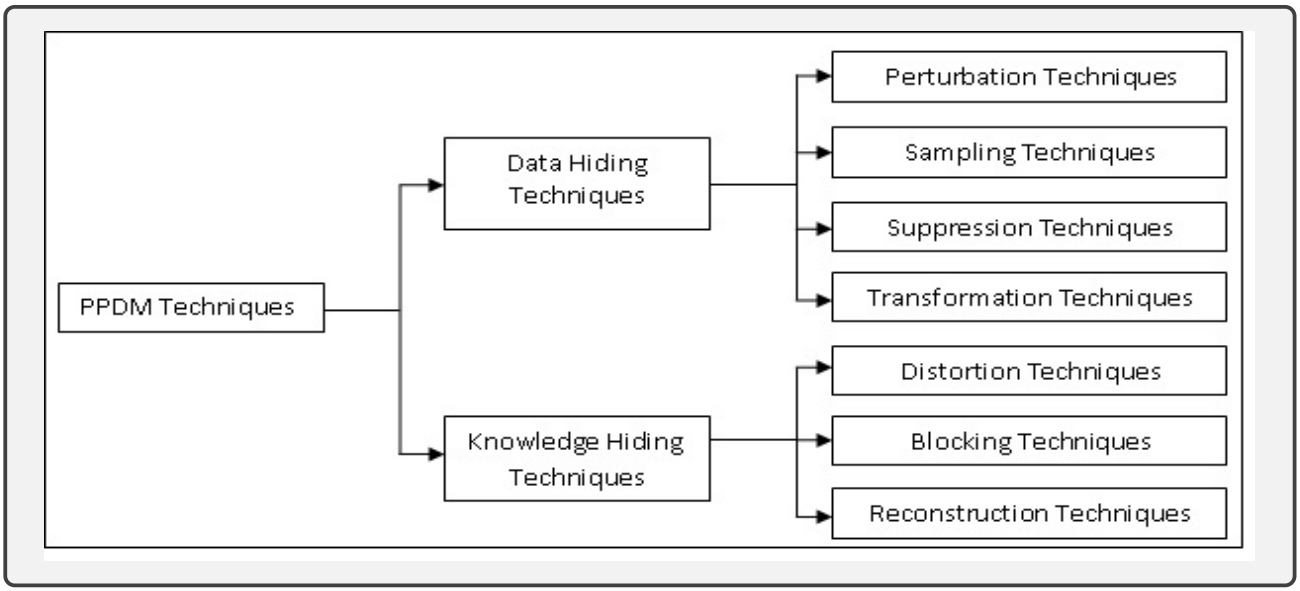


Figure 1: The Taxonomy of the PPDM Techniques.

lecting data from multiple parties without revealing others to their data was proposed in [7, 9].

In [18], the authors projected a classification rule hiding method based on reconstruction of categorical datasets. The methodology modifies the tuples of the original database which contains the values related to both sensitive and non-sensitive classification rules and then makes use of the tuples related to the non-sensitive rules to create its transformed database.

This paper [3] projected a novel method to defend the sensitive classification rules based on the reconstruction process for categorical datasets. Initially, the owner of the data will identify a set of sensitive rules that are to be concealed among the rules revealed from the original database. Later, the set of non-sensitive rules along with the characteristics extracted from the database are used to construct a decision tree. Finally, the new database is reconstructed which reveals only non-sensitive classification rules.

In [20], the authors proposed a template-based technique to protect against the threats caused by data mining functionality. The technique focuses on two points: preserving the privacy of knowledge and increase the usefulness of non-sensitive knowledge that can be derived from the data. Sensitive rules are indicated by a set of "privacy templates". Template includes the sensitive information which is to be concealed, a set of corresponding attributes, and the relationship between the two. Authors proved that suppressing the attribute values is an efficient approach to protect sensitive rules. For a large dataset, identifying an optimal possibility for suppression may be hard, because it needs to do optimization over all suppression's.

In [14], Verykios et al. projected a method for hiding the classification rules which are considered as private. Hiding is achieved before publishing the data on the web through data perturbation approach in categorical databases. The method used the characteristics of sequential covering classification algorithms. Modification will be done to the tuples of sensitive rules in such a way that the alterations are spread to the tuples of the significant non-sensitive rules. The spreading will be proportional to the rank in the rule set. So that, the method guarantees that the sensitive rules are hidden and maintains the current structure of the rule set, thereby the usefulness of the new database is maximized. Authors have proposed another distribution method with a modification to the basic method. Authors have proved that both the methods are effective in terms of privacy and usefulness of the new database.

3 Proposed method

3.1 Problem statement

Consider a database (D) consists of n tuples comprises of m dimensions along with associated labels known as class with number of distinct classes as C. By applying a classification rule mining algorithm on D, number of classification rules (CR) can be discovered. Given a set of classification rules among CR which are treated as sensitive classification rules (SCR $\subset$ CR) by domain expert (the data owner), the process of classification rule hiding is to appropriately reconstruct a database with the intention of mining the reconstructed database (D^1) by using any classification rule mining algorithms, reveals all the non-sensitive classification rules (NSCR=CR - SCR) that are revealed from the original database, whereas all the SCR are shielded from revelation and new rules (originally non-existent rules) cannot be mined.

3.2 Framework

The framework shown in Figure 2 addresses the problem statement. From the original database, a number of classification rules are discovered by applying any classification algorithm, which are useful to the data owner for forecasting purpose. The data owner or domain expert identifies the sensitive classification rules which must be preserved from revelation when classification rule mining algorithms are applied on the database before sharing the data with the others. The proposed method for classification rule hiding reconstructs a sanitized database by considering the original database, set of classification rules generated and a set of identified sensitive classification rules as input. By applying the classification rule mining algorithm on the reconstructed database, only non-sensitive classification rules which are discovered from the original database, are only be discovered and all the sensitive rules will be hidden from disclosure.

3.3 RCRH (Reconstruction based Classification Rule Hiding)Method

The proposed algorithm for classification rule hiding was reconstruction based algorithm, i.e. the transformed database will be reconstructed from the set of NSCR. The required input for the classification rule hiding is, the database D, classification rules CR mined from D and a set of sensitive classification rules SCR among CR which were decided by data owner depending on to whom they wish to share the database. The result of the algorithm is a reconstructed database D^1.

The proposed algorithm first eliminates the SCR from CR which are the possible classification rules from D (step 2 to 4 of Algorithm 1). Then for every rule in CR, calculate a measure called as capability of the rule. The Capability of the rule indicates the number of the tuples that are correctly classified by that rule (step 5 to 6 of Algorithm 1). The process of calculating the capability for a rule was shown in Algorithm 2. Then arrange the rules in the decreasing order of their capability values because high capability indicates the maximum ability of classifying the data in the database D. Now consider the rules in order and construct a decision tree with the non-sensitive classification rules only.

