RESEARCH ARTICLE OPEN ACCESS

Modified Approach for Hiding Sensitive Association Rules for Preserving Privacy in Database

Tania Banerjee, Esha Panse, Vinay Singh, Prasanna Kharche
Department of Computer Engineering Dr. D.Y Patil College of Engineering, Ambi, Pune, India.

Abstract
Data mining is the process of analyzing large database to find useful patterns. The term pattern refers to the items which are frequently occurring in set of transaction. The frequent patterns are used to find association between sets of item. The efficiency of mining association rules and confidentiality of association rule is becoming one of important area of knowledge discovery in database. This paper is organized into two sections. In the system Apriori algorithm is being presented that efficiently generates association rules. These reduces unnecessary database scan at time of forming frequent large item sets .We have tried to give contribution to improved Apriori algorithm by hiding sensitive association rules which are generated by applying improved Apriori algorithm on supermarket database. In this paper we have used novel approach that strategically modifies few transactions in transaction database to decrease support and confidence of sensitive rule without producing any side effects. Thus in the paper we have efficiently generated frequent item set sets by applying Improved Apriori algorithm and generated association rules by applying minimum support and minimum confidence and then we went one step further to identify sensitive rules and tried to hide them without any side effects to maintain integrity of data without generating spurious rules.

KEYWORDS: Association rule, confidence, Data mining methods and Algorithm, Minimum Support Threshold (MST), Minimum Confidence Threshold, (MCT), Rule hiding, Sensitive pattern, Sensitivity.

I. INTRODUCTION

Data mining is the process of analyzing large database to find useful patterns. The term pattern refers to the items which are frequently occurring in set of transaction. The frequent patterns are used to find association between sets of item. Association rule mining technique is widely used in data mining to find relationship between item sets. The efficiency of mining association rules and confidentiality of association rule is becoming one of important area of knowledge discovery in data base. However ,it breaks out many privacy issues. From a general point of view, we may classify privacy issues into two broad categories. The first is related to the data perse and is known as data hiding, while the second concerns the information ,or else the knowledge, that a data mining method may discover after having analyzed the data , and is known as knowledge .Data hiding tries to remove confidential or private information from the data before its disclosure . It is an important aspect in improving mining algorithm that deals with how to decrease item sets candidate in order to generate frequent item sets efficiently.

A. PROBLEM STATEMENT

The problem definition of our paper is

1. To hide sensitive rules without generating false rule and display only non-sensitive rules
2. To maintain privacy in database
3. To use ISR and DSL together so as to reduce the damage in database due to repeated sanitization.

Our paper also focuses towards making the code more optimized and for mining association rule we would use improved Apriori Algorithm given in [2].

The structure of this paper is as follows:
Section II describes the literature survey and theoretical data Section III includes MDSRRC algorithm with some more Efficient modification in detail with some Examples in detail Section IV includes the comparison DSRRC over MDSRRC Section V is conclusion

II. LITERATURE SURVEY AND THEORETICAL DATA

Abbreviation
D Original database
D' Sanitized database
R Association rules generated from original database
SR Sensitive association rules SR U R
MST Minimum support threshold

MCT Minimum confidence threshold
L.H.S Antecedent of an association rule

In [1], author proposed heuristic approach to prevent disclosure of sensitive patterns. They have mainly used two techniques : data distortion and data

blocking.[1] Data distortion is the process in which it changes the value of an item by a new value i.e decrease the confidence by altering '0' to '1' or '1' to '0'. [1]DATA Blocking instead of inserting or deleting item from database it replaces '1' or '0' with '?' in transaction.

