A Survey on Association Rule Hiding in Privacy Preserving Data Mining

A. Hekmatyar[1], N. Nematbakhsh[2], M. Naderi Dehkordi [3]
1- Department of Computer Engineering, Najafabad Branch, Islamic Azad University, Najafabad, Iran
Email: arezoo_212002@yahoo.com
2- Department of Computer Engineering, Najafabad Branch, Islamic Azad University, Najafabad, Iran
Email: n.nemat@gmail.com (Corresponding Author)
3- Departmenr of Computer Engineering, Najafabad Branch, Islamic Azad University, Najafabad, Iran
Email: naderi@iaun.ac.ir

Received: April 2015 Revised: April 2016 Accepted: September 2016

ABSTRACT:
Data mining has been used as a public utility in extracting knowledge from databases during recent years. Developments in data mining and availability of data and private information are the biggest challenge in this regard. Preservation of privacy in data mining has emerged as an absolute prerequisite for exchanging confidential information in terms of data analysis, validation, and publishing. The main purpose of techniques and algorithms in privacy preserving data mining is non-disclosure of sensitive and private data with minimum changes in databases so that it would not have adverse effects on the rest of data. The present paper intends to present a brief review of methods and techniques regarding privacy of data mining in association rules, their classification and finally, classification of hiding algorithms of association rules followed by a comparison between a numbers of these algorithms.

KEYWORDS: Privacy Preserving Data Mining, Association Rule, Sensitive Data, Data Disclosure.

1. INTRODUCTION

By development of data mining techniques and possibility of data extraction from databases, information privacy preserving has drawn much attention. Privacy preserving is a new field of research which prevents alien influence on the individuals and organizations' privacy.

Privacy is divided into two main classifications of hiding data and hiding information. The first classification operates based on support that is to say it is not allowed to extract sensitive data as frequent data while the second classification operates based on confidence and in fact it is not allowed to extract frequent sensitive association rules.

In the meantime, on the hiding process there are some problems, the first of which is that hiding algorithms might not have the ability to hide sensitive data or rules. Secondly, it is possible to reach hidden sensitive data or rules through other elements of database.

This paper is a general review on hiding methods and their classification as well as analysis of hiding methods of association rules and limitations. Finally, there will be a brief comparison of the hiding algorithms.

2. TAXONOMY OF PRIVACY TECHNIQUES FROM VARIOUS ASPECTS

Privacy Preserving data mining techniques have been analyzed from three different perspectives. As they are presented in this paper, there was an attempt to complete them. The first perspective point out to the techniques which are run on transaction databases.

In this group, the focus is more on limiting the access, augmenting the data, eliminating the unnecessary data and auditing. Among the techniques which are included in this group, randomization and K-anonymity could be pointed to [1]. Randomization technique eliminates the possibility of records' recovery by adding noise to data values. For this, the amount of noise should be big enough.

This is one of the most common perturbation methods in privacy preserving data mining in distributed environment. Since indirect recognition of the records by using a combination of features in the general database is possible (search-limited method), the K-anonymity method has been developed. For hiding, this method uses two techniques of generalization (like omission of days and months from year) and suppression (like complete omission of feature). K-anonymity technique carries out this hiding in a way that each record is conformed to k other records. The second perspective in classification of

techniques points to those algorithms which are run in distributed environment where users tend to share data without jeopardizing individuals' privacy. For hiding, this group of techniques uses two methods of horizontal and vertical partitioning.

In horizontal partitioning, the records are distributed in different sites but in vertical partitioning, the features are distributed in different sites.

The third perspective refers to the hiding techniques which exchange neither data nor any database but in this group, the output of the data mining algorithms is exchanged once its sensitive data and information are disappeared. However, in most of the cases, even if the data are not available, the output of the programs such as extraction of association rules, classification and query processing will violate individuals' privacy. In fact, the main problem arises where after analysis of data and insensitive patterns; it is possible to discover sensitive patterns. Accordingly, it could be concluded that hiding all of the sensitive patterns is not merely enough. Hiding methods of association rules and itemsets are included in this group which is explained later on [2-5].

The outputs of data mining algorithms are association rules which their elements are frequent i.e. they are repeated in most of the transactions of transaction database. These association rules are defined as $A \rightarrow B$ so that A, B are both frequent and their sharing is null. For these two association rules, two parameters of support and confidence are used. In fact, support and confidence of rule must be bigger and equal to MCT and MST.

