Abstract

Background: Data mining and privacy preserving are scientific concepts, the former of which detects the relationships among items of databases and the latter preserves sensitive rules. Their disclosure will threaten the database owner’s security. Methods: Sanitizing the database by privacy preserving algorithms causes some side effects such as lost rules, dissimilarity, etc. All privacy preserving algorithms attempt to reduce these side effects. In addition to the side effects, inferring sensitive data from insensitive data should also be taken into account. Results: In this paper, a new technique is presented in order to block inference channels and reduce lost rules caused by database sanitization. In this paper, combination of distortion and blocking techniques enabled the researcher to block the inference channels and also to reduce lost rules and CPU usage compared to CR algorithm. Finally, the proposed algorithm was assessed with CR and DSR algorithms and the assessment results indicated efficiency of it. Applications: The proposed algorithm had better performance compared to the CR. It was attempted to proximate the results obtained by the suggested algorithm to those obtained by the DSA.

Keywords: Association Rules, Data Mining, Hiding Sensitive Rules, Privacy Preserving, Sensitive Pattern

1. Introduction

Data mining technology detects the relationships between items in a database based on support and confidence criteria. Using two criteria of Minimum Confidence Threshold (MCT) and Minimum Support Threshold (MST), some beneficial and usable relationships were selected from the extracted relations. These criteria are determined by a data miner. Since the privacy preserving algorithm attempt to hide sensitive information with respect to MST and MCT criteria, the inference issue arises. Data inference refers to detection of sensitive data using insensitive data. Most privacy preserving algorithms hide sensitive data regardless of data inference issue and their aim is to reduce side effects caused by database sanitization. These side effects include lost rules, hiding failure and dissimilarity. A data miner can detect the sensitive data in data inference system when the criteria for extracting association rules were estimated less than those criteria selected by the database owner to sanitize database. In addition, the data miner can disclose sensitive information using insensitive information. Two types of Forward-Inference Attack and Backward-Inference Attack were introduced in work. The data miner succeeded in disclosing sensitive information using these attacks. Furthermore, an algorithm called DSA was introduced, which aimed to block inference channels. In fact, DSA algorithm proceeds to sanitize after extracting the association rules. It specifies relationships between association rules using a graph, then, it sanitizes based on sensitive rules. In this paper, an algorithm is presented, which aims to block the inference channels besides causing minimal side effects in the time of database sanitization. Moreover, this algorithm does not require minimum support threshold and minimum confidence threshold. Therefore, the data miner would not succeed in detecting the sensitive information using every available criterion. The major difference between the proposed algorithm and the DSA algorithm lies in the fact that the DSA algorithm just shares the association rules, so the data miner cannot detect the sensitive rules.
Using Distortion and Blocking Techniques for Preventing Association Rules’ Discovery

using inference system. However, the proposed algorithm shares a modified database and enables the data miner to extract the association rules by using any MST and MCT criteria. It should be noted that the data miner could not extract the sensitive rules by using every MST and MCT criteria and because of the technique used in this algorithm, inference is not possible.

2. Frame of Association Rules

As mentioned earlier, association rules are extracted from the database by two criteria of confidence and support. If we assume \( D = \{ T_1, T_2, \ldots, T_n \} \) as database and \( I = \{ i_1, i_2, \ldots, i_n \} \) as database items, the association rules are as \( X \rightarrow Y \) on the condition that \( X \cap Y = \emptyset \). The support of \( X \rightarrow Y \) is defined by computing the proportion of simultaneous frequency of \( X \) and \( Y \) in transactions over the total number of transactions of database. Equation 1 shows the way of calculating rule support.

\[
\text{Support}(X \rightarrow Y) = \frac{|X \cup Y|}{|D|} \tag{1}
\]

The confidence of a rule \( X \rightarrow Y \) is defined by computing the proportion of simultaneous frequency of \( X \) and \( Y \) in transactions over the number of \( X \) frequency alone in transaction of database. Equation 2 shows the way of calculating rule Confidence.

