Researchers

Prof Padam Gulwani

HOD (Computer Science & Engg)

People's college of Research & Technology

People's University , Bhopal (M. P.)

Email- padam24in@yahoo.co.in

A Novel Approach for Association Rule Hiding

Abstract:

Many strategies had been proposed in the literature to hide the information containing sensitive items. Some use distributed databases over several sites, some use data perturbation, some use clustering and some use data distortion technique. Present paper focuses on data distortion technique. Algorithms based on this technique either hide a specific rule using data alteration technique or hide the rules depending on the sensitivity of the items to be hidden. The proposed approach is based on data distortion technique where the position of the sensitive items is altered but its support is never changed. The proposed approach uses the idea of representative rules to prune the rules first and then hides the sensitive rules. Experimental results show that proposed approach hides the more number of rules in minimum number of database scans compared to existing algorithms based on the same approach i.e. data distortion technique.

Introduction:

PRIVACY is becoming an increasingly important issue in many data-mining applications that deal with health care, security, financial and other types of sensitive data. It is also becoming important in counter terrorism and defense related applications. These applications may require creating profiles, constructing social network models, and detecting terrorist communications. This calls for well-designed techniques that pay careful attention to hiding privacy-sensitive information [2], while preserving the inherent statistical dependencies, which are important for data mining applications. There are many techniques available for privacy preserving data mining.

Privacy Preserving Techniques for Data mining

There are many techniques, which have been adopted for privacy preserving data mining. We can classify them based on the following dimensions:

- Data distribution
- Data modification
- Data mining algorithm
- Data or rule hiding
- Privacy preservation

The first dimension refers to the distribution of data. Some of the approaches have been developed for centralized data, while others refer to a distributed data scenario. The second dimension refers to the data modification scheme. In

VOLUME 1 NUMBER 3 (May/June 2012)
ISSN: 2277–1891

general, data modification [1] is used in order to modify the original values of a database that needs to be released to the public and in this way to ensure high privacy protection. It is important that a data modification technique should be in concert with the privacy policy adopted by an organization. The third dimension refers to the data mining algorithm, for which the data modification is taking place. This is actually something that is not known beforehand, but it facilitates the analysis and design of the data hiding algorithm. Among them, the most important ideas have been developed for classification data mining algorithms, like decision tree inducers, association rule mining algorithms, clustering algorithms, rough sets[4] and Bayesian networks. The fourth dimension refers to whether raw data or aggregated data should be hidden.

The complexity for hiding aggregated data in the form of rules is of course higher, and for this reason, mostly heuristics have been developed. The lessening of the amount of public information causes the data miner to produce weaker inference rules that will not allow the inference of confidential values. This process is also known as “rule confusion”. The last dimension, which is the most important, refers to the privacy preservation technique [2] used for the selective modification of the data. Selective modification is required in order to achieve higher utility for the modified data given that the privacy is not jeopardized.

Problem Statement

Association Rule Mining:

The problem of mining association rules was introduced in [2]. Let I = { i,, i,;.., i, } be a set of literals, called items. Given a set of transactions D, where each transaction T is a set of items such that $T \subseteq I$, an association rule[1] is an expression $X \Rightarrow Y$ where $x \subseteq I$, $Y \subseteq I$, and $X \cap Y = \Phi$ .The X and Y are called respectively the body (left hand side) and head (right hand side) of the rule. An example of such a rule is that 90% of customers buy hamburgers also buy Coke. The 90% here is called the confidence of the rule, which means that 90% of transaction that contains X also contains Y.

The confidence is calculated as $|X \cup Y| / |x|$.The support of the rule is the percentage of transactions that contain both X and Y. which is calculated as $|X \cup Y| /N$. In other words, the confidence of a rule measures the degree of the correlation between itemsets, while the support of a rule measures the significance of the correlation between itemsets. The problem of mining association rules is to find all rules that are greater than the user-specified minimum support and minimum confidence

Description of Problem

Sensitive Rules Hiding (By Changing the Support or Confidence of the Rules).