Hiding sensitive rule is a subfield of privacy preserving data mining. It is divided into two categories- One is the preserving of data privacy, its goal is to blur the sensitive data without changing summary information .The other is the preserving of information privacy, its goal is to hide the sensitive information. A sampling approach makes a sample database such that the sensitive rules cannot be uncovered, while a modifying approach alters a few parts of the database to decrease the supports or confidences of the sensitive rules. Here, we adopt the modifying approach since it can retain the original data as much as possible. In the following, we focus the discussions on hiding sensitive rule. Researcher Atallah et al. refer to sensitive rule hiding as data sanitization, which aims at hiding a set of sensitive item sets, and prove that optimal sanitization is NP-Hard. Moreover, a heuristic approach is proposed using an item set graph to hide sensitive item sets in a one-by-one fashion. Author Oliveira and Zaı¨ne further address the efficiency and effectiveness issues. They plug a transaction retrieval engine to the hiding process and achieve a linear scalability in terms of database size. On effectiveness, they introduce three measures for hiding failures (sensitive patterns that are not hidden), missing costs (non-sensitive patterns falsely hidden), and artifactual patterns (spurious patterns falsely generated). Since these approaches only consider the decrease of supports, they may fail to hide a rule if the rule can be hidden only by decreasing the confidence. Researcher Saygin et al. argue that both the insertion and deletion of items will introduce false information and make it hard to determine whether the rules derived from the modified database can be trusted. Therefore, they propose a modification scheme to replace entries with unknowns. With this scheme, the item set support is represented as a range from the minimum support to the maximum support. Similarly, the confidence of a rule, say X -> Y , is also represented as a range from the minimum confidence (the minimum support of X -> Y divided by the maximum support of X) to the maximum confidence (the maximum support of X -> Y divided by the minimum support of X). By definition, a rule X ->Y is hidden if its minimum confidence is below MCT or the minimum support of X -> Y is below MST. From our view, the uncertainty due to the ranges on supports and confidences makes it hard to determine whether the derived rules can be trusted. Moreover, the side effects will be out of control since they do not consider the correlation among rules in their modification scheme. The work proposed by author Verykios et al. in should be the most relevant to our work. In that work, the authors propose five algorithms for rule hiding, which are also based on the decrease of supports and confidences. With their assumption, their algorithms can hide one rule at a time and decrease supports or confidences one unit at a time. Moreover, since the authors aim at hiding all sensitive rules instead of avoiding side effects, they do not consider the correlation among rules in their algorithms. Finally, to hide a rule, there can be a large number of "candidate" entries to modify. The "minimum impact" criterion used in the five algorithms only focuses on minimizing the number of modified entries. By contrast, in this system, we drop the assumption and decide the modification schemes and the entries to modify based on the correlation among three kinds of rules. Thus we are using APRIORI, MDSRRC, and ISR ,DSL n both by using both the algorithm of isl and dsr we can save the data base from get damage by constant sanitization.

Mining an association rule with support and confidence is defined as follow: let I= $\{i_1....i_N\}$ be distinct literals called items. Given a database D= {T1,...Tm} is a set of transaction where each transaction T is a set of items as Ti C I ($1 \leq i \leq m$). The association rule is define as $X \rightarrow Y$, where YCI, XCI and $Y \cap X = \emptyset$. X is called rule's antecedent (L.H.S) and Y is called rule's consequent (R.H.S). The support of rule $X \rightarrow Y$ is calculated using the following formula: $Support(X \rightarrow Y) = |XUY|/|D|$, where |D| define the total number of the transactions in the database D and |XUY| is the number of transactions which support item set XY. The confidence of rule is calculated using following formula: $Confidence (X \rightarrow Y) = |XUY| / |X|$, where |X| is number of transactions which support item set X. A rule $X \rightarrow Y$ is mined from database if support $(X \rightarrow Y) \geq$ MST (minimum support threshold) and confidence $(X \rightarrow Y) \geq$ MCT (minimum confidence threshold).

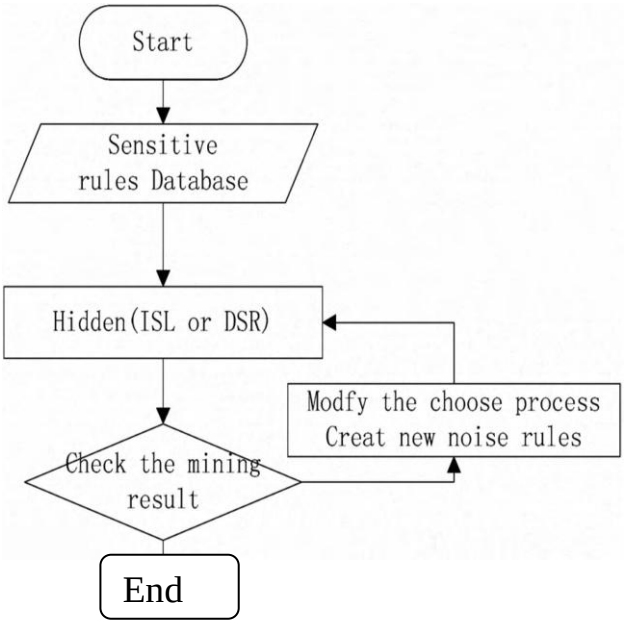

Figure 1: flow diagram.

In the ISL, the short item set is chosen as sacrifice item, because it has least item, negative impact cause by modify it may be the smallest. Our method first step is the same with ISL or DSR, then check the mining result, in this step, using privacy quantify to measure the result, if it's dissatisfied, return to change the choose policy. check the whole database, classify the items with their support, the items with support more close to Min_sup as "unsettled item sets" the other is "settled item sets", when choose the sacrifice item, the item in "settled item sets" will be chosen first.