The construction of the decision tree will be as follows: Consider the rules in decreasing order of their capability values. Calculate the information gain of all the attributes of the database with respect to D. Information gain of an attribute is the measure of the difference in entropy before and after the tuples are divided into groups based on that attribute (step 7 to 9 of Algorithm 1). The information gain of an attribute is calculated as: Gain(A)= Entropy(D)- Entropy(D,A). Entropy(D) and Entropy(D,A) can be calculated by using the equations (1) and (2). The process of calculating the info-gain of an attribute was shown in Algorithm 4.

$$E\left(D\right) = \sum_{i=1}^{c} -P_i \log_2 P_i \tag{1}$$

Where D is the database, c is number of distinct class labels, P_i is the probability of the i^{th} class label.

$$E\left(D, A\right) = \sum_{V \in A} P\left(V\right) * E\left(V\right) \tag{2}$$

Where D is a Database, A is an attribute for which entropy is calculated, V is value of an attribute, P (V) probability of value V, E (V) is entropy of value V.

Consider the rule in CR in the decreasing order of capability values. The attributes of that rule are considered in decreasing order of their info-gain values. By considering the attributes in the order of info-gain, construct a path in the decision tree with the attribute having the highest info-gain at the root node. The possible values of that attributes in database D are considered as possible branches from that node. The path will be extended in the similar manner by considering all the attributes in considered rule. The capability of a rule will be considered as a measure for the branch created in the decision tree. The class label of that rule is given as a leaf node in the branch. For the next rules, based on the order of the attributes path will be checked in the decision tree. If the path matches with the existing path it continues and whenever the match fails, the new path will be constructed from that point. The same process will be repeated to all the non-sensitive rules of D (step 10 to 16 of Algorithm 1).

After the decision tree has constructed, then the transformed database will be reconstructed from the decision tree. The process of reconstructing the database will be applied to all the paths of the decision tree by considering only one path at a time. Hence, consider a single path in the decision tree. A path in the decision tree is associated with capability which indicates the influence of that rule on the database D. Insert number of tuples in the transformed database D^1 equal to the capability of that path in the decision tree.

The path in the decision tree may not contain all the attributes of the database D. Hence, if tuples are added in the database for a path in the decision tree, the tuples in the constructed database may contain some missing values related to the attributes which were not existed in the path of the decision tree (step 17 to 23 of Algorithm 1).

The missing values in the reconstructed database are to be filled by using methods to fill the missing values efficiently. The process of filling the missing values is shown in Algorithm 5. Consider all the attributes of the D^1 as TA (step 3 of Algorithm 5). Select an attributes of the D^1 which are having not null values, i.e. the set of the attributes which are having some data values as SA (step 4 of Algorithm 5). Identify the combination of the distinct values in the set of attributes SA, as a string which is indicated by C (step 5 of Algorithm 5). Scan the database D to retrieve the set of tuples which matches

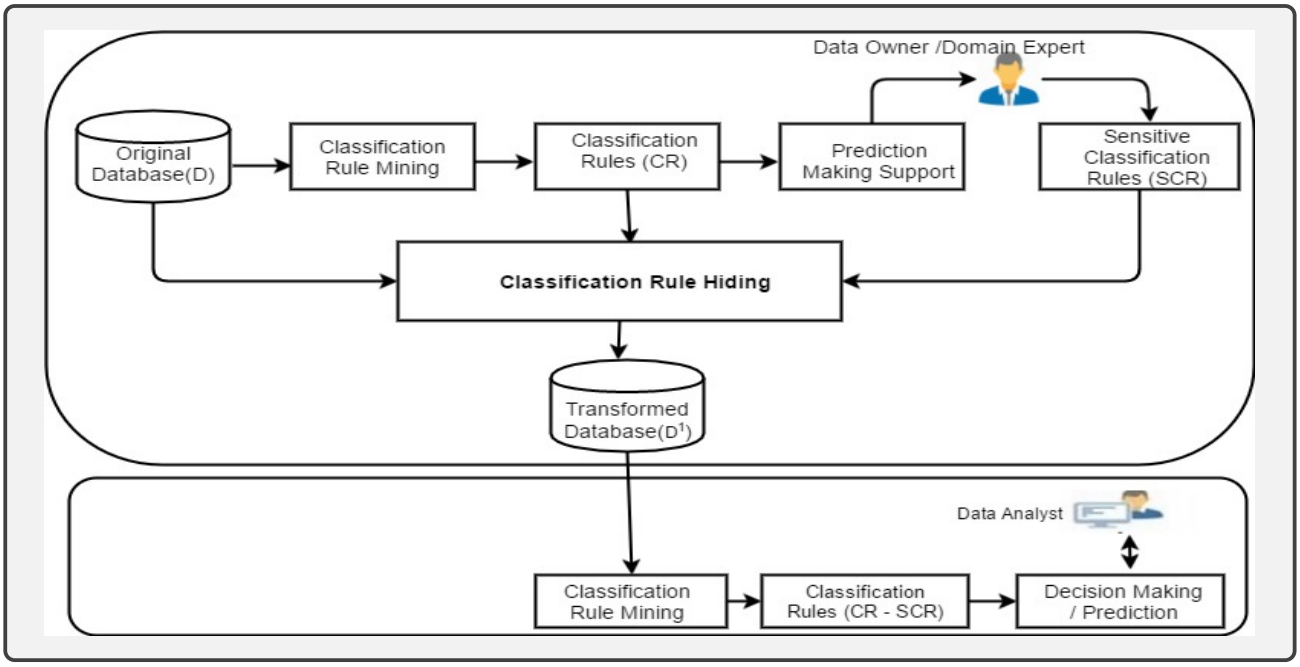


Figure 2: The Proposed Framework for Classification Rule Hiding.

Algorithm 1: RCRH(**R**econstruction based **C**lassification **R**ule **H**iding).

Data: Original Database D, Classification rules CR, Sensitive Classification Rules SCR.
Result: Transformed Database D^1

begin
  for *every rule* $r \in CR$ **do**
    if $r \in SCR$ **then**
      $CR = \{CR - r\}$; /* discard the sensitive rules */
  for *every rule i = 1 to* $|CR|$ **do**
    capb_i = Capability(R) ; /* calculating the classifying ability of the rule */
  Ent_D = Entropy (D); /* entropy of database D */
  for *every attribute* $A \in D$ **do**
    Ig_A = Info-gain(A,D); /* calculating the gain of attributes in D */
  while $(|CR| > 0)$ **do**
    RL=$\{r/r, \forall k \in CR, capb\_r \geq capb\_k\}$; /* select the rule with max capability */
    while *(RL is not empty)* **do**
      attri=$\{x/x \in i \ and \ \forall y \in i, Ig\_x \geq Ig\_y\}$;
      create *attri* as non-terminal node of DT; /* creating a path in the tree */
      Discard *attri* from *RL* ;
    Assign class label of *RL* as terminal node; /* adding of terminal node */
  for *every path* $P \in DT$ **do**
    count=0 ;
    repeat
      Generate a tuple in D^1 with the attributes in P; /* adding of tuples in D^1 */
      count++;
    until *(count==capb_P))*;
    Fill_Missing_Values(D^1) /* to fill the missing values in D^1 */
  Return D^1;

Algorithm 2: Function Capability(R).

Data: Original Database D,Classification rule R.
Result: Capability of Rule R.

```
1 begin
2     Count=0;
3     for each tuple T ∈ D do
4         if T ∈ R then
                                          /* if tuple is classified by rule R */
5             Count++;
6     Return Count;
```

Algorithm 3: Function Entropy(D).