The main objective of rule hiding is to transform the database such that the sensitive rules are masked, and all the other underlying patterns can still be discovered. For doing this the support or the confidence of the large item sets or the association rule is changed which helps in hiding them. In this regard, the minimum support and confidence will be altered into a minimum interval. As long as the support and/or the confidence of a sensitive rule lie below the middle in these two ranges, the confidentiality of data is expected to be protected. The rule hiding method hides a group of association rules, which is characterized as sensitive. One rule is characterized as sensitive if its disclosure risk is above a certain privacy threshold. Sometimes, sensitive rules should not be disclosed to the public since, among other things, they may be used for inferring sensitive data, or they may provide business competitors with an advantage

Related Work

Following section discusses two methods of hiding rules technique along with their complete analysis.

One Rule at A Time (Proposed By Veryki- os Et Al, Etc.) Distortion Based Technique (Sanitization)

In this work [1] authors propose strategies and a suite of algorithms for hiding sensitive knowledge from data by minimally perturbing their values. The hiding strategies proposed are based on reducing the support and confidence of rules that specify how significant they are .In order to achieve this, transactions are modified by removing some items, or inserting some new items depending on the hiding strategy. The constraint on the algorithms proposed is that the changes in the database introduced by the hiding process should be limited, in such a way that the information incurred by the process is minimal. Selection of the items in a rule to be hidden and the selection of the transactions that will be modified is a crucial factor for achieving the minimal information loss constraint.

On the basis of sensitive item (proposed by shyue-liang wang et al.) Distortion based Technique (sanitization)

Technique proposed in this work tries to hide certain specific items that are sensitive and proposes two algorithms to modify data in the Dataset so that sensitive items cannot be inferred through association rule mining algorithms. Concept of this paper says that if the sensitive item is on the LHS of the rule then increase its support and if the sensitive item is on the right of the rule then decrease its support. This work is in contrast with previous work as approach in [1] hides a specific rule and the approach in [2] tries to hide all the rules containing sensitive items (either in the right or in the left)

Analysis of Existing Techniques

Existing approaches have some problems. Data perturbation [5] considers the applications where individual data values are confidential rather than the data mining results and concentrated on a specific data mining model, namely, the classification by decision trees.
Additive noise can be easily filtered out in many cases that may lead to compromising the privacy.

A potential problem of traditional additive and multiplicative perturbation is that each data element is perturbed independently; therefore the pair-wise similarity of records is not guaranteed to be maintained.
In Secure Multiparty Computation (SMC) the functionality f to be computed is first represented as a combinatorial circuit, and then the parties run a short protocol for every gate in the circuit.
While this approach is appealing in its generality and simplicity, the protocols it generates depend on the size of the circuit. This size depends on the size of the input (which might be huge as in a data mining application), and on the complexity of expressing f as a circuit (for example, a naive multiplication circuit is quadratic in the size of its inputs). It is observed that secure two-party computation of small circuits with small inputs may be practical.
Unfortunately, because clustering algorithms are imperfect, they do not neatly group all occurrences of each acronym into one cluster, nor do they allow users to issue follow-up queries that only return documents from the intended sense. An underappreciated aspect of clusters is their utility for eliminating groups of documents from consideration. The disadvantages of clustering include their lack of predictability; their conflation of many dimensions simultaneously, the difficulty of labeling the groups and the counter intuitiveness of cluster sub hierarchies.
The techniques using the support and confidence thresholds for hiding rules fail to hide all the sensitive rules. Even if they do so they do it in too much number of passes.

Critical Analysis of Existing Methods

From the discussions done in previous section following limitations have been identified in the existing approaches
1. Approach in [1] tries to hide every single rule without checking if rules can be pruned after some transactions have been changed.
2. Approach in [2] definitely hides all the rules which has sensitive items either in the left or in the right and for this it runs two different algorithms one if sensitive item is on the LHS and another is the sensitive item is on

the RHS i.e. it fails to hide all the rules containing sensitive item and takes more number of passes to prune all the rules containing sensitive items.

