An Efficient Association Rule Hiding Algorithm for Privacy Preserving Data Mining

Yogendra Kumar Jain
Head of Computer Science & Engineering Department,
Samrat Ashok Technological Institute, Vidisha, (M.P.), India.
ykjain_p@yahoo.co.in

Vinod Kumar Yadav
Research Scholar M.Tech of Computer Science & Engineering
Samrat Ashok Technological Institute, Vidisha, (M.P.), India.
vinod.it210@gmail.com

Geetika S. Panday
A.P. of Computer Science & Engineering Department,
Samrat Ashok Technological Institute, Vidisha, (M.P.), India.
Geetika.silakari@gmail.com

Abstract-

The security of the large database that contains certain crucial information, it will become a serious issue when sharing data to the network against unauthorized access. Privacy preserving data mining is a new research trend in privacy data for data mining and statistical database. Association analysis is a powerful tool for discovering relationships which are hidden in large database. Association rules hiding algorithms get strong and efficient performance for protecting confidential and crucial data. Data modification and rule hiding is one of the most important approaches for secure data. The objective of the proposed Association rule hiding algorithm for privacy preserving data mining is to hide certain information so that they cannot be discovered through association rule mining algorithm. The main approached of association rule hiding algorithms to hide some generated association rules, by increase or decrease the support or the confidence of the rules. The association rule items whether in Left Hand Side (LHS) or Right Hand Side (RHS) of the generated rule, that cannot be deduced through association rule mining algorithms. The concept of Increase Support of Left Hand Side (ISL) algorithm is decrease the confidence of rule by increase the support value of LHS. It doesn't work for both side of rule; it works only for modification of LHS. In Decrease Support of Right Hand Side (DSR) algorithm, confidence of the rule decrease by decrease the support value of RHS. It works for the modification of RHS. We proposed a new algorithm solves the problem of them. That can increase and decrease the support of the LHS and RHS item of the rule correspondingly so that more rule hide less number of modification. The efficiency of the proposed algorithm is compared with ISL algorithms and DSR algorithms using real databases, on the basis of number of rules hide, CPU time and the number of modifies entries and got better results.

Keywords- Data Mining; Association Rules; Privacy Preserving Data Mining; Sensitive Items; Association Rule Hiding.

1. INTRODUCTION

Data mining is known as knowledge discovery process of analyzing data from different point of views and to work out into useful information which can be applied in various application [1], including advertisement, bioinformatics,

database marketing, fraud detection, e-commerce, health care, security, web, financial forecasting etc. The huge application of data mining technologies have raised concerns about securing information against unauthorized access is an important goal of database security and privacy. Privacy is a term which is associated with this a mining task so that we are able to a hide some crucial information which we don't want to disclose to the public. So the concept privacy preserving data mining is the process of preserving personal information from data mining algorithms. Privacy preserving data mining technique [2] is a new research area in data mining and statistical databases where mining algorithms are analyzed for the side effect they acquire in data privacy. The goal of privacy preserving data mining is to developed algorithms [3][4] for modifying the original data in some way, so that the private data and private knowledge remain private even after the mining process. There have been two types of privacy concerning data mining. The first type of privacy, called output privacy, is that is minimally altered so that the mining result will preserve certain privacy. The second type of privacy, input privacy, is that the data is manipulated so that mining result is not affected or minimally affected.

The problem for finding most favorable purification of a database against association rule analysis was introduced in [8]. The research can be divided into hiding sensitive rules and sensitive items. Vassilios S. Verykios et al. [5] conducted through research and introduce five algorithms for hiding sensitive rules. They conclude that among the proposed algorithms there is not a best solution for all the data table, including the execution time and the side effects produce by the proposed algorithms. Shyue Liang Wang [20] proposed algorithms to hide sensitive items instead of hiding sensitive association rules. The algorithm needs less number of database scans but the side effects generated are also high. Ali Amiri [19] proposed heuristic algorithms to hide sensitive items, while maximizing data utility at the expense of computational efficiency. Yi-Hung Wu et al. [9] proposed a heuristic method that could hide sensitive association rules with limited side effects. It also spent more time on comparing and hidden rules.