For hiding these rules, reducing support and confidence below the threshold is applied. For this, the transactions which include sensitive information are determined and the necessary operations and changes to reduce support and confidence are carried out on them.

This is possible through increase of support of element A in transactions on database or reduction of items from $A \cup B$ which supports that rule [4-6].

3. PROBLEM STATEMENT

As it has been observed in many references, among the main purposes of hiding algorithms, the following ones could be mentioned [4], [6].

- After hiding process, all of the insensitive rules should be still extractable by algorithms of data mining.
- The rules which have not been extractable before hiding process by algorithms of data mining should not be extracted after hiding process.
- In hiding process, minimum change on the main database should be made.

4. CLASSIFICATION OF HIDING ALGORITHMS IN ASSOCIATION RULES

The Hiding algorithms of association rules can be classified from different aspects. There are hiding algorithms of association rules which are divided into five Heuristic, exact, based on boundary, distortion and the approach based on encoding classes.

The algorithms of the Heuristic class are fast and efficient which choose a series of transactions selectively for hiding.

This method puts emphasis on optimization of hiding purposes but does not guarantee this. Through formulating the hiding problem, algorithms of the exact class guarantee that if there is a solution for the problem, it should find it; otherwise, either it leaves the problem or ignores some of the sub-purposes and satisfies the other ones in terms of priority.

By a preprocessing which is carried out before hiding process, the algorithms of based-on-boundary class introduce a group of rules as sensitive appropriate to each specific user so that less rules be hidden.

Two positive and negative boundaries are used for hiding in this method. The positive method includes maximum of sensitive rules and the negative one includes the minimum of sensitive rules which need to be hidden. In the algorithms of Distortion class, the database will be secured and it is constructed based on frequent itemset.

This method does not focus on how the database is changed but the aim is change of sensitive frequent items which are extracted by algorithms of data mining. It also focuses on construction of databases from network structure of collection of abundant elements, and algorithms of the encoding class which are used for secure sharing of association rules in distributed environment. In these algorithms two methods of horizontal and vertical divisions which were mentioned earlier are used for exchanging data. On the other hand, for classification of hiding algorithms of association rules, classification could be carried out in relation to the fact that whether these algorithms operate based on confidence or support. The other aspect deals with how database changes in hiding process.

From these perspective, two forms of distortion (conversion of 0 to 1 and vice versa) blocking (substitution? with data amounts) could be point to. Among other cases which are analyzed in classification of hiding algorithms there is this fact that algorithms carry out their hiding operation on the right side of left side or both sides of association rules. The final analyzing case of satisfaction of hiding purposes in association rules such as the amount of failure in hiding is the lost rules or unknown rules. Analysis of these cases is presented in comparative table of hiding algorithms [7, 8].

5. DISADVANTAGES OF HIDING METHODS In ASSOCIATION RULES

Heuristic algorithms might create undesirable side effects about insensitive patterns such as the lost or unknown rules in the hiding process. The based-on-boundary method has been improved in relation to the exploring method but it is still dependent to exploring methods in changes of main database. The based-on-boundary method has the capability of recognizing the optimum solution if existed.

The exact method has a rather more execution time complexity in relation to the other ones [2].

6. EVALUATION METERICS In ALGORITHM EFFICIENCY ANALYSISs

Analysis of efficiency in hiding algorithms is carried out by using two internal and external parameters. In terms of evaluating efficiency of hiding algorithms, the external parameters measure algorithm behavior against request from large dataset. In this regard, efficiency, scalability, data quality, and privacy level can be mentioned which will be explained below. On the other hand, dependent upon the kind of algorithms which use data sharing or pattern sharing, the internal parameters will be different. In fact, algorithms which share data are those that initially carry out hiding process of data series and then extract association rules by use of data mining algorithms.

Among the parameters which are included in this category one can mention hiding failure, lost rules, ghost rules and dissimilarity. Algorithms which use pattern sharing are those which initially extract association rules from data mining algorithms followed by hiding stages. Among the parameters of this group, side effect factor and recovery factor can be mentioned. A brief description of each one will be presented in the coming sections. [9], [10].

6.1. Side Effects Evaluation Metrics

Some of the most well-known evaluation metrics are represented in this section.