The Proposed Approach

This proposed work concentrates on the hiding sensitive rules by changing the support and the confidence of the association rule or frequent item set as data mining mainly deals with generation of association rules. Most of the work done by data miner revolves around association rules and their generation. As it is known that association rule is an important entity, which may cause harm to the confidential information of any business, defense organization and raises the need of hiding this information (in the form of association rules) that association amongst the data is what is understood by most of the data users so it becomes necessary to modify the data value(s) and relationships (Association Rules). Saygin et al [1] and Wang et al [2] have proposed some algorithms which help in reducing the support and the confidence of the rules. Hiding of association rules that expose the sensitive part of the data, researchers are bound to modify the data value(s) and relationships (Association Rules) because association amongst the data is what is understood by most of the data users. Changing the support or the confidence of the sensitive items existing in the database can modify data values. Many methods for hiding of association rules by changing the data values have been proposed in the literature [1, 2]. Existing approaches fail to hide all the rules, which contain sensitive items and even if they do so the number of passes required are many.

The proposed approach neither increases nor decreases the support of the sensitive items rather it just changes the position of the sensitive item in the database and results in hiding more number of association rules which contain sensitive items.

Hiding Association Rules Using R-Rules

The proposed approach selects all the association rules containing sensitive items either in the left or in the right from the set of all association rules generated from a dataset. Then these rules are represented in representative rules (RR) format with sensitive item on the LHS/RHS of the rules. Select a rule from the set of RR's, which has sensitive item on the LHS/RHS of the rule. Select a transaction that completely support RR i.e. it contains all the items in the RR. Now from this selected transaction delete the sensitive item and add the same sensitive item to a transaction which partially supports RR i.e. where items in RR are absent or only one of them is present.

Based on this a new algorithm for modifying database without changing the support of the sensitive item and still maintaining the secrecy of sensitive data has been proposed in the next Section.

Proposed Algorithm

This algorithm gives a modified dataset after distorting the database. Input to this algorithm is a Database, value of min_support, min_confidence, and a set of sensitive items. This algorithm computes the large item sets of all the sizes from the given dataset. Then it selects all the rules, which contain sensitive item from the association rules generated. The rules containing sensitive items are represented in the representative rules format and then the sensitive item is deleted from a transaction, which fully supports the selected RR and added to a transaction, which partially supports RR. The detailed pseudo-code for the algorithm is given below:

Algorithm: Hiding Of Sensitive Association Rules Using Support and Confidence

Input:

```
(1) D: A source database
(2) min_supp : A min_support.
(3) min_conf : A min_confidence.
(4) H: A set of sensitive items.
```

Output:

```
A transformed database D' where rules containing H on RHS/LHS will be hidden
1. Find all large item sets from D;
2. For each sensitive item h∈H {
3. If h is not a large item set then H=H- {h};
```

```
4. If H is empty then EXIT;
5. Select all the rules with min_supp containing h and store in U//h can either be on LHS or RHS
6. Repeat {
7. Select all the rules from U with same LHS
8. Join RHS of selected rules and store in R; //make representative rules
9. }Until (U is empty);
10. Sort R in descending order by the number of supported items;
11. Select a rule r from R
12. Compute confidence of rule r.
13. If conf>min_conf then   {//change the position of sensitive item h.
14. Find T1={t in D|t completely supports r ;
15. If t contains x and h then
16. Delete h from t
17. Find T1={t in D|t does not support LHS(r) and Partially supports x;
18. Add h to t
19. Repeat
20. {
21. Choose the first rule from U;
22. Compute confidence of r ;
23. } Until(U is empty);
24. }//end of if conf>min_conf
25. Else
26. Go to step 11;
27. Update D with new transaction t;
28. Remove h from H;
29. Go to step 2;
30. }//end of for each h∈H
```

A brief description of important steps of the algorithm is given below:
Step 5 of the proposed algorithm selects all the rules containing sensitive item(s) either in the left or in the right.
Steps 6 - 9 convert these rules in representative rules (RR) format.
Step 11 selects a rule from the set of RR's, which has sensitive item on the left of the RR is selected.
Step 13 - 18 deletes the sensitive item(s) from the transaction that completely supports the RR i.e. it contained all the items in of RR selected and add the same sensitive item to a transaction which partially supports RR i.e. where items in RR are absent or only one of them is present.
Step 20 – 24 recomputed the confidence of the rules in U.