III. THE MODIFIED ALGORITHM

The modified algorithm starts with mining the association rule from the original database D using association rule mining algorithm e.g. Apriori algorithm [5]. Then user specifies some rules as sensitive rules (SR) from the rules generated by the association rule mining algorithm. Then algorithm counts occurrences of each item in R.H.S of sensitive rules. Now algorithm finds IS= $\{is_0, is_1 \ldots is_k\}$ $k \leq n$, by arranging those items in decreasing order of their counts. After that sensitivity of each item is calculated then sensitivity of each transaction is calculated. Then transactions which support is_0 are sorted in descending order of their sensitivities. Now rule hiding process starts by selecting first transaction from the sorted transactions with higher sensitivity, delete item is_0 from that transaction. Then update support and confidence of all sensitive rules and if any rules have support and confidence below MST and MCT respectively then delete it from SR. Then update sensitivity of each item, transaction and IS. Again select transaction with higher sensitivity and delete is_0 from it. This process continues until all sensitive rules are hidden. As a result, modified transactions are updated in the original database and new database is generated which is called sanitized database D', which preserves the privacy of sensitive information and maintains database quality.

Proposed algorithm MDSRRC is shown below, which is used to hide the sensitive rules from database. Given a database D, MCT (minimum confidence threshold) and MST (minimum support threshold) algorithm generates sanitized database D'. Sanitized database hides all sensitive rules and maintains data quality.

MDSRRC Algorithm

INPUT:
MCT (Minimum Confidence Threshold), Original database D, MST (Minimum support threshold).
OUTPUT:
Database D' with all sensitive rules is hidden.

1. Apply Apriori algorithm [3] on given database D.
2. Generate all possible association rules R
3. Select set of rules SR⊆ R as sensitive rules.
4. Calculate sensitivity of each item j א D.
   Calculate sensitivity of each Transaction.
5. Count occurrences of each item in R.H.S of sensitive rules, find IS={is0, is1...isk} k≤n, by arranging those items in descending order of their count. If two items have same count then sort those in descending order of their actual support count
6. Select the transactions which supports is0, then sort them in descending order of their sensitivity. If two transactions have same sensitivity then sort those in increasing order of their length.
7. While(SR is not empty)
8. {
9. Start with first transaction from sorted transactions,
10. Delete item is0 from that transaction. 11. For each rule r א SR
11. {
13. Update support and confidence of the rule r.

14. If(support of r < MST or confidence of r < MCT)
15. {
16. Delete Rule r from SR.
17. Update sensitivity of each item.
18. Update IS (This may change is0).
19. Update the sensitivity of each transaction
20. Select the transactions which are supports is0,
21. Sort those in descending order of their sensitivity.
22. }
23. Else
24. {
25. Take next transaction from sorted transactions, go to step 10.
26. }
27. }
28. }
29. End

MDSRRC select best items so that deleting those items hide maximum rules from database to maintain data quality.

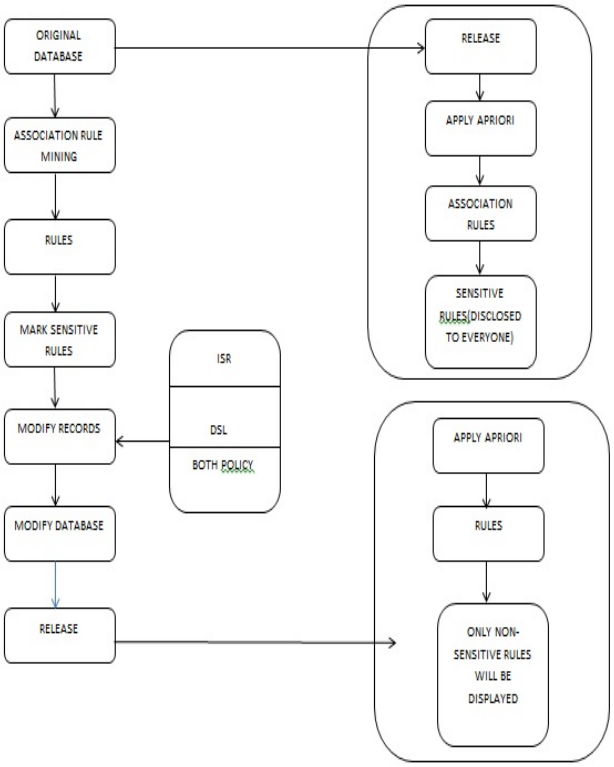

Figure 2: MDSRRC algorithm in flow diagram.