A. hiding failure:

The percent of sensitive patterns which are still remained in data series after hiding process is called hiding failure and it is calculated by the Eq. (1). (RP'= the number of sensitive patterns which are extracted after hiding from database and RP= the number of sensitive patterns which are extracted from the main database)[7].

$$HF = \frac{|RP\prime|}{|RP|} \quad (1)$$

B. lost itemset:

The percent of abundant insensitive patterns which are extracted from database after hiding process is called lost itemset and it is calculated by the Eq. (2). (R'P is the number of the extracted insensitive patterns before hiding and RP' is the number of extracted insensitive patterns after hiding) [8].

$$MC = \frac{(|RP'|-|RP|)}{|RP\prime|} \quad (2)$$

C. Ghost itemset:

The percent of new patterns extracted after hiding process is called Ghost itemset and is calculated the by the Eq. (3). (P' is the number of extracted patterns from database after hiding process and p is the number of extracted patterns from database before hiding process) [8].

$$AP = \frac{(|P\prime|-|P|)}{|P\prime|} \quad (3)$$

D. Dissimilarity:

Dissimilarity is the parameter which shows he difference between the main database and hiding database and it is calculated by the Eq. (4) (fd(i) is indicative of frequency of ith item in data series of X and n is the individual data items in data series) [7].

$$DISS(D,D') = \frac{1}{\sum_{i=1}^{n} fd(i)} \times \sum_{i=1}^{n}[fd(i) - fd'(i)] \quad (4)$$

E. Side effect factor:

Side effect factor is similar to measuring lost rules parameter.

This parameter shows the amount of insensitive association rules which are deleted by the effect of hiding process. It is calculated by the Eq. (5) [10].

$$SEF = \frac{(|P|-(|P'|+|R_p|)}{(|P|-|R_p|)} \quad (5)$$

6.2. Performance Evaluation Metrics

A. Efficiency:

Efficiency is a protecting privacy algorithm in appropriate use of available resources and also running with good performance. This is related to the processor usage time, the amount of memory needed and connections.

B. Scalability:

Scalability refers to how effectively privacy protection methods during increase size of data are managed and also it is assurance of accuracy of hidden data and extracted data. Scalability is measured when the efficiency of algorithm is reduced or when storage request and connective costs are increased while algorithm encounters with large data base.

C. Data quality:

Data quality depends on two parameters. The first one is quality of data series after sanitize process and the second one is quality of data mining in comparison to the results before sanitize process.

Accordingly, three quality parameters are analyzed the first of which is accuracy that is the degree of approximation before and after sanitize. The second one is completeness that refers to the degree of lost rules in sanitize data base. And the third one is consistency which deals with evaluation of connections between data items.

D. Privacy level:

Analysis of predicting information after hiding process is called privacy level.

7. HEURISTIC METHODS IN PRIVACY PROTECTION

DSR algorithm: at first, this algorithm obtains abundant rules based on support and confidence and chooses the first sensitive rule and determines the transactions which fully support that rule.

Then, RHS of the rule is deleted from transactions so that the confidence of rule would not be less than threshold. Also there is a transaction for omission. This is continued and if there was not a transaction and confidence is still more than threshold, then, the rule would not be hidden and we change the database to the initial status and consider the next sensitive rule [3].

ISL algorithm: this algorithm is similar to DSR algorithm with the difference that it chooses the transactions which do not support sensitive rule and adds sensitive LHS to them and if there is not any transaction and the amount of confidence is not still less than threshold, the rule will not be hidden and database is turned to the initial status [3].

RRLR algorithm: at first, sensitive rules are analyzed in this algorithm and their sensitivity will be achieved (for example how many times each item has been repeated in sensitive rule series) and the rules with form of $X \rightarrow YZ$ are taken into account. After that, sensitivity of transactions will be achieved based on sensitivity of sensitive items and we will arrange them based on sensitivity and length of the items in descending order. Up to the time that all of the sensitive rules are not hidden, the LHS element is deleted from the transactions which totally cover that sensitive rule and it starts from the next transaction where there is not at least one of the RHS elements by adding the LHS element.

Then, coverage and confidence are calculated again and if they are less than threshold, it goes for the next sensitive rule; otherwise, it continues this process [7], [11], [12].

ADSRRC algorithm: This algorithm classifies sensitive rules based on different RHS elements of them and after that they will be valued based on the number of items.

