International Journal of Application or Innovation in Engineering & Management (IJAIEM)
Web Site: www.ijaiem.org Email: editor@ijaiem.org
Volume 3, Issue 12, December 2014 **ISSN 2319 - 4847**

A Survey on the Privacy Preserving Algorithms of Association Rule Hiding

Mohammadreza Lorestani[1], EynollahKhanjari Miyaneh[2]

[1]M.Sc Student at School of Computer Engineering, Iran University of Science and Technology, Iran, Tehran

[2]Assistant Professor School of Computer Engineering, Iran University of Science and Technology, Iran, Tehran

ABSTRACT

Following the ever-increasingly pace ofgrowth of the Internet, data storage devices, and data processing technologies, privacy preserving has emerged as one of the paramount issues in data mining. The issue has been widely studied given increase of sensitive data on the Internet. People are growingly expressing concerns about their privacy and personal information. To deal with this new demand, privacy preserving data mining (PPDM) has become increasingly popular because it allows sharing of privacy data for analysis purposes. However, there is a natural tradeoff between privacy and accuracy, though this tradeoff is affected by the particular algorithm which is used for privacy preservation. Association rule miningis oneofthe most widely useddata miningtechniques thus this paper focuses on privacy preserving algorithms of association rulesmining.

Keywords*:* privacy preserving data mining (PPDM), association rule hiding, hiding failure, loss of non-sensitive rules, ghost rules

1. INTRODUCTION

Recent development in data mining and knowledge discovery have had notable effect on scientific and technologicalfields. On one hand, data mining techniques enable us to analyze great deal of data in shorter time and on the other hand, many data mining algorithms implemented on the data may be featured with breach of privacy of the users. Privacy preserving data mining was developed in response to this [1].Association rule is a data mining technique, which can detect relationship among variety of data of great volume. The technique, however, may result in private and organizational data leak[2]. Association rule hiding was developed to solve the problem. Over the years, many algorithms have been proposed in this regard and here we try to discuss them. The rest of the paper is organized as follows. Section 2 introduces association rule hiding strategy and purposes and strategies association rule hiding are discussed in section 3. Section 4 introduced the algorithms of association rule hiding algorithms and section 5 represent tests results and the study concludes with conclusion in section 6.

2. Association Rule Strategy

Extraction of association rule is a data mining operation which searches for association between attributes in datasets. Another name of this method is market basket analysis. It's aim is to extract rules to quantify relationship between two or more attributes. Association rules are defined by *if-then* structure with support and confidence measures[3].

An association rule finding problem is defined as follows:

- Set of item: $I = \{I_1, I_2, I_3, \dots, I_m\}$
  Where, T stands for a set of I known as transaction; D denotes set of transactions in T; and TID stands for a unique index assigned to each transaction.
- In general, an association rule looks as equation (1):
  $$X \rightarrow Y[Support, \text{Confidence}] \quad (1)$$
  According to equation (2):
  $$X \subset I, Y \subset I \ , X \cap Y = \emptyset \quad (2)$$
  Support (X, Y): represent rate or number of transactions D including X and Y.
  Confidence: indicates dependency of one item to another item, which is obtained as follows:
  $$Confidence\ (X \rightarrow Y) = \frac{Support\ (X \cup Y)}{Support(X)} \quad (3)$$
  This index yields dependency between the sets $X$and$Y$, which is used to measure power of a rule. Generally, rule with higher confidence index are preferred[3].

3. Purpose of Association Rules Hiding Strategies

Association rules hiding strategies try to meet the following goals:

- Hiding failure: the rate of missed sensitive rule in the algorithm, which are discoverable in the new data base. According to below equation (4):

$$HF = \frac{|RS(D')|}{|RS(D)|} \quad (4)$$

Where, $|RS(D)|$and $|RS(D')|$indicate number of sensitive rules in transaction database D and D' respectively [4].

- Lost rule rate: rate of non-sensitive rules in$D$, which are hidden by hiding algorithm. According to below equation (5):

$$LR = \frac{|RNS(D)-RNS(D')|}{|RNS(D)|} \quad (5)$$

Where, $|RNS(D)|$ and $|RNS(D')|$ represent number of non-sensitive rules in D and D' database [4].

- Ghost rule: usually, implementation of hiding algorithm generate new rules in the database, which are known as ghost rule. The fewer the ghost rules, the more successful the algorithm. According to below equation (6):

$$GR = \frac{(|R'|-|R\cap R'|)}{|R'|} \quad (6)$$

Where, $|R|$indicate total number of rules in the database D and $|R'|$ indicates total rules in D'[4].