Data: Original Database D, Number of distinct class labels C.
Result: Entropy of D.

```
1 begin
2     Ent_D = 0;
3     for i= 1 to C do
                                  /* getting no.of tuples with i^th class label */
4         Tc = Select count(*) from D where class=C_i;
5         L = log(Tc/|D|);
6         Ent_D = Ent_D + (Tc/|D|) * L;
7     Return ( - Ent_D);
```

Algorithm 4: Function Info_gain(A,D).

Data: Database D, No.of distinct values V in A, Ent_D, No.of distinct class labels C.
Result: Information gain of A

```
1 begin
2     Ent_A = 0 ;
3     for i= 1 to V do
4         Tv = select * from D where A=V_i;        /* getting tuples with i^th value of A */
5         E_Tv = 0;
6         for j = 1 to C do
                                  /* getting the no.of tuples with j^th class label */
7             Tvc = select count(*) from Tv where class=C_j;
8             L = log(Tvc/|Tv|) * L;
9             E_Tv = E_Tv + (Tvc/|Tv|) * L;
10        Ent_A = Ent_A + (|Tv|/|D|) * (−E_Tv);          /* calculating entropy of A */
11    Ig_A = (Ent_D) - (Ent_A);                          /* Gain of attribute A */
12    Return Ig_A;
```

Algorithm 5: Fill_Missing_Values(D^1).

Data: Original Database D,Reconstructed Database D^1.
Result: Reconstructed Database D^1

```
1  begin
2      repeat
3          TA[ ]= attributes in D^1 ;
                                          /* get all the attributes of D^1 */
4          SA[ ] = attributes which are not empty in D^1 ;
5          Let C be the combination of values in the attributes of SA ;
6          for every tuple t ∈ D do
7              if (values of SA[] in t == C) then
                                      /* values of the SA[]attributes in tuple */
8                  temp = temp ∪ t
                                          /* add the tuple to temp buffer */
9          for each tuple t ∈ temp do
10             Count the occurrences of each distinct value in the attributes other than SA[] ;
11         Select the attribute A in which more number of occurrence are related to the same distinct value V ;
12         Insert value 'V' in attribute 'A' in D^1;
13     until (SA[ ]==TA[ ]);
```

to the combination C in the set of the attributes SA. Let the retrieved tuples be in the buffer temp (step 6 to 8 of Algorithm 5). By scanning the tuples in the temp buffer, count the number of occurrences of each distinct value of the attribute which does not belong to the set SA (step 9 to 10 of Algorithm 5). Then, select the attribute which has the major importance i.e. occurrence of a particular value in the attribute is more than the other values (step 11 of Algorithm 5). The selected value is filled with the value which has the maximum number of occurrences (step 12 of Algorithm 5). Repeat the process of filling the missing values by considering the new set of selected attributes SA, which are filled with the values until the selected attributes are equal to the total set of attributes in the database i.e. all the attributes are filled completely.

Let us consider a small example to demonstrate the working of the proposed method. Table 1 shows the sample database considered for the demonstration. The database contains 30 tuples with 6 attributes A0 to A5 which are binary-valued attributes with two possible values True and False. A class label which has two distinct classes C0 and C1 is associated with each tuple.

Table 2 includes the 12 classification rules which are identified by applying a classification rule mining on the database of Table 1. Rule number 7 of Table 2 is considered as the sensitive classification rule which requires protection from the disclosure.

Consider the non-sensitive rules among the rules mined from the database to construct a decision tree from which the database was reconstructed in classification rule hiding. Hence, among the 12 rules discovered from the database, we are considering 11 rules (other than the rule 7 which is sensitive). For every non-sensitive rule calculate the capability which indicates the classification ability of that rule on the database (Steps 5 to 6 of Algorithm 1). The rules and their capability values are shown in Table 2.

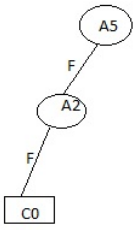


Figure 3: The Decision Tree Path for Rule 1 in Table 2.

Calculate the measure info-gain for every attribute A0 to A5. The info-gain specifies how much information we gained by doing the split using that particular attribute. The attribute which will have maximum info-gain will be better for splitting the database (Steps 7 to 8 of Algorithm 1). The info-gain values of the attributes A0 to A5 are shown in Table 3.

Construction of the decision tree is as follows: consider the non-sensitive rules in the decreasing order of their capability. Hence, consider the rule A2=False & A5=False $\Rightarrow$ C0 which has highest capability 8. The rule contains the attributes A2 and A5. Order these attributes based on the info-gain. Hence the attributes will be considered in the order A5 and A2. Create a path in the decision tree with the values of the rule in the order A5 and A2. The tree is as shown in Figure 3.In figures False is indicated with "F" and True is indicated with "T".

Table 1: The Sample Database.

Tuple.No	A0	A1	A2	A3	A4	A5	Class
1	TRUE	FALSE	FALSE	FALSE	TRUE	FALSE	C0
2	TRUE	TRUE	FALSE	FALSE	FALSE	TRUE	C0
3	FALSE	FALSE	TRUE	TRUE	FALSE	TRUE	C1
4	FALSE	TRUE	FALSE	TRUE	TRUE	FALSE	C0
5	TRUE	TRUE	FALSE	FALSE	FALSE	TRUE	C0
6	TRUE	TRUE	TRUE	TRUE	FALSE	FALSE	C1
7	TRUE	TRUE	FALSE	TRUE	TRUE	FALSE	C0
8	FALSE	FALSE	TRUE	FALSE	TRUE	TRUE	C1
9	TRUE	TRUE	FALSE	TRUE	FALSE	TRUE	C0
10	FALSE	FALSE	TRUE	TRUE	TRUE	TRUE	C1
11	FALSE	TRUE	FALSE	FALSE	FALSE	TRUE	C0
12	TRUE	FALSE	TRUE	FALSE	TRUE	FALSE	C0
13	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	C1
14	FALSE	FALSE	FALSE	TRUE	TRUE	TRUE	C1
15	FALSE	FALSE	FALSE	TRUE	TRUE	FALSE	C0
16	TRUE	FALSE	FALSE	TRUE	FALSE	FALSE	C0
17	TRUE	TRUE	TRUE	FALSE	FALSE	TRUE	C1
18	TRUE	FALSE	FALSE	TRUE	FALSE	TRUE	C0
19	TRUE	FALSE	TRUE	FALSE	FALSE	FALSE	C0
20	FALSE	FALSE	FALSE	TRUE	FALSE	TRUE	C1
21	FALSE	TRUE	FALSE	TRUE	TRUE	TRUE	C1
21	FALSE	TRUE	FALSE	TRUE	TRUE	TRUE	C1
22	FALSE	TRUE	FALSE	TRUE	TRUE	FALSE	C0
23	FALSE	TRUE	TRUE	TRUE	FALSE	FALSE	C0
24	FALSE	TRUE	FALSE	FALSE	TRUE	FALSE	C0
25	TRUE	TRUE	TRUE	FALSE	FALSE	TRUE	C0
26	TRUE	FALSE	TRUE	TRUE	FALSE	FALSE	C0
27	TRUE	FALSE	FALSE	TRUE	FALSE	TRUE	C0
28	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	C1
29	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	C0
30	TRUE	TRUE	FALSE	TRUE	FALSE	FALSE	C0

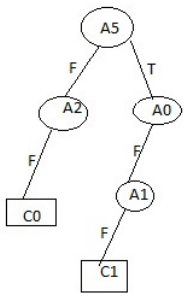


Figure 4: The Decision Tree Path for Rule 2 in Table 2.