For association rules hiding, two basic approaches have been proposed. The first approach [5] hides one rule at a time. First selects transactions that contain the items in a give rule. It then tries to modify transaction by transaction until the confidence or support of the rule fall below minimum confidence or minimum support. The modification is done by either removing items from the transaction or inserting new items to the transactions. The second approach [6] deals with groups of restricted patterns or association rules at a time. It first selects the transactions that contain the intersecting patterns of a group of restricted patterns. Depending on the disclosure threshold given by users, it sanitizes a percentage of the selected transactions in order to hide the restricted patterns.

In our work we are concern of hiding certain association rules which contain some sensitive information which are on the Right hand side or left hand side of the rule, so that rules containing confidential item can't be reveal. Our approached is based on modifying the database in a way that confidence of the association rule can be reduce with the help increase or decrease the support value of RHS or LHS correspondingly. As the confidence of the rule is reduce below a specified threshold, it is hidden or we can say it will not be disclosed.

Our method is based on [7] proposed two algorithms namely ISL (Increase Support of Left hand side) and DSR (Decrease Support of Right hand side) to hide useful association rule from transactions data with binary attributes. In ISL method, confidence of a rule is decreased by increasing the support value of Left Hand Side (L.H.S.) of the rule. For this purpose, only the items from L.H.S. of a rule are chosen for modification. In DSR method, confidence of a rule is decreased by decreasing the support value of Right Hand Side (R.H.S.) of a rule. For this purpose, only the items from R.H.S. of a rule are chosen for modification.

The reminder of this paper is organized as follows. Section 2 presents the statement of the problem and the notation used in the paper. Section 3 presents the proposed algorithms for hiding informative association rule. Section 4 shows example of the proposed algorithms. Section 5 shows the experimental results of the proposed algorithms. Section 6 shows the analysis of the proposed algorithm. Concluding remarks and future works are described in Section 7.

2. PROBLEM DESCRIPTIONS

The goal of data mining is to extract hidden or useful unknown interesting rules or patterns from databases. However, the objective of privacy preserving data mining is to hide certain confidential data so that they cannot be discovered through data mining techniques. In this work, we assume that only sensitive items are given and propose one algorithm to modify data in database so that sensitive items cannot be deduced through association rules mining algorithms. More specifically, given a transaction database D, a minimum support, a minimum confidence and a set of items H to be hidden, the objective is to modify the database D such that no association rules containing H on the right hand side or left hand side will be discovered.

2.1 Theoretical Background and Related Work

The problem of mining association rules was introduced in [8]. The problem of mining association rules is to find all rules that are greater than the user-specified minimum support threshold and minimum confidence threshold. Association rule using support and confidence can be defined as follows. Let $I = \{i_1, i_{2...} i_n\}$ be a set of literals, called items. Database $D= \{T_1, T_{2,}T_{3,...,}T_n)$ is a set of transactions, where each transaction T is a set of items such that $T \subset I$, an association rule is an expression, $X \rightarrow Y$ where $X \subset I, Y \subset I$ and $X \cap Y = \emptyset$. The X and Y are called correspondingly 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 milk also buys bread. The 90% here is called the confidence of the rule, which means that 90% of transaction that contains X also contains Y. The confidence c is calculated as $|X \cup Y| \div |X| \geq c$. The support s of the rule is the percentages of transactions that contain both X and Y, which is calculated as $|X \cup Y| \div |D| \geq s$. In other words, the confidence of a rule measures the degree of the correlation between item sets, while the support of a rule measures the significance of the correlation between item sets. We consider user specified thresholds for support and confidence, MST (minimum support threshold) and MCT (minimum confidence threshold).