Relationship between transaction database before and after modification is pictured in Figure1. In the figure, R indicates extracted association rule from the main database, $R_R$indicates association rules of modified database, and$\sim R_R$ indicates non-sensitive association rule of the main database. Regions 1, 2 and 3 refer to Hiding Failure, Lost rule and Ghost rule, respectively.

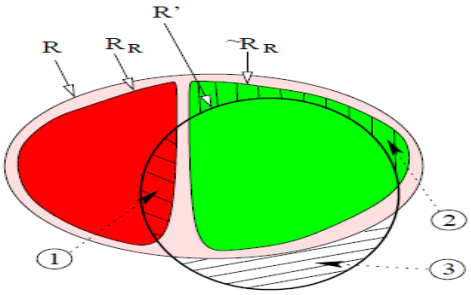

Figure1. Purposes of association rule hiding strategies on main and modified databases[4]

The three goals noted above are the measures to survey the algorithm as to association rule hiding[5]. The following section discusses algorithms based on these measures. The problem of finding a modified database in which the three goals are met is a NP-Hard Problem [6].

4. Association Rule Hiding Algorithms

There are three types of association rule hiding algorithms demonstrated as follow:

4.1 Heuristic Algorithms

The employed strategies include efficient, fast, and scalable algorithms that try to hide the sensitive rules by sampling a set of transactions of the database. Given efficiency and scalability of the strategies, authors show great interest in these types of algorithms [2].Increment Support of Left hand side (ISL) algorithm is introduced in[7]. The algorithm receives a set of sensitive items as inputs and after implementation of the algorithm, the items on the left would have no rule. Sensitive transaction refers to transactions that do not include the sensitive items with minimum number of items. Assume the rule $X \rightarrow Y$ denotes a sensitive rule, the algorithm increases support of X taking into account definition of confident of rule with respect to (3), and consequently confidence of the rule decrease. The algorithm is featured with 12.5% hiding failure and 33% ghost rules, while no non-sensitive rule is hidden [8].Another algorithm known as Decrease Support of Right hand side (DSR) is proposed in[7]. The algorithm reduces support of right hand of sensitive rule, and consequently reduces support of the rule below minimum support threshold. Sensitive transaction refers to those including items of sensitive rule and minimum number of items. Afterward, items of right hand side rule are removed from the transaction. DSR algorithm can hide all sensitive rules by generating 5% ghost rules and hiding 11% of non-sensitive rules; it outperforms ISL [8].DCIS algorithm [9]takes a set of items to be hidden as input and hides the rules of which the items are on the right side. Here, the sensitive transaction is the one that does not include the items on the right and left sides with minimum number of items. By adding the items of the left side of the rule to the sensitive selected transaction, the algorithm tires to reduce rule of confidence below minimum confidence threshold. The algorithm sets the number of transactions to be modified for each sensitive rule. It is successful in hiding sensitive rules and does not hide non-sensitive rules, still, it generate 75% ghost rule [8].DCDS algorithm [9]takes a set of items as input and tries to hide all rules of which the items are on the right side. By reducing support of the right side of sensitive rules, the algorithm reduces confidence of the rule. Take rule$X \rightarrow Y$; sensitive transaction in the algorithm is the one that itincludes items X and Y and fewer items at the same time. Then, the items Y are removed from the sensitive transaction. The algorithm hides all sensitive rules, generates 1% ghost rules and hides 4% of non-sensitiverules [8].Decrease Support and Confidence (DSC) algorithm is introduced in[10]. It is

featured with scanning the database once using the algorithm proposed in [11]. The algorithm takes a set of items as input, which is used to hide sensitive rules to which the items are on the left side. The sensitive transaction is defined as the transaction with items on the left and right sides and minimum number of items. Items of the right hand are removed from the algorithm. For each sensitive rule, number of transactions to be modified must be set by the operator. The algorithm generates 4% ghost rules and hides 9% of the non-sensitive rules; however, it is completely successful in hiding sensitive rules [8].Different algorithms are introduced in [12]. Transactions that cover the sensitive rules are detected and then transactions with maximum sensitive rules (higher conflict degree) are selected as sensitive transactions and sorted based on conflict degree in descending order (the transaction list is used for the next sensitive rule without updating) and μ,based on threshold disclosure, determines how many sensitive transactions must be modified. Afterward, victim item is set based on type of algorithm; for instance in Minimum Frequency Item Algorithm (MINFIA) the victim item is the least frequent item in the transaction. On the other hand, Maximum Frequency Item Algorithm (MAXFIA) adopts the item with highest frequency in the transaction as the victim item. Item Grouping Algorithm (IGA) places rules with shared items in one group. In this way, modifying shared items hides more than one rule at a time and fewer changes are implemented on the database. With threshold disclosure equal with zero, hiding performed by the three algorithm will be perfect, while MINFIA, MAXIFA, and IGA hide 65%, 69%, and 44% non-sensitive rules respectively. Nohiding is performed when threshold disclosure is set to 1, while no non-sensitive rule is hidden.