Then consider the next rule with maximum capability 5 which is A0=False & A1=False & A5 =True ⇒ C1. Create a path in the decision tree corresponding to this rule as shown in Figure 4.

The next rule in order is A0=True & A2=False & A4=False & A5=True ⇒ C0 with next maximum capability 5. The tree after creating a path in the decreasing order of their info-gain values is as shown in Figure 5.

By repeating the process for all the non-sensitive rules the complete decision tree can be constructed. The complete decision is as shown in Figure 6.

The first path in the decision tree which is with A2 and A5 attributes with false value and class label as C0 is considered and corresponding to this path, 8 (the capability of rule) tuples are inserted into the reconstructed database. The remaining attributes are indicated by null values. To fill these null values consider the combination of the values in the attributes, in which values are available. In this case it is False, False, C0 for the attributes A2, A5 and class correspondingly. By comparing this combination in the original database, the number of tuples found is 8. Count the number of occurrences of each distinct value in each of the attributes A0, A1, A3 and A4. The value True occurred 4, 5, 6 and 6 times in A0, A1, A3 and A4 attributes respectively. The value False occurred 4, 3, 2 and 2 times in A0, A1, A3 and A4 attributes respectively. Since, the majority of the occurrences are for A3 and A4 by value True, the

Table 2: Capability Values of the Classification Rules.

Rule.No	Classification Rules	Capability
1	A2 = False & A5 = False ⇒ C0	8
2	A1 = False & A2 = True & A5 = False ⇒ C0	3
3	A0 = False & A1 = True & A2 = True & A5 = False ⇒ C0	1
4	A0 = True & A1 = True & A2 = True & A5 = False ⇒ C1	1
5	A0 = False & A1 = False & A5 = True ⇒ C1	5
6	A0 = False & A1 = True & A3 = False & A5 = True ⇒ C0	1
7	A0 = False & A1 = True & A3 = True & A5 = True ⇒ C1	–
8	A0 = True & A2 = False & A4 = False & A5 = True ⇒ C0	5
9	A0 = True & A2 = True & A4 = False & A5 = True ⇒ C0	1
10	A0 = True & A1 = False & A4 = True & A5 = True ⇒ C1	1
11	A0 = True & A1 = True & A2 = False & A4 = True & A5 = True ⇒ C1	1
12	A0 = True & A1 = True & A2 = True & A4 = True & A5 = True ⇒ C0	1

Table 3: Information Gain of the Attributes in Table 1.

S.No	Attribute Name	Info - Gain
1	A0	0.0598
2	A1	0.0258
3	A2	0.0598
4	A3	0.0304
5	A4	0.0258
6	A5	0.1835

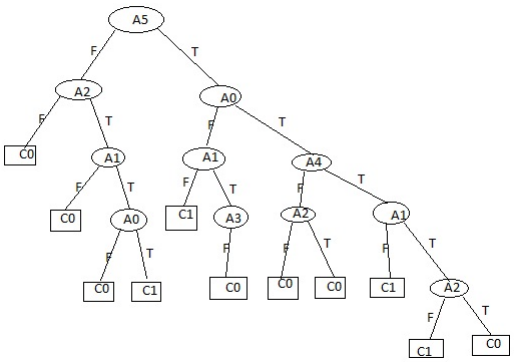


Figure 6: The Complete Decision Tree of all the Rules in Table 2.

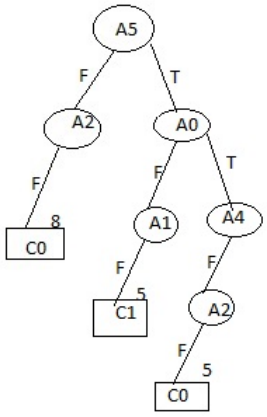


Figure 5: The Decision Tree Path for Rule 3 in Table 2.

missing values of A3 and A4 are filled with value True. Now for the tuples corresponding to the first path, the values are available for A2, A3, A4, A5 and class. Repeat the process for filling of A0 and A1 by considering the combination values in these attributes. After all the attributes are filled up the next path in the tree will be considered in the similar manner until the process of generation and filling will be completed for all the paths in the constructed decision tree. Finally, the reconstructed database obtained is shown in Table 4.

4 Evaluation measures

To assess the performance or efficiency of an algorithm some metrics are to be considered. Classification rule hiding algorithms are also be assessed with a set of measures. The four metrics for the evaluation of the proposed method are as follows:

The first measure is Hiding Failure, which measures the fraction of sensitive classification rules that are revealed from the reconstructed database. Through this, the amount

Table 4: The Reconstructed Database.

Tuple.No	A0	A1	A2	A3	A4	A5	Class
1	FALSE	TRUE	FALSE	TRUE	TRUE	FALSE	C0
2	FALSE	TRUE	FALSE	TRUE	TRUE	FALSE	C0
3	FALSE	TRUE	FALSE	TRUE	TRUE	FALSE	C0
4	FALSE	TRUE	FALSE	TRUE	TRUE	FALSE	C0
5	FALSE	TRUE	FALSE	TRUE	TRUE	FALSE	C0
6	FALSE	TRUE	FALSE	TRUE	TRUE	FALSE	C0
7	FALSE	TRUE	FALSE	TRUE	TRUE	FALSE	C0
8	FALSE	TRUE	FALSE	TRUE	TRUE	FALSE	C0
9	TRUE	FALSE	TRUE	FALSE	FALSE	FALSE	C0
10	TRUE	FALSE	TRUE	FALSE	FALSE	FALSE	C0
11	TRUE	FALSE	TRUE	FALSE	FALSE	FALSE	C0
12	FALSE	TRUE	TRUE	TRUE	FALSE	FALSE	C0
13	TRUE	TRUE	TRUE	TRUE	TRUE	FALSE	C0
14	FALSE	FALSE	TRUE	TRUE	TRUE	TRUE	C1
15	FALSE	FALSE	TRUE	TRUE	TRUE	TRUE	C1
16	FALSE	FALSE	TRUE	TRUE	TRUE	TRUE	C1
17	FALSE	FALSE	TRUE	TRUE	TRUE	TRUE	C1
18	FALSE	FALSE	TRUE	TRUE	TRUE	TRUE	C1
19	FALSE	TRUE	FALSE	FALSE	FALSE	TRUE	C0
20	FALSE	TRUE	FALSE	FALSE	FALSE	TRUE	C0
21	TRUE	TRUE	FALSE	FALSE	FALSE	TRUE	C0
22	TRUE	TRUE	FALSE	FALSE	FALSE	TRUE	C0
23	TRUE	FALSE	FALSE	TRUE	FALSE	TRUE	C0
24	TRUE	FALSE	FALSE	TRUE	FALSE	TRUE	C0
25	TRUE	FALSE	FALSE	TRUE	FALSE	TRUE	C0
26	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	C1
27	TRUE	TRUE	TRUE	FALSE	FALSE	TRUE	C0
28	TRUE	TRUE	TRUE	FALSE	FALSE	TRUE	C0
29	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	C1
30	TRUE	TRUE	TRUE	TRUE	TRUE	FALSE	C0

of sensitive knowledge that is preserved can be also be estimated.