Algorithm

ISL Input:
(1) a source database D,
(2) a min_support,
(3) a min_confidence,
(4) a set of predicting items X Output: a transformed database D', where rules containing X on LHS will be hidden
1. Find large I-item sets from D;
2. For each predicting item x Є X
3. If x is not a large I-item set, then X = X-{x};
4. If X is empty then EXIT; II no rule contains X in LHS
5. Find large 2-item sets from D;
6. For each x Є X {
7. For each large 2-itemset containing x {
8. Compute confidence of rule U, where U is a rule like x-y;
9. If conf{U} < min_con , then
10. Go to next large 2-itemset;
11. Else {Increase Support of LHS
12. Find T L = {t in Dlt does not support U};
13. Sort TL in ascending order by the number of items;
14. While {conf(U) '?:. min _ conf and TL is not empty }{
15. Choose the first transaction t from TL;
16. Modify t to support x, the LHS(U);
17. Compute support and confidence of U;
18. Remove and save the first transaction t from TL;
19. }; II end While
20. }; II end if conf(U) < min_cmif
21. If TL is empty, then {
22. Cannot hide x-y;
23. Restore D;
24. Go to next large-2 itemset;
25. } II end if TL is empty
26. } II end of for each large 2-itemset
27. Remove x from X;
28. } II end of for each xE X
29. Output updated D, as the transformed D'

A. EXAMPLES

Example 1: Let a bag store that purchase luggage bag from two companies, XYZ and PQR, and both can access customers' database of the store. Now XYZ applies data mining techniques and mines association rules related to PQR's products. XYZ had found that most of the customer who buy luggage bag of the PQR also buy college bag. Now XYZ offers some discount on college bag if customer purchases XYZ's luggage bag. As result the business of PQR goes down. So releasing the database with sensitive information can cause the problem. This scenario gives the direction to research on sensitive rules (or knowledge) hiding in database.

Example 2:

TID	List of Items
1	Tea(I1),Sugar(I2),Toothbrush(I3),Toothpaste(I 4),Pen(I5),Refill(I6)
2	Toothbrush(I3),Toothpaste(I4)
3	Tea(I1),Sugar(I2),Maggie(I7)
4	Sugar(I2), Tea(I1),Coffee(I8)
5	Tea(I1), Coffee(I8)

The support of an association pattern is the percentage of task-relevant data transactions for which the pattern
is true.

$$Support(A => B) = \frac{\#\ tuples\ containing\ both\ A\ and\ B}{total\ \#\ of\ tuples}$$

Confidence is defined as the measure of certainty or trustworthiness associated with each discovered
pattern.

$$confidence\ (A => B) = \frac{\#\ tuples\ containing\ both\ A\ and\ B}{\#\ tuples\ containing\ A}$$

Item ID	Item	Suport
I1	Tea	4/8
I2	Sugar	3/8
~~I3~~	~~Toothbrush~~	~~2/8~~
~~I4~~	~~Toothpaste~~	~~2/8~~
~~I5~~	~~Pen~~	~~1/8~~
~~I6~~	~~Refill~~	~~1/8~~
~~I7~~	~~Maggie~~	~~1/8~~
~~I8~~	~~Coffee~~	~~2/8~~

Id	Item	Support
I1	Tea	4/8
I2	Sugar	3/8

Min_sup = 37.5%(3/8)

IV. COMPARISON OF MDSRRC AND DSRRC

Modified Decrease Support of R.H.S Item of Rule Clusters (MDSRRC) is an improved version of Decrease Support of R.H.S Item of Rule Clusters. This algorithm selects the items and transactions based on certain criteria which modify transactions to hide the sensitive information. DSRRC could not hide association rules with multiple items in antecedent (L.H.S) and consequent (R.H.S). To overcome this limitation, we have proposed an algorithm MSDRRC which uses count of items in consequent of the sensitive rules. MDSRRC modifies the minimum number of transaction to hide maximum sensitive rules and maintain data quality.

In the modified algorithm, we used heuristic approaches for hiding the sensitive rules these approaches are data distortion and data blocking. In data distortion we changed the item value with a new value, means it alter '0' to '1' or '1' to '0' for selected items to decrease the confidence. In data blocking instead of inserting or deleting items from the data base it simply replaces '1' and '0' with '?' in selected transactions.