Algorithms proposed in [13]adopts different assumptions for hiding. Some ofthese assumptions are no share items among the sensitive rules, carrying out only support or confidence decrease at a time, and scanning the database for each sensitive rule. These algorithms are discussed in what follows.

Algorithm (1.a) uses increase of support of left had for hiding sensitive rule and keeps increasing the support until confident rule reaches minimum confidence threshold. The algorithm differs DCIS in selecting sensitive transaction. Sensitive transaction of $X \rightarrow Y$ in (1.a)refers to the transaction including Y and not include X, which at the same time includes the highest number of items. As noted regarding DCIS, however, sensitive transaction does not include X and Y. Algorithm (1.b) functions based on reducing right hand support and reduces support or confidence rule. Take the sensitive rule$X \rightarrow Y$, sensitive transaction is the one that includes items X and Y with fewer items.

Algorithm (2.a) reduces sensitive rule support until confidence and or support reaches a minimum confidence threshold.

Decrease Support of Right hand side Item of Rule Clusters (DSRRC) algorithm is introduced in[14]. Sensitive rules are clustered based on the right hand items and the algorithm tries to find several rules with minimum changes on the transactions. Sensitivity of the items is measured based on frequency of items and sum of sensitive rules and sensitivity of items of each transaction is taken as transaction sensitivity degree. Afterward, sensitive items of right hand rule is removed from the sensitive transaction and support and confidence rules are recalculated. The algorithm scans the database more than once and hides the rules that has one item on their right side. The algorithm is completely successful in hiding sensitive rules with an item on their right side, generate 27.28% ghost rules and hides 36% of non-sensitive rules [15].Advance Decrease Support of Right hand side Item of Rule Clusters (ADSRRC) is introduced in[15]; it outperforms DSRRC. The algorithm determines frequency of each item in the sensitive rules after extracting sensitive rules. Afterward, sensitivity of each transaction is obtained based on frequency of the items in the transactions of sensitive rule. Based on the sensitive rules, sensitivity of each item is obtained based on frequency of item on the right side of the rules and the sensitive item is selected for removal. Like DSRRC, sensitive rules are clustered based on items on the right side of the rules. In fact, to choose the sensitive transaction, frequency of the items are calculated for the both sides, while only frequency of the right side is taken into account to choose the sensitive item. When two items have equal frequency, the one with higher support over the database is selected. Afterward, the sensitive item is selected out of the sensitive transaction. Support of the sensitive rules are calculated after removing the item from the transaction, the whole procedure is repeated when hiding does not take place. It is notable that by hiding a sensitive rule, sensitivity of the transaction and the items are updated. ADSRRC is effective in hiding sensitive rules without generating ghost rules, still is hides 36.36% of non-sensitive rules.

Remove and Reinsert L.H.S of Rule (RRLR) is introduced by [16]with higher performance than DSRRC and ADRRC. The algorithm is capable of hiding the rules with several items on the right side. To perform hiding, the algorithm uses decreasing support or confidence. In fact, the algorithm is based both on support and confidence. For instance, in the case of rule$X \rightarrow YZ$, support of the rule is reduced by reducing support of XYZ and to reduce the confidence, item X is added to the transaction including right side rule (in this case "YZ") and not include left-side rule (X). The process of selection of sensitive transaction and item is like that of ADSRRC, except the fact that RRLR does not cluster right side of the rules. By finding the sensitive transactions and sorting them based on descending order of their sensitivity, sensitive rules are sorted based on descending order of confidence so that the rule with higher confidence is selected for hiding process. Afterward, hiding starts from the first transaction, while inclusion for the right/lest side of the rule is checked and when the both sides are included, the left side of the rule is removed from the transaction. Afterward, the algorithm comes to sensitive transaction which the does not include the right and left side and when it is spotted, the left side rule is added. Then, support and confidence of the sensitive rule is computed and in the case it is not hidden,

the above process is repeated and otherwise the next sensitive rule with higher confidence is selected. RRLR is successful in hiding sensitive rules and generates no ghost rule; although, it hides 22.73% non-sensitive rules.
Sliding Window Algorithm (SWA) is introduced by[17]. Taking into account the sensitive rules, the algorithm detects the sensitive transaction and then computes frequency of all items in the sensitive transactions. The item with highest frequency is detected for each rule and number of the transaction in each rule to be modified is determined based on threshold disclosure µ. Afterward, transaction with minimum items are selected and sensitive items is removed from the sensitive transaction. Disclosure threshold has a notable effect on HF, LR, and GR. To measure LR, threshold disclosure is set zero, which means all the sensitive rules must be hidden and then the rate of non-sensitive rules to be hidden is determined. Value of LR for SWA with zero threshold disclosure is 18.30%.