The second and third measures are related to the side-effects of the hiding process. Second metric Miss Cost is one that deals with the fraction of the non-sensitive classification rules which are mined from D and cannot be mined from the reconstructed database D^1. The third metric Artifactual Rules is the fraction of the rules which are not derived from the original database D, but can be derived from the reconstructed database D^1.

The fourth measure is the Usability of the reconstructed database. It is measured through the ability of an attribute to classify the database. In order to increase the usability of the reconstructed database the classification model constructed from the reconstructed database should be as close as to the model constructed with the original database. It means the parameter information gain of the attributes in the reconstructed database must be with the minimum difference with the information gain of the attributes in the original database. Hence usability is calculated as the sum of the differences between the information gains of the attributes in D and D^1.

4.1 Hiding Failure (HF)

The hiding failure is calculated as follows:

$$HF = \frac{|SCR(D^1)|}{|SCR(D)|}$$

where $|SCR(D^1)|$ indicates the number of sensitive classification rules revealed from D^1, and $|SCR(D)|$denotes the number of sensitive classification rules discovered from D.

4.2 Miss Cost (MC)

The miss cost is calculated as:

$$MC = \frac{|NSCR(D)| - |NSCR(D^1)|}{|NSCR(D^1)|}$$

Where $|NSCR(D)|$ refers to the number of non-sensitive classification rules revealed from D and $|NSCR(D^1)|$

Table 5: Characteristics of the Datasets.

S.No	Name of the Database	No.of Instances	No.of Attributes
1	PIMA - DIABETES	768	9
2	GERMAN CREDIT RATING	1000	21
3	CONGRESSIONAL VOTING RECORDS	435	17
4	MUSHROOM	8124	23

refers to the number of non-sensitive classification rules discovered from D^1.

4.2.1 Artifactual Rules (AR)

This is measured as:

$$AR = \frac{|CR'| - |CR \cap CR'|}{|CR'|}$$

Where $|CR|$ and $|CR'|$ stands for, number of classification rules that are generated from D and D^1 respectively.

4.2.2 Usability

The difference between the gains of the attributes is measured as:

$$U = \sqrt{\frac{\sum_{i=1}^{m} (\frac{o_i - r_i}{o_i})^2}{m}} * 100$$

Where o_i and r_i are the gain ratios for the i^{th} attribute on D and D^1 and m is the number of attributes in D.

A classification rule hiding algorithm with no hiding failure and artifactual rules i.e. 0% of HF and AR and with reduced miss cost and high usability of the D^1 is considered as an efficient algorithm.

5 Experimental results

Experiments were conducted by considering the real life databases PIMA-DIABETES, GERMAN CREDIT RATING, CONGRESSIONAL VOTING RECORDS and MUSHROOM which are available in UCI data repository[10]. The characteristics of the databases used in the experiments were shown in Table 5.

The results of the proposed method are compared with a classification rule hiding method by considering gain ratios, proposed by Natwichai in [3]. The Natwichai(Gain) method was also a reconstruction based method. Initially it constructs a decision tree from non-sensitive classification rules, and then each path is simply generated as a set of tuples in reconstructed database. In the proposed method, after constructing the tree from the non-sensitive classification rules and at the time of reconstructing the database the missing values are identified efficiently by considering the probability of the possible values in the original database. Hence the usability of the reconstructed database increases by reducing the miss cost and artifactual rules.

Experiments were conducted with four classification algorithms: C4.5(J48), PART, BF TREE and AD TREE which are rule based algorithms available in weka tool. In the experiments, same classification algorithm was used twice i.e. once on D and second on D^1 to discover the classification rules which are used to evaluate the performance measures. All the experiments were done by selecting only one classification rule as sensitive rule while all the remaining as non-sensitive rules. After the classification rules are generated by the algorithm, randomly one rule is selected as sensitive.

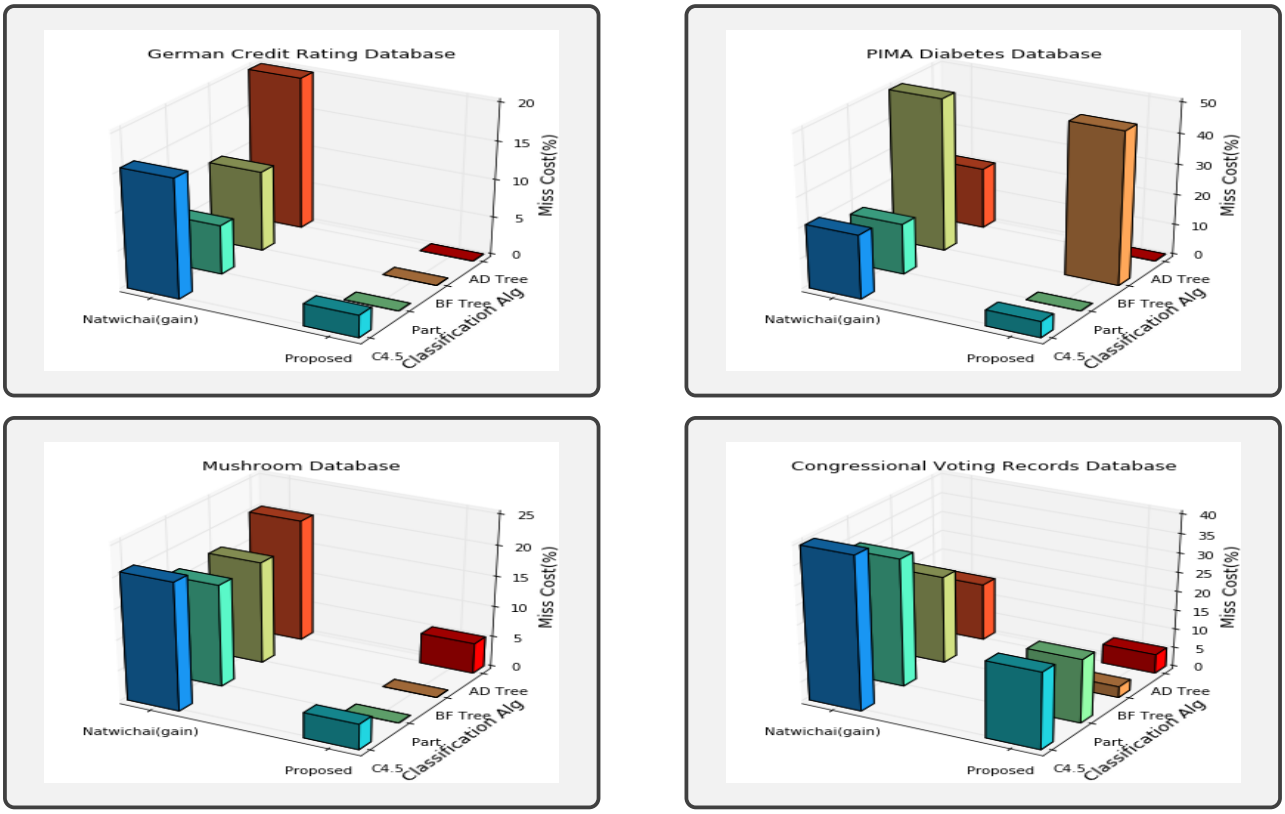
By applying C4.5, PART, BF TREE and AD TREE algorithms on the PIMA-DIABETES database the generated classification rules are 20, 13, 3 and 21 with an accuracy of 84.5, 81.25, 77.21 and 79.69 respectively. After reconstructing the database by using the proposed algorithm, the rules generated are 20, 12, 2 and 20 with an accuracy of 83.98, 80.48, 76.02 and 79.04 respectively. With C4.5 on the reconstructed PIMA-DIABETES database one non-sensitive rule was loosed, and one new rule was generated.