There are many approaches have been proposed to preserve privacy for crucial knowledge or sensitive association rules in database. They can be classified in to following classes: Heuristic based, these approaches can be further divided in to two groups based on data modification techniques: data distortion techniques and data blocking techniques. Data distortion techniques try to hide association rules by decreasing or increasing support. To increase or decrease support, they replace 0's by 1's or vice versa in selected transactions. So they can be used to address the complication issue. But they produce undesirable side effects in the new database, which lead them to suboptimal solution [9]. The method of reduce the side effects in sanitized database, which are produced by other approaches [10]. An efficient clustering based approach [11] to reduce the time complexity of the hiding process. Data blocking techniques replace the 0 and 1 by unknowns "?" in selected transaction instead of inserting or deleting items. So it is difficult for an opponent to know the value behind "?". First introduce blocking based technique [12] for sensitive rule hiding. Border based approaches, these use the notion of borders introduced in [13]. These approaches preprocess the sensitive rules so that minimum numbers of rules are given as input to hiding process. So, they maintain database quality while minimizing side effects. Hiding process in greedily [14] selects those modifications that lead to minimal side effects. Reconstruction based approaches generate [15] privacy aware database by extracting sensitive characteristics from the original database. These approaches generate minor side effects in database than heuristic approaches. Mielikainen [16] was the first analyzed the computational complexity of inverse frequent set mining and showed in many cases the problems are computationally difficult. Cryptography based approaches used in multiparty computation. If the database of one organization is distributed among several sites, then secure computation is needed between them. These approaches encrypt original database instead of distorting it for sharing. So they provide input privacy. Vaidya and Clifton [17] proposed a secure approach for sharing association rules when data are vertically partitioned. The secure mining of association rules over horizontal partitioned data.

Many researchers have worked on the basis of reducing the support and confidence of sensitive association rule. ISL and DSR are the common approaches used to hide the sensitive rules. Some of the researchers have used data perturbation techniques to modify the confidential data value in such a way that the approximant data mining results could be obtained from the modified version of the database. Our work also has the basis of reduction of confidence using increase or decrease support value of generated sensitive rule.

3. PROPOSED ALGORITHM

In order to hide an association rule, $X \rightarrow Y$, we can either decrease its support or its confidence to be smaller than user-specified minimum support transaction (MST) and minimum confidence transaction (MCT). To decrease the confidence of a rule, we can either (1) increase the support o of X, the left hand side of the rule, but not support of $X \cup Y$, or (2) decrease the support of the item set $X \cup Y$. For the second case, if we only decrease the support of Y, the right hand side of the rule, it would reduce the confidence faster than simply reducing the support of $X \cup Y$. To decrease support of an item, we will modify one item at a time by changing from 1 to 0 or from 0 to 1 in a selected transaction.

Based on these two concepts, we propose a new association rule hiding algorithm for hiding sensitive items in association rules. In our algorithm, a rule $X \rightarrow Y$ is hidden by decreasing the support value of $X \cup Y$ and increasing the support value of X. That can increase and decrease the support of the LHS and RHS item of the rule

correspondingly. This algorithm first tries to hide the rules in which item to be hidden i.e., X is in right hand side and then tries to hide the rules in which X is in left hand side. For this algorithm t is a transaction, T is a set of transactions, R is used for rule, RHS (R) is Right Hand Side of rule R, LHS (R) is the left hand side of the rule R, Confidence (R) is the confidence of the rule R, a set of items H to be hidden.

ALGORITHM:

INPUT**:** A source database D, A minimum support min_support (MST), a minimum confidence min_confidence (MCT), a set of hidden items X.

OUTPUT**:** The sanitized database D, where rules containing X on Left Hand Side (LHS) or Right Hand Side (RHS) will be hidden.