The proposed algorithm can be illustrated with the following example for a given set of transactional data given in Table – 2.1.

Evaluation of Proposed Algorithm.

For the Dataset given in Table - 1.1 at a min_supp of 33% and a min_conf of 70 % and sensitive item H={C} we choose all the rules containing 'C' either in RHS or LHS and represent them in representative rule format. Out of the 8 association rules the rules containing sensitive items are 6 as shown in Table – 1.2:

Table.1 Transactional Dataset1

TID	ITEMS
T1	ABC
T2	ABCD
T3	BCE
T4	ACDE
T5	DE
T6	AB

Table.2 Sensitive association rules (w.r.t sensitive item C)

AR	SUPP	CONF
A => C	50	75
C => A	50	75
A, D => C	33.33	100
C, D => A	33.33	100
B => C	50	75
C => B	50	75

From this rules set select the rules that can be represented in the form of representative rules Like C=> A and C=> B can be represented as C=> AB Now delete C from a transaction where ABC all the three are present and add C to a transaction where A and B both are absent or only one of them is present. For this we change transaction T2 to ABD and transaction T5 to CDE. This results in changing the position of the sensitive item without changing its support. The dataset after this change becomes

Table.3 Modified Dataset1 for the proposed Approach (Sensitive Item – C)

TID	ITEMS
T1	ABC
T2	ABD
T3	BCE
T4	ACDE
T5	CDE
T6	AB

And the new set of association rules generated from this modified dataset is:

Table.4 Association rules remaining unhidden after modifying the Dataset1

AR	SUPP	CONF
B => A	50	75
A => B	50	75

i.e. all the rules of the original association rules set containing sensitive items on the LHS or on the RHS are hidden.

Table references: 1 and 2
min_supp = 33%, min_conf = 70%
Sensitive item = B
Similarly if H={B} i.e. if sensitive item is B then change transaction T2 to ACD and transaction T5 to BDE and the modified dataset is:

Table.5 Sensitive Association rules (w.r.t. Sensitive item B)

AR	SUPP	CONF
B => C	50	75
C => B	50	75
B => A	50	75
A => B	50	75

Modified dataset for the proposed approach is represented in table 6

Table.6 Modified Dataset1 for the proposed approach (Sensitive Item – B)

TID	ITEMS
T1	ABC
T2	ACD
T3	BCE
T4	ACDE
T5	BDE
T6	AB

Table.7 Association rules remaining unhidden after modifying the Dataset1

AR	SUPP	CONF
A => C	50	75
C => A	50	75
A, D => C	33.33	100
C, D => A	33.33	100

We see that all the rules of the original association rules set containing sensitive items on the LHS or on the RHS are hidden.

Above discussed Problem Evaluation shows that the proposed algorithm hides maximum number of rules containing sensitive item either in the LHS or in the RHS in minimum number of passes (all the problem evaluation results are for a single pass of existing algorithms and the proposed one).

Comparison with Existing Approaches

As discussed the approach used by Verykios et al. [1] tries to hide every single association rule without checking if some rules could be pruned out after some changes have been made in the database while hiding some rules previously.

If the number of association rules is too large then the number of passes taken by this approach is equal to the number of rules, which can be a great overhead for hiding algorithms.

Another approach proposed by Wang et al. [2], which tries to hide the rules on the basis of the sensitive item contained in a rule. If the sensitive item is on the LHS of the rule then it uses an algorithm, which increases the support of the sensitive item, and if the sensitive item is on the RHS of the rule then it uses another algorithm, which decreases the support of the sensitive item. In any case it only hides the rules, which has sensitive item either on the LHS or on the RHS of the rule.