Similarly, by applying C4.5, PART, BF TREE and AD TREE algorithms on the GERMAN CREDIT RATING database the generated classification rules are 103, 78, 39 and 21 with an accuracy of 85.5, 89.7.84.2 and 75.4 respectively. After reconstructing the database by using the proposed algorithm, the rules generated are 101, 77 38 and 20 with an accuracy of 84.9, 89.01, 87.6 and 75.1 respectively. With C4.5 on GERMAN CREDIT RATING reconstructed database one non-sensitive rule was loosed, and three new rules were generated.

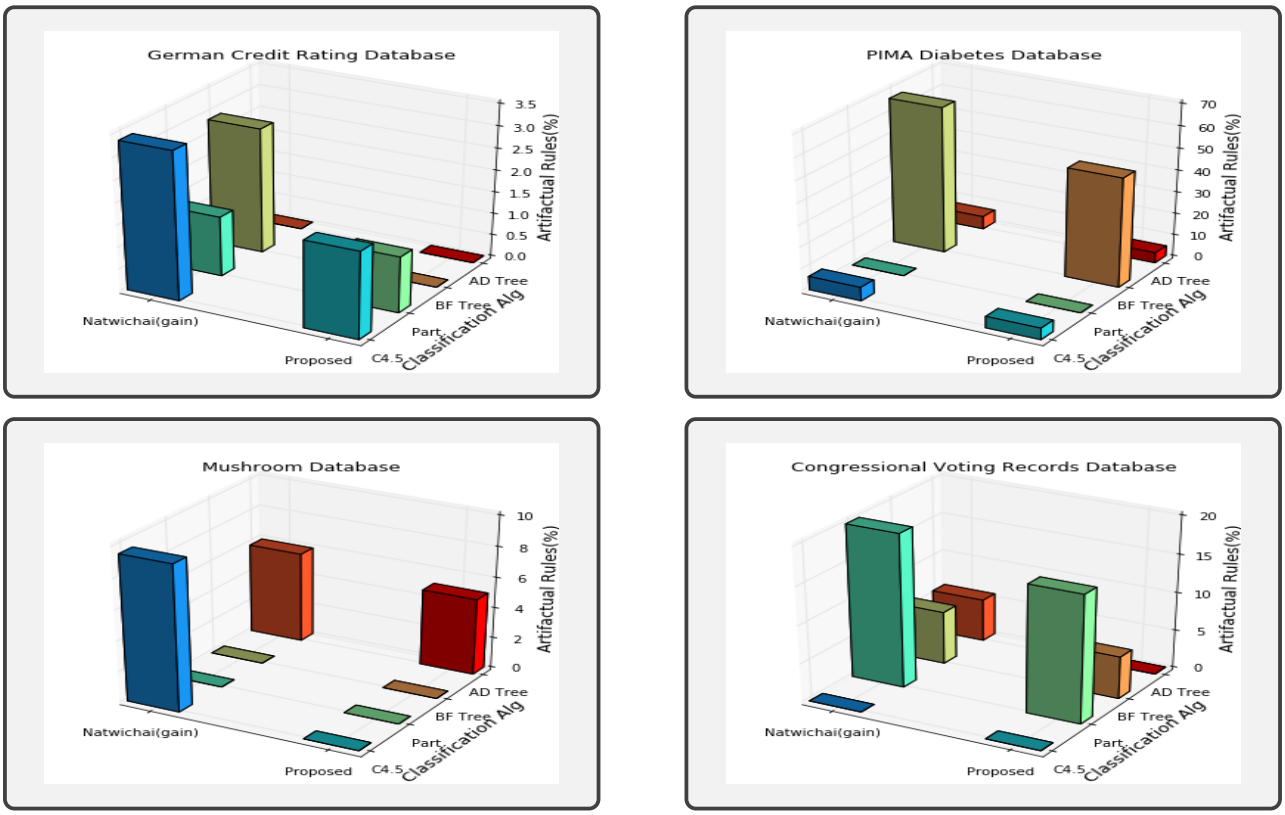
By applying C4.5, PART, BF TREE and AD TREE algorithms on MUSHROOM database the generated classification rules are 25, 13, 7 and 21 with an accuracy of 100, 100, 99.95 and 99.9 respectively. After reconstructing the database by using the proposed algorithm, the rules generated are 23, 12, 6 and 20 with an accuracy of 100, 100, 98.53 and 98.14 respectively. With C4.5 and AD TREE on Mushroom reconstructed database one non-sensitive rule was loosed, and with AD TREE one new rule was generated.

By applying C4.5, PART, BF TREE and AD TREE algorithms on CONGRESSIONAL VOTING database the generated classification rules are 6, 7, 36 and 21 with an accuracy of 97.24, 97.47, 98.39 and 97.93 respectively. After reconstructing the database by using the proposed algorithm, the rules generated are 4, 6, 36 and 19 with an accuracy of 96.25, 95.87, 98.14 and 96.89 respectively. With all the four algorithms on CONGRESSIONAL VOTING reconstructed database one non-sensitive rule was loosed, and 1 and 2 new rules were generated with PART and BF TREE respectively.

The results of the experiments with the proposed method and Natwichai (Gain) method [3] on four databases with four classification algorithms were shown in Table 6 and Table 7 respectively.