Steps of algorithm:

1. Begin
2. Generate all possible rule from given items X;
3. Compute confidence of all the rules for each hidden item H, compute confidence of rule R.
4. For each rule R in which H is in RHS

4.1 If confidence (R) < MCT, then
Go to next 2-itemset;
Else go to step 5

5. Decrease Support of RHS item H.

5.1 Find T=t in D fully support R;
5.2 While (T is not empty)
5.3 Choose the first transaction t from T;
5.4 Modify t by putting 0 instead of 1 for RHS item;
5.5 Remove and save the first transaction t from T; End While

6. Compute confidence of R;
7. If T is empty, then H cannot be hidden;
8. For each rule R in which is in LHS
9. Increase Support of LHS;
10. Find T=t in D| t does not support R;
11. While (T is not empty)
12. Modify t by putting 1 instead of 0 for LHS item;
13. Remove and save the first transaction t from T; End While
14. Compute confidence of R;
15. If T is empty, then H cannot be hidden;
End For;
End Else;
End For;
16. Output update D, as the transformed D;

4. EXAMPLE

This section shows an example of the proposed algorithm in hiding sensitive item in association rule mining. Consider Table 1 as a database, MST=33%, MCT=70%, each element has value 1 if the corresponding item is supported by the transaction and 0 otherwise. Size means the number of elements in the list having value 1.

Table 1: Database D using specified notation

Tid	Items	ABC	Size
T1	ABC	111	3
T2	ABC	111	3
T3	ABC	111	3
T4	AB	110	2
T5	A	100	1
T6	AC	101	2

The all possible rules with confidence are: A→B (66.6%), A→C (66.6%), B→A (100%), B→C (75%), C→A (100%), C→B (75%). Suppose we first want to hide item A, first take rule in which A is in RHS. These rules are B→A and C→A both has greater confidence from MCT. First take rule B→A search for transaction which support both B and A, B=A=1. There are four transactions T1, T2, T3, T4 with A=B=1. Now update table put 0 for item A in all four transactions. Now calculate confidence of B→A, it is 0% which is less than MCT so now this rule is hidden. Now take rule C→A, search for transaction in which A=C=1, only transaction T6 has A=C=1, update transaction by putting 0 instead 1 in place of A. Now take the rules in which A is in LHS. There are two rules A→B and A→C but both rules have confidence less than MCT so there is no need to hide these rules. So Table 2 shows the modified database after hiding item A.

Table 2: Update table after hiding item A

Tid	ABC	Size
T1	011	2
T2	011	2
T3	011	2
T4	010	2
T5	100	1
T6	101	2

5. RESULTS

We have performed all experiments on a PC with Pentium IV and 512 MB RAM, under the Windows XP. In this work we used database "Breast Cancer", dataset from UCI Machine Learning Repository [18]. We performed four different experiments to compare the performance of proposed algorithm with ISL and DSR algorithm. For each data set, various sets of association rules are generated under various minimum supports and minimum confidences. The minimum support range is from 10% to 30%. The minimum confidence range is from 40% to 70%. The first experiment finds the relationship between CPU time and No. of modifies entries, and number of transactions. Table 3 present the experimental results. In this experiment, the minimum confidence value is set 70% and minimum support values are taken as 10, 20, and 30 for 500, 1000 and 1500 (T) transaction respectively.

Table 3: The experiment results of ISL, DSR and Proposed Algorithms for MCT=70%.

T	CPU Time (milliseconds)			No. of Modifies Entries		
	ISL	***DSR***	***Proposed Algorithm***	***ISL***	***DSR***	***Proposed Algorithm***
500	531	344	47	173	511	182
1000	750	1234	110	413	1072	379
1500	1328	1985	203	621	1454	550

Table 4 present the experimental results of the second experiment, finds the number hidden rules for above minimum support and transactions.

Table 4: The experiment results of Hiding Rules.

T	Minimum Support	No. of Rule Hide		
		ISL	***DSR***	***Proposed***
500	10	2	7	10
1000	20	2	10	11
1500	30	2	11	12

Table 5 present the experimental results of the third experiment, finds the relationship between CPU time and No. of modifies entries, and number of transactions. In this experiment, the minimum support value is set 30% and minimum confidence values are taken as 40, 50, and 60 for 500, 1000 and 1500 transaction respectively.

Table 5: The experiment results of ISL, DSR and Proposed Algorithms for MCT=70%.