4.2 Border-based Algorithm

This approach considers association rule hiding through the modification of the borders in the lattice of the frequent and the infrequent itemset of the original database. Border-based algorithms achieve to hide the sensitive association rules by tracking the border of the non-sensitive frequent itemset and greedily applying the data modifications that have minimal impact on the quality of the border to accommodate the hiding of the sensitive rules. The first frequent itemset hiding methodology that is based on the notion of the border is proposed in[18]–[19]. It maintains the quality of database by greedily selecting the modifications with minimal side effect.

4.3 Exact Algorithms

This class of algorithm include non-heuristic algorithms that takes hiding process as a constraints satisfaction problem or an optimization problem, which can be solved by integer programming. Exact algorithms have high memory and time demand, while they meet required accuracy in data mining results. For instance, in [20]an exact algorithm tries to minimize difference between modified and original database.

5. Analyzing the Algorithms

Table 1 shows different algorithms based on HF, LR, and GR measures. Algorithms that try to reach higher hiding performance through increasing item support lead to generation of ghost rules and algorithms that rely on reducing support of the items for the same purpose, lead to hiding non-sensitive rules.

Table 1:AssociationRulesHeuristicAlgorithms-based on Side Effects Measure

Algorithm	HF	LR	GR
ISL	12.5%	0%	33%
DSR	0%	11%	5%
DCIS	0%	0%	75%
DCDS	0%	4%	1%
DSC	0%	9%	4%
DSRRC	0%	36.36%	27.27%
ADSRRC	0%	36.36%	0%
RRLR	0%	22.73%	0%
MINFIA(µ=0)*	0%	65%	0%
MAXFIA (µ=0)	0%	69%	0%
IGA (µ=0)	0%	44%	0%
SWA (µ=0)	0%	18.3%	0%

InFigure 2,the proposedalgorithmbased onside effects HF, LR, GR have been compared.

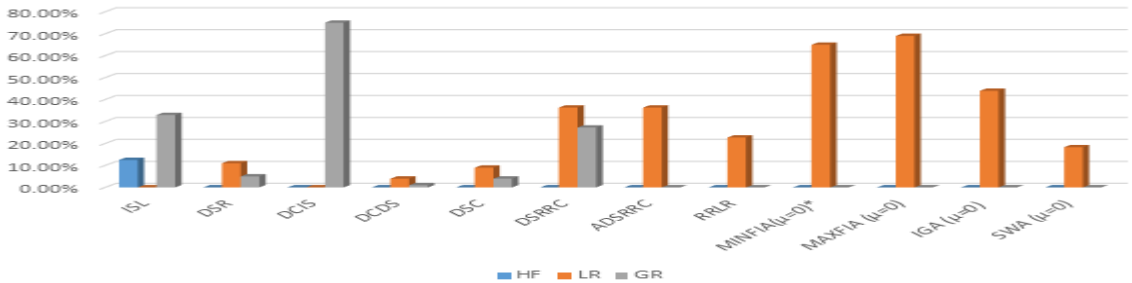

Figure 2. Compare of algorithms according to side effects

6. Conclusion

The strategies of associate rule hiding were introduced and then heuristic algorithms were discussed. Taking into account that transaction database in manipulated, the modified database must meet standards such as hiding failure, sensitive rule lost, and no generation of ghost rule. It is notable that fining an optimum modified database is an NP-Hard problem. On the other hand, there is an inverse relationship between privacy and accuracy of data mining results. The introduced algorithms showed this point.