Then, according to this, sensitivity of transactions is calculated and based on sensitivity and the number of items of each transaction; they will be arranged in descending order. It starts from the highest transaction and highest sensitivity classification to consider the first sensitive rule.

If the RHS member of sensitive rule does exist in the transaction, it will delete it and calculate support and confidence. If it is not less than threshold, the other transactions will be tried to hide the rule [7].

DSRRC algorithm: this is similar to the previous algorithm with this difference that by each omission, it calculates all stages of algorithm again including determining sensitivities that increases the algorithm performance time.

CR algorithm: firstly, in this algorithm, the transactions which totally cover the sensitive rule are determined, then, it arranges them in ascending order based on the number of items which it can cover. And as long as the degree of rule confidence is not less than difference of SM from MCF (SM input parameter), it continues its work.

It chooses the first transaction and selects an item of J from the rule with the highest support degree and puts "?" instead of it. It calculates support and confidence of the rule again as well as confidence of the rules that are effective on them. After that, it deletes the transaction from transaction series and after the fact that the rule is secure; it will delete the rule from the rule series [6].

CR algorithm: In this algorithm, at first, transactions which cover LHS of sensitive rule trivially and do not totally cover the RHS are placed in the series.

After that, for each transaction of the series it calculates the number of items of LHS of sensitive rule in them, i.e. each rule has some of the LHS items and arranges them in descending order and as long as the confidence of the rule is not less than difference of SM from MCF, it chooses the first transaction, then, instead of LHS items we put question mark (?). If it does not have that item, maximum confidence of LHS and minimum confidence of the rule will be calculated and the transaction will be deleted from the series. If it is hidden, we will delete the rule from the sensitive series, too [8].

RR algorithm: in this algorithm, sensitive transactions are determined for each rule, then, a random number is given to each rule. The numbers of rule items are divided by that number and the remaining is calculated. The remaining is the number of the item which is deleted. Then, the Eq. (7) is used for calculating the number of transactions for sanitization (Ψ is given as the input of algorithm), after that, equal to this number, the transactions which have the considered item will be deleted [4], [5].

$$\text{Number of Transaction for Sanitization} = \text{Number of Sensitive Transaction for R} \times (1 - \Psi) \quad (7)$$

RA algorithm is similar to the previous one with this difference that the remaining is not calculated anymore and continues the process based on the number of items chosen randomly [5], [6].

HF algorithm: an immunization matrix is used in this algorithm for hiding process. The dimensions of this matrix are equal to database items and the main diameter elements are valued with 1 and where the rows and columns of the matrix are subcategory equal to the sensitive item. And the numbers of category J (column) in insensitive elements is less than category I (row) in insensitive one -1; otherwise, it is considered zero. Therefore, by multiplying this matrix by the main matrix (database) it immunizes the database [12].

NHF algorithm: this algorithm is the same as the previous one but with this difference that one condition has been added to its immunization matrix and it is the fact that if the factor of row and column of the matrix is not subcategory of sensitive items, and it is subcategory of insensitive ones, its amount will be considered 1.

HPCME algorithm: this algorithm equals to NHF. In multiplying immunization matrix by database matrix, where answer zero is obtained by multiplying -1 in database matrix, it only takes the answer with possibility of PR of zero and possibility of 1-PR of one.

These scholars have introduced an algorithm as SWA. Without considering the size of database and number of sensitive rules, this algorithm needs to scan databases just once. Since it is not based on memory, it could be applied in very big databases. In the operation of this algorithm, initially, the sensitive and insensitive transactions are determined. Then, among the elements of sensitive transactions it chooses an element with highest frequency as the victim element and it also calculate the numbers of transactions which need to be modified.

As for the next step, it arranges the transactions which support each sensitive pattern or rule in ascending order based on length. After that, deletion of the victim element from them is continued until they are hidden. The disadvantages of this method are hidden failure method and lost rules. In this algorithm, a parameter called K is used as the size of window that means the number of chosen transaction for hiding process. Also, if they have several rules of one or more sharing elements, one of the sharing elements is chosen as the victim and all of the support rules will be hidden. This is from the advantages of this algorithm [13].