\[
\text{Confidence}(X \rightarrow Y) = \frac{|X \cup Y|}{|X|} \tag{2}
\]

3. Relevant Works

In general, two support-based and confidence-based approaches can be used to hide the sensitive rules\(^1\). In support-based approach, the support of sensitive rule is reduced by eliminating one of the consisting elements of the sensitive rule. The element on right side of the sensitive rule is the best element to be eliminated. There are two hiding methods in confidence-based approach. The first method is increasing the support of the item set on the left side of sensitive rule. This method has the highest rate of hiding failure. In the second method, the support of item on the left is fixed while support of total item set of the sensitive rule is reduced. In this case, the left item set is eliminated and inserted in another transaction. In this method In addition to quickly hiding of the sensitive rules, dissimilarity of the database is reduced to zero in this method.

Most privacy preserving algorithms use two distortion and blocking techniques to sanitize the database. In this section, related works are classified based on these techniques.

3.1 Support-based Technique

3.1.1 Distortion Approach (Eliminating the Victim Item)

Attallah et al. have introduced sensitive association rules’ hiding in 1999 for the first time. They suggested a Lattice-like graph and hid sensitive rules by reducing supports of abundant element set\(^2\). Wang et al. presented two algorithms to hide association rules. DSR algorithm by selecting a sensitive item by database owner detects all sensitive rules of a single element in which the sensitive item is on the left side of the sensitive rules. Then, the algorithm eliminates the sensitive item from the transactions which fully support the sensitive rule. The advantage of this algorithm is decreasing ghost rules and hiding failure and disadvantages of this algorithm are lost rules and high amount of dissimilarity\(^3,4\).

Domadiya et al. proposed an algorithm called MDSRRC. The MDSRRC algorithm is proposed to improve the ADSRRC algorithm. This algorithm hides the number of items on the left and right sides of sensitive rules with no limitations by selecting the best item on the right side based on their sensitivity and support. One disadvantage of this algorithm is sorting the transactions orderly based on the selected item. In certain cases, this algorithm leads to hiding failure. The advantage of this algorithm is reducing the lost rules\(^5\).

Hong et al. proposed an algorithm called SIF-IDF algorithm for hiding sensitive item sets. This algorithm selects the best transaction using SIF-IDF and eliminates the victim item. The algorithm expends high run time for sanitization. In this algorithm, hiding failure is equal to zero\(^6\).

In \(^7\) generalized the hiding problem to a combination of sensitive rules hiding and sensitive item set hiding. Algorithm 1.b selects the best subset from the item set on right side of the sensitive rule and selects the first item as the victim item and eliminates the latter from those transactions fully supporting the sensitive rule to hide the sensitive rule. The advantages of this algorithm
are reducing hiding failure and relatively proper CPU usage.

In 8 proposed two algorithms based on two database scans. Certain indexes are determined for transactions in the first scan while sanitization is performed in the second scan. The algorithm is proposed to reduce the number of lost rules. Disclosure limit threshold is used in this algorithm, which specifies how many association rules were extracted and causes a compromise among the lost rules and hidden rules. The proposed algorithms are Round Robin and Random in which the first algorithm selects the sensitive items alternatively and the second algorithm selects the sensitive items randomly.

In 9 introduced an algorithm based on genetic algorithm. In this algorithm genetic algorithm operators are used for hiding association rules. It must be noted that in this algorithm pre-processing is utilized before hiding operations. Pre-processing on database transactions leads to reduction of searching storage space and algorithm operation time. 4 assessment functions are proposed in the algorithm which do association sensitive rules’ processing by reducing sensitive rules’ support, reducing confidence coefficient of sensitive rules, reducing support and reducing confidence coefficient, respectively.

Oliveira and Zaïane 10 were the first ones suggested hiding several rules simultaneously. They presented MaxFIA and MinFIA algorithms which hide rules without considering number of sensitive elements and with two database scanning. In first scanning they identified sensitive transactions and determined indices for them. In second scanning sanitization was done by selective elimination of minimal single elements (MinFIA by selecting the element with least support and MaxFIA by selecting the element with most support).