T	**CPU Time (milliseconds)**			**No. of Modifies Entries**		
	ISL	***DSR***	***Proposed Algorithm***	***ISL***	***DSR***	***Proposed Algorithm***
500	453	1188	62	440	1606	377
1000	1063	2469	156	968	2141	627
1500	1313	2097	141	641	1805	581

Table 6 present the experimental results of the fourth experiment, finds the number hidden rules for above minimum confidence and transactions.

Table 6: The experiment results of Hiding Rules.

T	**Minimum Confidence**	**No. of Rule Hide**		
		ISL	***DSR***	***Proposed***
500	40	0	1	9
1000	50	1	5	10
1500	60	2	11	12

6. ANALYSIS

This section analyses some of the characteristics of the proposed algorithm based on our experimental results and compare with the previous work [7]. The first characteristic we observe the total number of rules hidden for different values of support and confidence. Table 4 shows the relationship between number of hidden rules and number of transactions, and shows the relationship between the numbers of hidden rules for different values (10, 20, and 30) of minimum support. The Table 6 shows the relationship between the numbers of hidden rules for different values (40, 50, and 60) of minimum confidence. The numbers of hidden rules increase quickly with increase in minimum confidence value because only a few transactions need to be modified to lower the confidence of the rule for higher minimum confidence value. From this experiment results, it can be easily seen that our algorithm hides more rules in comparison to previous work for different value of minimum support and minimum confidence value.

From these experimental results, it can be easily seen that our algorithm hides more rules in comparison to previous work for different user specified value of MST and MCT. The reason is that in our algorithm, a rule $X \rightarrow Y$ is hidden by decreasing the support value of $X \cup Y$ and increasing the support value of X. That can increase and decrease the support of the LHS and RHS item of the rule correspondingly. In contrast, in previous work, ISL algorithm a rule $X \rightarrow Y$ is hidden by increase the support value of X, the left hand side of the rule but not support count $X \cup Y$. In DSR algorithm a rule $X \rightarrow Y$ is hidden by decrease the support count of the itemset $X \cup Y$ in the transactions contain both X and Y, if we decrease the support value of Y only, the right hand side of the rule. Also, the condition used by previous work allows only a small number of transactions to be modified for the rule under hidden. Therefore, our algorithm hides more number of rules in comparison to previous work.

The second characteristic we observe the database effects. Table 3 shows the relationship between total number entries modified and number of transaction, different values (10, 20, and 30) of minimum support. The Table 5 shows the relationship between total number entries modified for different values (40, 50, and 60) of minimum confidence and number of transaction. Our algorithm modifies a few numbers of entries for hiding a given set of rules in all the datasets.

The last characteristic we observe is the CPU time requirement. Table 3 shows the relationship between total CPU time for number entries modified and number of transaction, different values (10, 20, and 30) of minimum support. The Table 5 shows the relationship between total CPU time for number entries modified and number of transaction, different values (40, 50, and 60) of minimum confidence. Our algorithm modifies a few CPU time for hiding rule and modifies entries a given set of rules in all the datasets.

7. CONCLUSION

The purpose of the Association rule hiding algorithm for privacy preserving data mining is to hide certain crucial information so they cannot discovered through association rule. In this paper, we have proposed an efficient Association rule hiding algorithm for privacy preserving data mining. This is based on association rule hiding approach of previous algorithms and modifying the database transactions so that the confidence of the association rule can be reduce. In our proposed algorithm we can hide the generated crucial association rule on the both side (LHS and RHS) correspondingly, so it reduce the number of modification, hide more rule in less time. The efficiency of the proposed algorithm is compared with ISL and DSR approach. Our algorithm prunes more number of hidden rules with same number of transactions scanned, less CPU time and modification. In future work, we will continue to improve the efficiency of the algorithm by reducing the number of database and develop an integrated secure association rule mining tool which protects data from unamortized access. Extend our research to other task of data mining like clustering, fuzzy set, classification etc.

REFERENCES

[1] Razali, A.M. and S. Ali, "Generating treatment plan in medicine: A data mining approach". Am. J. Applied Sci., 6: pp. 345-351, 2009.