GIH algorithm: in this algorithm, at first, sensitive rules are arranged in descending order based on size and support. After that, for each rule, transactions are arranged based on their size in ascending order and then, the amount of N is calculated by use of the Eq. (6) and equal to its number, question mark "?" is put instead of the item that was sensitive and had the most support. Finally, support is calculated again and the database is updated [14].

$$\text{N} - \text{Iterations} = |\text{t2}| - (\text{MST} - \text{SM}) \times |\text{D1}| \quad (6)$$

Wang et al. [15] have developed two novel algorithms. The first one (DCIS) decreases confidence amount to below disclosure threshold by increasing support of LHS. The second algorithm (DCDS) decreases the amount of rule confidence below disclosure threshold by decreasing the amount of support of RSH. Like the previous methods, the flaw of this method is in the order of transactions in database.

Oliveira et al., [4], [5] were the first ones who presented some methods for simultaneous hiding of several sensitive rules. This algorithm only needs to times database scan for hiding process. In the first scan an index file is created for sensitive transactions so that the speed of accessibility to them will be increased. The second scan is for hiding sensitive rules. The authors have presented four algorithms based on the proposed method.

The first algorithm (Naïve) initially recognizes sensitive transactions then, it determines the victim element for each sensitive pattern. After that, it determines for each sensitive pattern that how many deletions should be carried out for hiding. And at the end, it will delete the victim element from the considered sensitive transaction. Based on the proposed method, like Naïve, the second algorithm (MinFia) at first recognizes the sensitive transactions. For the next step, for each sensitive pattern, most impressive amount of support is chosen as the victim element. Then, deletion numbers for each sensitive pattern is determined followed by arranging the related sensitive transactions to each sensitive pattern in ascending order based on the degree. After that, the victim elements will be deleted from transactions. The third algorithm (MaxFia) is similar to the second algorithm with this difference that in this one, the chosen victim element is an element with maximum amount of support.

The fourth algorithm (IGA) operates based on divisions of patterns to groups based on sharing element series [13].

They proposed another algorithm using immunization matrix. This algorithm is a combination of the previous methods with IGA algorithm. In this method, because of analysis of sensitive with insensitive sharing elements, there is hiding failure in some cases. Li et al have proposed an algorithm called HarRFI for hiding sensitive patterns. For hiding dependency rules, the method of patterns' support reduction has been used and they have proposed two approaches for reduction of side effects.

In the first approach, the way of choosing victim element, different elements based on different transactions for deletion are chosen, the elements which are in sensitive patterns but not in the insensitive ones.

In the second approach, deletion of the victim element is based on opposition of sensitive transactions.

We can delete the victim element from the transactions which their degree of opposition is below the minimum of opposition. By degree of opposition we mean the numbers of sensitive patterns in one transaction. The minimum of opposition is a number which the users import as input. The relative insensitive patterns are insensitive patterns which include common elements with sensitive patterns.

The sensitive transactions are classified into four groups in this algorithm:

- Including one sensitive pattern only
- Including one sensitive pattern and one relative insensitive pattern
- Including one sensitive pattern and more than one relative insensitive pattern
- Including more than one sensitive pattern and more than one relative insensitive pattern

This algorithm has a good efficiency as long as the majority of sensitive transactions are like the second group. Among the advantages of this algorithm are using subcategories in the sensitive pattern and considering only one victim element for each pattern. Among its disadvantages, long time performance and limited to certain conditions for having high efficiency could be mentioned. In the rest of the cases it has very lost elements. Also, by reduction of the minimum opposition of the failure in hiding sensitive patterns and by increase of the minimum opposition, the numbers of lost elements will be increased. Also, sensitive patterns in the order of arrangement of sensitive transactions and order of insensitive patterns are abundant on the list [2].

Verykios et al. [8] were the first person who presented an exploring algorithm for hiding the association rules through reducing support of productive abundant patterns of that sensitive rule below the disclosure threshold. They have proved in his paper that the immunized database has been created based on an algorithm with NP degree of difficulty.

This algorithm initially arranges the patterns based on their support in descending order then it takes the first sensitive pattern and hides it. This algorithm hides the sensitive patterns one by one. After each running of the algorithm, the list of sensitive patterns will be analyzed. If a sensitive one has not been hidden, it hides it and if a pattern is hidden it will be deleted from the list.

In this algorithm, graphs have been used for hiding where after sketching the graph, the subcategories of the sensitive pattern which are in higher level will be analyzed. Then, a subcategory with the highest amount of support is chosen for deletion from the transaction which includes the main sensitive pattern and has less length. Among the disadvantages of this method, one can mention the great numbers of lost abundant patterns [10].