11 suggested an algorithm called integer programming which its aims are to decrease changes and to increase accuracy. They proposed two strategies of blanket and intelligent for sanitization. The advantages of this algorithm are quick hiding, assuring the best level of accuracy, Formulization for measuring and efficient problem solving. 12 suggested 3 algorithms first one called Aggregate, reduces sensitive rules’ support by eliminating some transactions. Second one called Disaggregate, reduces sensitive rules’ support by eliminating some sensitive elements. The third algorithm called Hybrid determines considering transactions by using Aggregate notion and determines needed elements for elimination by using disaggregate notion.

13 proposed a heuristically algorithm called ABC based on honey bees’ motions for finding food sources. In this algorithm the best transaction for eliminating sensitive items is found by using random selection of transactions and calculating possibilities of selected transactions, thus no beneficial association rules will be missed.

3.1.2 Blocking Approach

14 proposed three algorithms for sanitizing the database. The first algorithm called GIH reduces support of the sensitive rule, in order to hide the sensitive rule. The second and third algorithms called respectively CR and CR2 hide the sensitive rule by reducing confidence in the sensitive rule. CR algorithm replaces 1 with it and CR2 algorithm replaces 0. GIH and CR algorithms are associated with less side effects compared to CR2.

3.2 Confidence-based Techniques

3.2.1 Distortion Approach (Deletion and Insertion of Victim Item)

15 presented two algorithms for sanitizing the database. RRLR algorithm was proposed to eliminate restrictions on the number of items on the right side of the ADSRRRC algorithm. In this algorithm, the victim item on the left side of the sensitive rule is sensitive. In the first stage of hiding, the sensitive item is eliminated from the transaction which fully supports the sensitive rule. In the second stage, the eliminated item is inserted in the transaction which does not support the sensitive rule. The advantage of this algorithm is reducing dissimilarity and lost rules. Disadvantages of this algorithm are restriction on the number of items on the left hand side and hiding failure 16,17.

The second algorithm proposed by Wang et al. called ISL sanitizes based on confidence-based technique. The ISL algorithm uses a sensitive item which database owner has specified earlier in order to select rules for one item on the left side of sensitive item. Then, the algorithm inserts the item on the left side among transactions which do not support the sensitive rule. Advantage of this algorithm is reducing the lost rules. Disadvantages of this algorithm are hiding failure and generating ghost rules 4.

Algorithm 1.a is another algorithm proposed by 7 Algorithm 1.a hides sensitive rule by increasing support of the item set at left side of the sensitive rule. Advantage of this algorithm is supporting the rules by several items on the left and right sides. In addition, the lost rules are
reduced. Disadvantages of this algorithm are hiding failure and generating lots of ghost rules.

3.2.2 Blocking Approach

proposed two algorithm called WSDA and BA. The former uses data distortion technique while the latter uses blocking technique for hiding the sensitive rules. The BA algorithm hides the sensitive rules by inserting. The algorithm decreases confidence in sensitive rules to below MCT-SM. This will reduce the possibility of inferring sensitive rule. In this algorithm, transactions are divided into two categories. The first category are transactions which fully support the sensitive rule. The second category are transactions partially support the left side of sensitive rule. This category does include right side of the sensitive rule. In this algorithm, the item is eliminated in the first category and is inserted instead of the eliminated item in the second category. The number of insertions and deletions are calculated as N0 and N1.

4. The Proposed Solution

In this paper, a hybrid technique is presented, which combines distortion and blocking techniques. The hybrid technique aims are hiding the sensitive rules along with blocking inference channels with minimum lost rules. In this technique, two confidence-based and support-based approaches were used in combination.

In the hybrid technique, because removing item, the inference possibility is eliminated. In addition, it is not necessary to specify MST and MCT in this technique. The sanitized database does not disclose sensitive rules when the database is data mined with any criteria.