[2] Evfimievski, A., R. Srikant, R. Agrawal and J. Gehrke. "Privacy preserving mining of association rules". Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, July 23-25, ACM Press, Edmonton, AB., Canada, pp. 1-12, 2002.

[3] Saygin, Y., V.S. Verykios and A.K. Elmagarmid. "Privacy preserving association rule mining". Proceedings of the 12th International Workshop on Research Issues in Data Engineering: Engineering E-Commerce/E-Business Systems, Feb. 24-25, IEEE Xplore Press, San Jose, CA. USA., pp. 151-158, 2002.

[4] Vaidya, J., H. Yu and X. Jiang. "Privacy preserving SVM classification". Knowl. Inform. Syst., pp. 161-178, 2008.

[5] Verykios, V.S., A.K. Elmagarmid, E. Bertino, Y. Saygin and E. Dasseni. "Association rule hiding". IEEE Trans. Knowl. Data Eng., 16:pp. 434-447., 2004.

[6] Oliveira, S., & Zaiane, O. "Privacy preserving frequent itemset mining". In Proceedings of IEEE international conference on data mining, November pp. 43–54, 2002.

[7] Wang, S.L., B. Parikh and A. Jafari." Hiding informative association rule sets". Exp. Syst. Appli., 33: pp. 316-323, 2007. M. Atallah, E. Bertino, A. Elmagarmid, M. Ibrahim, and V. S. Verykios. Disclosure limitation of sensitive rules, In P*roc.* of the 1999 IEEE Knowledge and Data Engineering Exchange Workshop, pp. 45–52, 1999.

[8] Y. H. Wu, C.M. Chiang and A.L.P. Chen. "Hiding Sensitive Association Rules with Limited Side Effects", IEEE Transactions on Knowledge and Data Engineering, vol.19 (1), pp. 29–42, Jan. 2007.

[9] 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(4), pp. 434–447, April 2004.

[10] K. Duraiswamy, and D. Manjula, "Advanced Approach in Sensitive Rule Hiding", Modern Applied Science, vol. 3(2), Feb. 2009.

[11] Y. Saygin, V. S. Verykios, and A. K. Elmagarmid, "Privacy preserving association rule mining", In Proc. Int'l Workshop on Research Issues in Data Engineering (RIDE 2002*)*, pp. 151–163, 2002.

[12] H. Mannila and H. Toivonen,. "Level wise search and borders of theories in knowledge discovery, Data mining and Knowledge Discovery", vol.1 (3), pp. 241–258, Sep. 1997.

[13] X. Sun and P.S. Yu, "A Border-Based Approach for Hiding Sensitive Frequent Itemsets", In Proc. Fifth IEEE Int'l Conf. Data Mining (ICDM '05), pp. 426–433, Nov. 2005.

[14] T. Mielikainen, "On inverse frequent set mining," In Proc. 3rd *IEEE* ICDM Workshop on Privacy Preserving Data Mining. IEEE Computer Society, pp.18–23, 2003.

[15] Y. Guo, "Reconstruction-Based Association Rule Hiding," In Proc. Of SIGMOD2007 Ph.D. Workshop on Innovative Database Research 2007(IDAR2007*)*, June 2007.

[16] J. Vaidya and C. Clifton, "Privacy preserving association rule mining in vertically partitioned data," In proc. Int'l Conf. Knowledge Discovery and Data Mining, pp. 639–644, July 2002.

[17] Experiment performed used database from this url, http://www.ics.uci.edu//~mlearn/ML.

[18] Shyue-Liang Wang, "Hiding sensitive predictive association rules", Systems, Man and Cybernetics, 2005 IEEE International conference on Information Reuse and Integration, vol.1, pp.164-169,2005.

[19] Ali Amiri, "Dare to share: Protecting sensitive knowledge with data sanitization", Decision Support System archive vol. 43, issue 1, pp.181-191, 2007.

[20] R. Agrawal and R. Srikant, " Privacy preserving data mining ", In ACM SIGMOD Conference on Management of Data pages 439-450,Dallas, Texas, May 2000.