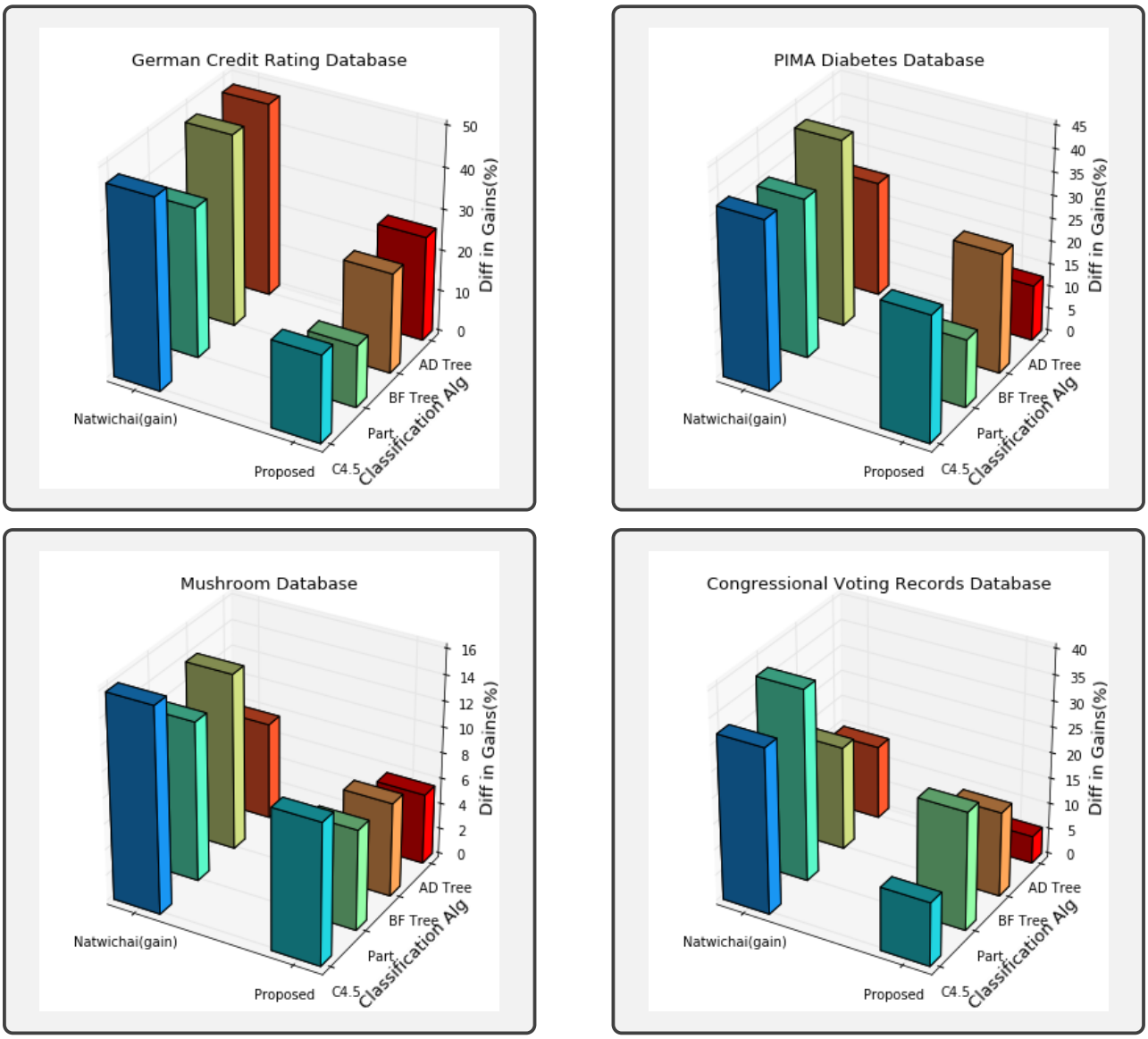


(a)



(b)

Figure 7: (a)Comparison of Miss Cost on Four Databases. (b)Comparison of Artifactual Rules on Four Databases.



(a)

Figure 8: Comparison of Difference between Gains of the Attributes on Four Databases.

Table 6: Experimental Values of Proposed Method.

Database	Classification Algorithm	Original Database		Reconstructed Database		Performance Measures		
		No.of Rules	Accuracy of the Model	No. of Rules	Accuracy of the Model	HF	MC	AR
PIMQ-DIABETES	C4.5	20	84.15	19	83.98	0	1	1
	PART	13	81.25	12	80.48	0	0	0
	BF TREE	3	77.21	2	76.02	0	1	1
	AD TREE	21	79.69	21	79.04	0	0	1
GERMAN CREDIT RATING	C4.5	103	85.5	101	84.9	0	3	2
	PART	78	89.7	78	89.01	0	0	1
	BF TREE	39	84.2	38	87.6	0	0	0
	AD TREE	21	75.4	20	75.1	0	0	0
CONGRESSIONAL VOTING RECORDS	C4.5	6	97.24	4	96.25	0	1	0
	PART	7	97.47	6	95.87	0	1	1
	BF TREE	36	98.39	36	98.14	0	1	2
	AD TREE	21	97.93	19	96.89	0	1	0
MUSHROOM	C4.5	25	100	23	100	0	1	0
	PART	13	100	12	100	0	0	0
	BF TREE	7	99.95	6	98.53	0	0	0
	AD TREE	21	99.9	20	98.14	0	1	1

Table 7: Experimental Values of Natwichai (Gain) Method.

Database	Classification Algorithm	Original Database		Reconstructed Database		Performance Measures		
		No. of Rules	Accuracy of the Model	No. of Rules	Accuracy of the Model	HF	MC	AR
PIMQ-DIABETES	C4.5	20	84.15	16	71.32	0	4	1
	PART	13	81.25	10	66.25	0	2	0
	BF TREE	3	77.21	3	25.73	0	1	2
	AD TREE	21	79.69	17	64.51	0	4	1
GERMAN CREDIT RATING	C4.5	103	85.5	89	71.87	0	16	3
	PART	78	89.7	73	81.93	0	5	1
	BF TREE	39	84.2	35	77.56	0	4	1
	AD TREE	21	75.4	16	61.03	0	4	0
CONGRESSIONAL VOTING RECORDS	C4.5	6	97.24	3	68.62	0	3	0
	PART	7	97.47	5	75.69	0	2	1
	BF TREE	36	98.39	29	80.25	0	8	2
	AD TREE	21	97.93	18	84.93	0	3	1
MUSHROOM	C4.5	25	100	21	89.06	0	5	2
	PART	13	100	10	86.92	0	2	0
	BF TREE	7	99.95	5	81.39	0	1	0
	AD TREE	21	99.9	17	89.62	0	4	1

Generally the performance metrics are to be evaluated in terms of the percentage as % of hiding failure, % of miss cost, % of artifactual rules and % of the difference between the gains of the attributes. The comparison of these parameters for both proposed and Natwichai (Gain) methods was plotted in the Graphs. In both the methods, percentage of hiding failure was zero i.e. no sensitive rules will be generated from the reconstructed databases. So the Graphs are included only for the other two parameters i.e. miss cost, artifactual rules and difference in gains of the attributes in D and D^1. The graphs were drawn in python by considering the Natwichai (Gain) and proposed method on X-axis, the classification algorithms used to generate the rules from the databases are on Z-axis and the parameter used for comparison in terms of percentages on Y-axis.

The comparison of the miss cost on four databases is shown in Figure 7(a). with C4.5, PART, BF TREE and AD TREE algorithms on PIMA-DIABETES the % of miss cost with proposed method was 5.3, 0, 50 and 0 respectively. with same algorithms on GERMAN CREDIT RATING database the % of miss cost with proposed method was 2.9, 0, 0 and 0 respectively. with same algorithms on CONGRESSIONAL VOTING RECORDS database the % of miss cost with proposed method was 20, 16.67, 2.8 and 5 respectively. with same algorithms on MUSHROOM database the % of miss cost with proposed method was 4.17, 0, 0 and 5 respectively. In all the four databases the percentage of miss cost was reduced in proposed method when compared to the existing method.