After applying both Algorithms on sample database we have done evaluation by considering the performance parameters which are given in [13] viz. (a) HF (hiding failure): It is the percentage of the sensitive data that remain exposed in the sanitized dataset. (b) MC (misses cost): It is the percentage of the non-sensitive data that are hidden as a side- effect of the sanitization process. (c) AP (artificial patterns): It is the percentage of the discovered patterns that are artifacts. (d) DISS (dissimilarity): It is the difference between the original and the sanitized datasets. (e) SEF (side effect factor): It is the amount of non-sensitive association rules that are removed as an effect of the sanitization process. Experimental results show that MDSRRC increase efficiency and reduce modification of transactions in database. Performance comparison of MDSRRC with algorithm DSRRC is given in Table VI.

Parameter	DSRRC	MDSRRC
MC	36%	26.66%
DISS(D,D')	6.4%	5.4%
HF	0%	0%
SEF	36.5%	26.66%
AP	0%	0%

Table VI. Performance result

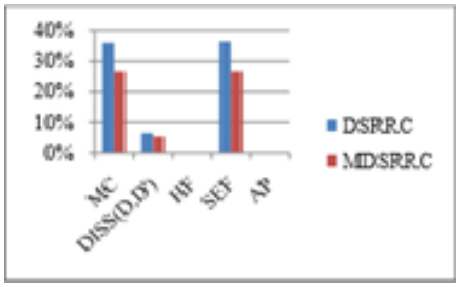

Fig. 2. Performance Comparison between DSRRC, MDSRRC

V. CONCLUSION

We proposed an algorithm named MDSRRC which hides sensitive association rules on data base to maintain data quality. The algorithm MDSRRC is an improved version of DSRRC .MDSRRC modifies minimum no transaction to hide maximum sensitive rules and improved data quality. We have used an improved Apriori algorithm which is used in data mining to extract data. The improved Apriori takes less time for generating frequent item sets. In future MDSRRC can be used further to increase the efficiency and reduce the side effects by minimizing modification on data base. In future the improved Apriori algorithm will be more efficient to generate strong rules while hiding the sensitive rules.

REFERENCES

[1] Nikunj H. Domadiya and Udai Pratap Rao, "Hiding Sensitive Association Rules to Maintain Privacy and Data Quality in Database"3rd IEEE International Advance Computing Conference (IACC) 2013.

[2] V. S. Verykios, A. K. Elmagarmid, E. Bertino, Y. Saygin, and E. Dasseni, "Association rule hiding," IEEE Transactions on Knowledge and Data Engineering, vol. 16, pp. 434–447, 2004.

[3] C. N. Modi, U. P. Rao, and D. R. Patel, "Maintaining privacy and data quality in privacy preserving association rule mining," 2010 Second International conference on Computing, Communication and Networking Technologies, pp. 1–6, Jul. 2010.

[4] J. Han, Data Mining: Concepts and Techniques. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2005

[5] Y.-H. Wu, C.-M. Chiang, and A. L. Chen, "Hiding sensitive association rules with limited side effects," IEEE Transactions on Knowledge and Data Engineering, vol. 19, pp. 29–42, 2007.

[6] S.-L. Wang, B. Parikh, and A. Jafari, "Hiding informative association rule sets," Expert Systems with Applications, vol. 33, no. 2, pp. 316 – 323, 2007.

[7] S.-L.Wang, D. Patel, A. Jafari, and T.-P. Hong, "Hiding collaborative recommendation association rules," Applied Intelligence, vol. 27, pp. 67–77, 2007.

[8] D.F. Gleich and L.-H. Lim. Rank aggregation via nuclear norm minimization. In KDD, 2011.

[9] Y. Saygin, V. S. Verykios, and A. K. Elmagarmid, "Privacy preserving association rule mining." in RIDE. IEEE Computer Society, 2002, pp 151–158.

[10] C. N. Modi, U. P. Rao, and D. R. Patel, "An Efficient Solution for Privacy Preserving Association Rule Mining," (IJCNS) International Journal of Computer and Network Security, vol. 2, no. 5, pp. 79–85, 2010.

[11] Wu and H. Wang, "Research on the privacy preserving algorithm of association rule mining in centralized database," in Proceedings of the 2008 International Symposiums on Information Processing, ser. ISIP'08. Washington, DC, USA: IEEE Computer Society, 2008, pp. 131–134

[12] V. Verykios and A. Gkoulalas-Divanis, A Survey of Association Rule Hiding Methods for Privacy, ser. Advances in Database Systems, C. [13] M. Atallah, A. Elmagarmid, M. Ibrahim, E. Bertino, and V. Verykios, "Disclosure limitation of sensitive rules," in Proceedings of the 1999 Workshop on Knowledge and Data Engineering Exchange, ser. KDEX '99. Washington, DC, USA: IEEE Computer Society, 1999, pp.