References

[1] K. Sathiyapriya, Dr. G. Sudha Sadasivam,, "A Survey on Privacy Preserving Association Rule," International Journal of Data Mining & Knowledge Management Process (IJDKP), vol. 3, no. 2, pp. 119-131, 2013.
[2] Ahmed K. Elmagarmid, Amit P. Sheth, Association Rule Hiding for Data Mining, New York: Springer Science+Business Media, LLC , 2010.
[3] Gordon S. Linoff, Michael J. A. Berry, Data Mining Techniques: For Marketing, Sales, and Customer Relationship Management,, New York: Wiley, 2011.
[4] Stanley R. M. Oliveira, Osmar R. Zaiane, "Algorithms for Balancing Privacy and Knowledge Discovery in Association Rule Mining," 2002.
[5] Sarra Gacem, Djamila Mokeddem and Hafida Belbachir, "Privacy Preserving Data Mining: Case of Association Rules," International Journal of Computer Science (IJCSI), vol. 10, no. 3, pp. 91-96, 2013.
[6] M. Atallah, E. Bertino, A. Elmagarmid, M. Ibrahim, and V. S. Verykios, "Disclosure Limitation of Sensitive Rules," In Proc. of the 1999 IEEE Knowledge and Data Engineering Exchange Workshop (KDEX'99), pp. 227-245, 2000.
[7] Shyue-Liang Wang, Bhavesh Parikh, Ayat Jafari, "Hiding informative association rule sets," ELSEVIER, Expert Systems with Applications, pp. 316-323, 2006.
[8] Komal Shah, Amit Thakkar, Amit Ganatra, "A Study on Association Rule Hiding Approaches," International Journal of Engineering and Advanced Technology (IJEAT), vol. 1, no. 3, pp. 72-76, 2012.
[9] Shyue-LiangWang, Dipen Patel, Ayat Jafari, Tzung-Pei Hong, "Hiding collaborative recommendation association rules," Springer Science+Business Media, LLC , pp. 67-77, 2007.
[10] Shyue-Liang Wang, Rajeev Maskey, Ayat Jafari, Tzung-Pei Hong, "Efficient sanitization of informative association rules," ACM, Expert Systems with Applications: An International Journal, vol. 35, no. 1-2, 2008.
[11] H. Huang, X. Wu, and R. Relue, "Association Analysis with One Scan of Databases," in Proceedings of IEEE International Conference on Data Mining, Maebashi City, Japan, 2002.
[12] Stanley R. M. Oliveira, Osmar R. Zaiane, "Privacy Preserving Frequent Itemset Mining," in IEEE International Conference on Data Mining Workshop on Privacy, Security, and Data Mining. Conferences in Research and Practice in Information Technology, Maebashi City, Japan, 2002.
[13] Vassilios S. Verykios, Ahmed K. Elmagarmid, Elisa Bertino, Yucel Saygin, and Elena Dasseni, "Association Rule Hiding," IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, vol. 16, no. 4, pp. 434-447, 2004.
[14] Chirag N. Modi, Dhiren R. Patel, "Maintaining Privacy and Data Quality in Privacy Preserving Association Rule Mining," Second International conference on Computing, Communication and Networking Technologies, pp. 1-6, 2010.
[15] Komal Shah, Amit Thakkar, Amit Ganatra, "Association Rule Hiding by Heuristic Approach to Reduce Side Effects & Hide Multiple R.H.S. Items," International Journal of Computer Applications (0975 – 8887), vol. 45, no. 1, 2012.
[16] Nilesh R. Radadiya, Nilesh B. Prajapati, Krupali H. Shah, "Privacy Preserving in Association Rule mining," International Journal of Agriculture Innovations and Research (IJAIR), vol. 2, no. 4, pp. 208-213, 2013.
[17] Stanley Oliveira Osmar, Stanley R. M. Oliveira, "Protecting Sensitive Knowledge By Data Sanitization," in In Proc. of the 3rd IEEE International Conference on Data Mining, 2003.
[18] Xingzhi Sun, Philip S. Yu, "A border–based approach for hiding sensitive frequent itemset," in Proceedings of the 5th IEEE International Conference on Data Mining (ICDM), pp. 426-433, 2005.
[19] Xingzhi Sun, Philip S. Yu, "Hiding sensitive frequent itemset by a border–based approach," Journal of Computing Science and Engineering, vol. 1, no. 1, pp. 74-94, 2007.
[20] A. Gkoulalas-Divanis, V.S. Verykios, "An Integer Programming Approach for Frequent Itemset Hiding,," in ACM Conf. Information and Knowledge Management (CIKM '06), Proc, 2006.

AUTHORS

MohammadrezaLorestani received the B.Sc. from Razi University, Iran in 2011 and pursuing M.Sc. degree in Computer Engineering (Software) at Iran University of Science and Technology. His research interests including Data Mining, Knowledge Discovery andData Privacy.

Dr. EynollahKhanjari is Assistant Professor in School of Computer Engineering at Iran University of Science and Technology (IUST), his research interests including Data Mining, Knowledge Discovery,Data Warehousing, Programming Language Design and Implementation (PLDI)