The comparison of artifactual rules on four databases is shown in Figure 7(b). with C4.5, PART, BF TREE and AD TREE algorithms on PIMA-DIABETES the % of artifactual rules with proposed method was 5.3, 0, 50 and 4.8 respectively. with same algorithms on GERMAN CREDIT RATING database the % of artifactual rules with proposed method was 1.9, 1.3, 0 and 0 respectively. with same algorithms on CONGRESSIONAL VOTING RECORDS database the % of artifactual rules with proposed method was 0, 16.67, 5.7 and 0 respectively. with same algorithms on MUSHROOM database the % of artifactual rules with proposed method was 0, 0, 0 and 5 respectively. In all the four databases the percentage of ghost rules generated was reduced in the proposed method when compared to the existing method.

The comparison of the difference between the information gains of the attributes in four databases is shown in Figure 8(a). with C4.5, PART, BF TREE and AD TREE algorithms on PIMA-DIABETES the % of difference between the information gains with proposed method was 27.42, 14.83, 26.12 and 12.23 respectively. with same algorithms on GERMAN CREDIT RATING database the % of difference between the information gains with proposed method was 21.19, 15.1, 24.3 and 25.5 respectively. with same algorithms on CONGRESSIONAL VOTING RECORDS database the 5.3 respectively. with same algorithms on MUSHROOM database the % of difference between the information gains with proposed method was 10.9, 7.7, 7.3 and 5.5 respectively. The proposed algorithm reduces the difference in gains of the attributes thereby in-

creasing the usability of the reconstructed database which is going to be released without compromising on privacy of the sensitive rules.

Hence, the experimental assessment clearly indicates that the proposed method will reconstruct a database by hiding all the sensitive rules, with minimum loss in non-sensitive rules, minimum artifactual rules generated and by improving the usability of the reconstructed database.

6 Conclusion

Preserving the privacy of sensitive classification rules is a very important issue in application areas that involves collaboration with data sharing. A new algorithm is projected for defending the sensitive classification rules from disclosure. With the projected method which is reconstruction based classification rule hiding, new database will be reconstructed from which sensitive rules will not be disclosed and the side effects of the hiding process miss cost and artifactual rules are kept minimal. Moreover, the usability of reconstructed database will be maximized to make it useful with valid data mining results for a data analyst. The experimental analysis of the results is the evidence to indicate that the proposed algorithm is effective, i.e. it can preserve the privacy and data utility very well.

References

[1] Mahdi Aghasi and Rozita Jamili Oskouei (2016),Privacy Preserving Data Mining Survey of Classifications, *Soft Computing Applications, Advances in Intelligent Systems and Computing*,Springer.

[2] Neha Jain and Lalit Sen Sharma, An Ontology based on the Methodology Proposed by Ushold and King, International Journal of Synthetic Emotions, Volume 7 Issue 1, January 2016 , Pages 13-26, DOI: 10.4018/IJSE.2016010102.

[3] Reena, Raman Kumar, Effect of Randomization for Privacy Preservation on Classification Tasks, Proceedings of the International Conference on Informatics and Analytics(ICIA-2016), ICPS: ACM International Conference Proceeding Series.

[4] Kalles, Dimitris, Vassilios S. Verykios, and Athanasios Papagelis. "Hiding decision tree rules by data set operations." Information, Intelligence, Systems and Applications (IISA), 2015 6th International Conference on. IEEE, 2015.

[5] Aldeen, Yousra Abdul Alsahib S., Mazleena Salleh, and Mohammad Abdur Razzaque. "A comprehensive review on privacy preserving data mining." SpringerPlus 4.1 (2015): 694.

[6] Xu L.,Jiang C.,Wang J.,Yuan J.and Ren Y.(2014).Information security in big data: privacy and data mining.IEEE, 1149-1176.

[7] Dhanalakshmi, M., and E. Siva Sankari. (2014).Privacy preserving data mining techniques-survey.*In Proc.of International Conference on Information Communication and Embedded Systems(ICICES)*,IEEE.

[8] Chouragade, Komal N., and Trupti H. Gurav. (2014).A Survey on Privacy-Preserving Data Mining using Random Decision Tree.*Int.J.Science and Research*,pp 2891-2894.

[9] Taneja S., Khanna S.,Tilwalia S. and Ankita .(2014).A Review on Privacy Preserving Data Mining:Techniques and Research Challenges.*International Journal of Computer Science and Information Technologies* pp 2310-2315.

[10] Lichman M. (2013). UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science.

[11] Stephen O'Shaughnessy and Geraldine Gray, Development and Evaluation of a Dataset Generator Tool for Generating Synthetic Log Files Containing Computer Attack Signatures, : International Journal of Ambient Computing and Intelligence (IJACI),2011, DOI: 10.4018/jaci.2011040105.

[12] Alka Gangrade , Durgesh Kumar Mishra , Ravindra Patel, Classification Rule Mining through SMC for Preserving Privacy Data Mining: A Review, International Conference on Machine Learning and Computing IPCSIT vol.3 (2011) © (2011) IACSIT Press, Singapore.

[13] Hatice Gunes and Maja Pantic, Automatic, dimensional and Continuous Emotion recognition, International Journal of Synthetic Emotions, Volume 1, Issue 1 © 2010, IGI Global, DOI: 10.4018/jse.2010101605.

[14] Delis A, Verykios V.S, Tsitsonis A.(2010) A Data Perturbation Approach to Sensitive Classification Rule Hiding,*25th Symposium On Applied Computing*.

[15] Gkoulalas-Divanis A, Verykios VS (2010) Association rule hiding for data mining. Springer, New York.

[16] Kadampur M., D.V.L.N S.(2010)A noise Addition scheme in Decision tree for privacy preserving data mining. *Journal of Computing*,137-144.

[17] Upadhayay A. K., Aggarwal A.,Masand R. and Gupta R. (2009).Privacy Preserving Data Mining: A New Methodology for Data Transformation. *Proc. of the First International Conference on Intelligent Human Computer Interaction*, Springer India.

[18] Aliki Katsarou, Aris Gkoulalas-Divanis, and Vassilios S. Verykios (2009) Reconstruction-based Classification Rule Hiding through Controlled Data Modification, *IFIP International Federation for Information Processing, Volume 296; Artificial Intelligence Applications and Innovations III*,Springer pp. 449–458.

[19] Juggapong Natwichai Xue Li Maria E. Orlowska (2006) Reconstruction-based Algorithm for Classification Rules Hiding, *Seventeenth Australasian Database Conference (ADC2006), Hobart, Australia. Conferences in Research and Practice in Information Technology (CRPIT)*, Vol.49.

[20] K. Wang, B.C.M. Fung, P.S. Yu (2005) Template-Based Privacy Preservation in Classification Problems, *5th IEEE International Conference on Data Mining*, pp. 466-473.

[21] Natwichai J., Li X.,Orlowska M. (2005), Hiding classification rules for data sharing with privacy preservation, *Proceedings of 7th International Conference on Data Warehousing and Knowledge Discovery*, Lecture Notes in Computer Science, Springer, pp. 468–467.