The proposed approach uses an entirely different approach of not changing the support of the sensitive item. The proposed algorithm is run on the same dataset used by Verykios et al. [1] and Wang et al. [2] and hides almost all the rules, which contain sensitive items either in the left or in the right of the rule by not changing the support of the sensitive item. Some characteristics of the proposed algorithm and its comparison with the existing approaches are discussed next.

Characteristic of the Proposed Algorithm

First characteristic of Proposed Algorithm is that the support of the sensitive item is not changed. Thing which is changed is only the position of the sensitive item. It is seen from the above tabulated result that the Proposed Algorithm prunes more number of rules containing sensitive item in same number of DB scans. Graphical results of the above comparison where total No. of data base scanning is done on six transaction of different types where support and confidence are also different are shown in the Fig. 1.1 where total No. of prumes rules are represented. Below histogram graph shows that the number of rules hidden by Verykios et al. [1] approach for Dataset1 is 0. Using Wang et al [2]. Approach this number raises to 1 and the proposed approach hide 6 rules out of 8 rules containing sensitive item.

Table.8 Database for Dataset1 in Table.1 before and after hiding C and B

TID	ITEMS	D1(C sensitive)	D2(B sensitive)
T1	ABC	ABC	ABC
T2	ABCD	ABD	ACD
T3	BCE	BCE	BCE
T4	ACDE	ACDE	ACDE
T5	DE	CDE	BDE
T6	AB	AB	AB

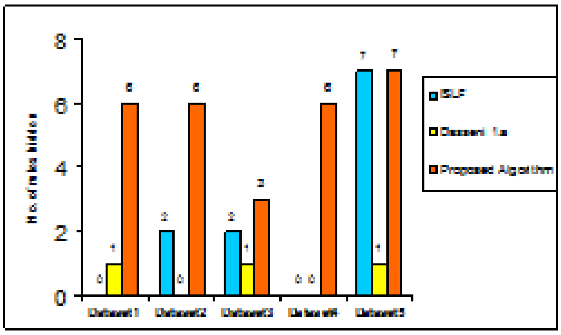

Fig.1 Performance Evaluation and Comparison of Existing and Proposed Algorithm

Conclusions

In this work, we have studied the database privacy problems caused by data mining technology and proposed e algorithm for hiding sensitive data in association rules mining. The proposed algorithm doesn't modifying the database transactions so that the support &confidence of the association rules remains unchanged. Examples demonstrating the proposed algorithms are shown. The efficiency of the proposed approach is shown in graph. It was shown that our approach required less number of database scanning and prune more number of hidden rules. However, our approach must hide all rules containing the hidden items

References:

[1] Vassilios S. Verykios,, Ahmed K. Elmagarmid , Elina Bertino, Yucel Saygin, Elena Dasseni. "Association Rule Hiding", IEEE Transactions on knowledge and data engineering, Vol.6, NO.4, April 2004

[2] Shyue-Liang Wang, Yu-Huei Lee, Billis S., Jafari, A. "Hiding sensitive items in privacy preserving association rule mining", IEEE International Conference on Systems, Man and Cybernetics, Volume 4, 10-13 Oct. 2004 .

[3] T. Dalenius and S.P. Reiss, "Data-Swapping:
A Technique for Disclosure Control",J. Statistical Planning and Inference, vol.6, pp 73-851982.

[4] L. Sweeney, "k-Anonymity: A Model for Protecting Privacy", Int'l J. Uncertainty, Fuzziness, and Knowledge-Based Systems, vol. 10, no. 5, pp. 557-570, 2002.

[5] K. Liu, H. Kargupta, J. Ryan, and K. Bhaduri, "Distributed Data Mining Bibliography", http://www.csee.umbc.edu/~hillol/DDMBIB/, 2004.

Author Details:

Prof Padam Gulwani Is Presently Serving As Head of the Department-School of Research & Technology, People's University, Bhopal.