In the proposed algorithm, we specify which approach should be used for hiding according to the selected item. If the selected item was on the right side, the support-based approach will be used. If the victim item was on the left side, confidence-based approach will be used.

In the hybrid technique, the victim item is eliminated as follows:

- If the transaction fully supports the sensitive rule and if the transaction does not include the victim item will be eliminated and replaced.
- If the transaction does not fully support the sensitive rule and if the transaction does not include the victim item will be eliminated without insertion.
- If confidence-based technique was selected for hiding the sensitive rule in the hybrid technique, the item would be eliminated on the condition that the victim item was eliminated. However, the victim item is inserted as follows:
  - If the transaction does not fully support the sensitive rule and does not include the victim item, the victim item will be inserted.
  - If the transaction does not fully support the sensitive rule and does not include the victim item while it is presented in the transaction, the victim items will be inserted.

4.1 Introducing Signs and Concepts used in the Algorithm

Rh shows the set of sensitive rules, Lr denotes left side of the sensitive rule, r shows right side of the sensitive rule, DESC denotes descending sort while ASC shows ascending sort. Algorithm 1 illustrates pseudo code for proposed algorithm.

Degree of transactional interaction: the number of sensitive rules which fully supports a transaction.

Support of a sensitive item: the number of repeating a sensitive item in the original database.

Algorithm 1: Pseudo code for proposed algorithm

Input: a set R_h of rules to hide, the source database D
Output: The Sanitized Database D'
Find Sensitive Item and sort them with Sensitivity DESC and Support ASC
Sort R_h with Support DESC
For each rule r in R_h do
  Select first sensitive item where Sensitive item r
  If victim item in LHS r
    For each t in D do
      Sort t ∈ D with Conflict DESC and Length ASC
      If t fully support r then
        If belong in t
          Delete victim item and insert
        Else
          Delete victim item
      For each t in D do
        Sort t ∈ D with Conflict ASC and Length ASC
        If t partially support L_r and doesn't fully support r,
          If belong in t
            Insert victim item
          Else
            Delete victim item
  Else
    Delete victim item
Insert victim item
Else
For each t in D do
  If t fully support r then
    If belong in t
      Delete victim item and insert
    Else
      Delete victim item
  Else
  Insert victim item

4.2 An example
Consider the database in Table 1. Consider the sensitive rule h → cd, e → cd and d → ac. According to the algorithm, the sensitive rule with maximum support is selected at first. Then, the victim item is selected based on the selected rule. Finally, sanitization operation is specified. For every three sensitive rule, item c is selected as the victim item since this item on the right side just operates the elimination operation. Table 2 shows the sanitized database.

5. Comparison and Evaluation
The proposed algorithm is assessed with CR and DSR algorithms. CR algorithm results depend on SM value. If the SM value increases, more inference channels are blocked and the data miner cannot detect the sensitive rules with less criteria. On the other hand, if more criteria are used, more rules are lost and dissimilarity increases. In this study, we assumed SM = 15, so that less rules would be lost. DSR algorithm attempts to hide the rules with a single item on the left and right sides by eliminating the victim item. This helps to minimize the inference considering the forward inference attack despite it has many lost rules. On the other hand, DSR algorithm hides the sensitive rules to certain MST and MCT criteria. Thus, if data miner performs less number of data mining operations, more sensitive rules will be disclosed.

We used two Chess and mushroom databases for assessment. Characteristics of these databases are shown in Table 3.

Since the DSA algorithm in most cases strongly blocks the inference channels with less lost rules, we attempted to proximate the results obtained by the proposed algorithm to those obtained by the DSA algorithm. Since the DSA algorithm randomly selects victim item sets in most cases, the results of each run may vary with previous CPU usages.

As observed in Figures 1 and 2, fewer rules are lost in the proposed algorithm compared to CR. Since the

Table 1. Sample database

<table>
<thead>
<tr>
<th>TID</th>
<th>Items</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>acd</td>
</tr>
<tr>
<td>1</td>
<td>acdh</td>
</tr>
<tr>
<td>2</td>
<td>abcdefh</td>
</tr>
<tr>
<td>3</td>
<td>abdfg</td>
</tr>
<tr>
<td>4</td>
<td>bcde</td>
</tr>
<tr>
<td>5</td>
<td>abc</td>
</tr>
<tr>
<td>6</td>
<td>cdefh</td>
</tr>
<tr>
<td>7</td>
<td>abcg</td>
</tr>
<tr>
<td>8</td>
<td>bcfg</td>
</tr>
<tr>
<td>9</td>
<td>acdg</td>
</tr>
<tr>
<td>10</td>
<td>abcede</td>
</tr>
<tr>
<td>11</td>
<td>acdefh</td>
</tr>
<tr>
<td>12</td>
<td>abcdh</td>
</tr>
</tbody>
</table>

Table 2. Final sanitized database

<table>
<thead>
<tr>
<th>TID</th>
<th>Items</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>ad</td>
</tr>
<tr>
<td>1</td>
<td>adh</td>
</tr>
<tr>
<td>2</td>
<td>abdefh</td>
</tr>
<tr>
<td>3</td>
<td>abdfg</td>
</tr>
<tr>
<td>4</td>
<td>bde</td>
</tr>
<tr>
<td>5</td>
<td>abc</td>
</tr>
<tr>
<td>6</td>
<td>adefh</td>
</tr>
<tr>
<td>7</td>
<td>abcg</td>
</tr>
<tr>
<td>8</td>
<td>bcfg</td>
</tr>
<tr>
<td>9</td>
<td>adg</td>
</tr>
<tr>
<td>10</td>
<td>abde</td>
</tr>
<tr>
<td>11</td>
<td>adefh</td>
</tr>
<tr>
<td>12</td>
<td>abdh</td>
</tr>
</tbody>
</table>

Table 3. Characteristics of the real database

<table>
<thead>
<tr>
<th>Database name</th>
<th>Number of transactions</th>
<th>No. items</th>
<th>Avg. trans. Length</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chess</td>
<td>3196</td>
<td>74</td>
<td>37</td>
<td>Dense</td>
</tr>
<tr>
<td>Mushroom</td>
<td>8124</td>
<td>119</td>
<td>23</td>
<td>Sparse</td>
</tr>
</tbody>
</table>
lost rules in CR algorithm depend on SM value, it can be concluded that more rules are lost as SM value increases. On the other hand, the data miner can detect sensitive rules in case of using less MST and MCT than the designated MST and MCT for hiding. Therefore, if SM value decreases, the data miner possibly detects more sensitive rules.

Figures 3 and 4 show dissimilarity in the database. The rate of dissimilarity in CR algorithm depends on SM value. If SM value increases, the dissimilarity will be greater. However, dissimilarity in the proposed algorithm is a constant value. According to this issue, if SM increases, dissimilarity of CR algorithm will be greater than the proposed algorithm. Dissimilarity of DSR algorithm is
Figure 6. CPU usage for hiding sensitive rules in Mushroom database.

smaller than other algorithms due to lack of insertion of the element in database.

Since the proposed algorithm does not consider MST and MCT criteria for hiding and the transactions are sorted based on degree of conflict, sanitization time is reduced. Figures 5 and 6 show CPU usage of the proposed algorithm, CR algorithm and DSR algorithm.

6. Conclusion

The database owner specifies MST and MCT criteria for sanitizing the database in a way that sensitive rules would not be disclosed after publishing the database. Since selecting the criteria is strict and sensitive rules might be disclosed by the database, we proposed an algorithm in which selecting MST and MCT criteria was not needed. The proposed algorithm not only blocks the inference channels, but also reduces side effects such as lost rules, dissimilarity and CPU usage. However, the CR algorithm presents criterion called SM to database owner to preserve sensitive information. This criterion tries to balance the side effects. According to the results, the proposed algorithm had better performance compared to the CR algorithm. On the other hand, it was attempted to approximate the results obtained by the proposed algorithm to those obtained by the DSA algorithm, However, the DSA algorithm results are random.

7. References