Dasseni et al., [11] have designed three heuristic algorithms for hiding association rules based on the reduction of support or confidence. This algorithm also hides one of the sensitive rules in each time. The first algorithm reduces confidence amount in two ways.

DSA algorithm: in this algorithm, at first. The graph of abundant items is sketched then, the item equivalent to sensitive rule ($A \rightarrow BC$, its item is ABC) is issued. Since access to the hidden rule should be impossible from other directions of graph, all of its categories should be blocked, too. This process continues to the first level which is called leading attack inference. Then backward attack inference is performed that item series should include item marked. Among features of this algorithm the following ones could be mentioned [16].

1) DSA blocks some inference channels to prevent recovery of sensitive rules.

2) DSA decreases the side effect factor significantly.

3) DSA is an acceptable method to protect sensitive rules before rule sharing.

4) DSA provides two metrics for measuring (SEF) and (RF)

In the first method, this operation continues through increase of support amount of LHS of sensitive rule so that the rule is hidden. The disadvantage of this method is possibility of creation of Ghost rules and among advantages of this method we can point to lack of lost rules.

In the second method, hiding of sensitive rules is carried out through reduction of support of RHS as long as the amount of confidence rule becomes less than disclosure threshold.

The third algorithm proposed by him, instead of reducing sensitive confidence rule carries out hiding by usage of reduction of support amount of LHS or RSH. In this paper the focus has been on the time of algorithm performance. Among limitations of this algorithm, we can refer to lack of hiding sensitive rules which has overlap [10].

Saygin et al., [6] has developed the proposed method by Atallah [10]. He has also proposed two more algorithms based on reduction of support of productive patterns of sensitive rules. This algorithm functions in two ways. In the first one, with usage of support reduction with deletion from a transaction with the least length and in the second one, at first the sensitive patterns are arranged based on support amount. After that, the hiding process is carried out in a cyclic turn. The first algorithm (1.a) does hiding of sensitive rules with increase of support of LHS. This algorithm operates in a way that the transactions which support LHS trivially, based on the numbers of the elements of LHS that are supported, are arranged in descending order.

Then, it chooses the first transaction and adds all of the existed LHS elements which are not (it is 0) in that transaction. This is continued as long as the rule confidence gets below disclosure threshold. Disadvantages of this method include hiding failure and creating Ghost rules. The second algorithm (1.b) chooses those transactions which totally support that rule and then arrange them in ascending order based on their length.

After that, it chooses the first transaction and deletes an element from RHS elements from that transaction. This process continues as long as support threshold or rile confidence get below disclosure threshold. The third algorithm (2.a) at first finds the transactions which cover the rule. Then, the transactions are arranged in ascending order based on their length. After that, the first transaction is chosen and an element of the rule with lowest amount of support will be deleted as the victim element.

This process continues as long as the rule support gets below disclosure threshold. The fourth algorithm (2.b) initially finds the sensitive patterns that construct sensitive rules and after that it arranges their support in descending order based on their length. Then, it embarks on hiding these patterns one by one. Hiding operation based on support reduction is below disclosure threshold.

To follow the mentioned purpose, an element with highest support is deleted as the victim element from the transaction with minimum length that has the considered pattern.

Advantage of this algorithm is that it chooses transaction with minimum length. This method reduces the side effects on the insensitive patterns. Also, choosing an element with the most amount of support as the victim will have a less possibility in the lost pattern. The fifth algorithm (2.c) carries out hiding process through support reduction of the producing patterns of the sensitive rules.

In this algorithm, initially the sensitive patterns of sensitive rule constructors are arranged based on length and support amount. Then the first pattern is chosen and hidden. In this algorithm, cyclic turn method is used for deleting the patterns from transactions. Being fair, and having low side effects are among the advantages of this algorithm, and lack of overlapping between critical rules is its disadvantage [8].

Lin et al. [17] proposed the HMAU algorithm for hiding the sensitive itemset. In this algorithm, a suitable transaction is selected based on side effects, including hiding failure, lost itemsets and new itemsets for removal. The aim of transaction removal is to reduce the support of sensitive itemset.

Table 1. The comparison between methods in efficiency metrics and categories

Algorithm	Support Base	Confidence Base	Using RHS	Using LHS	Blocking Method	Distortion Method	Hiding Failure	Ghost Rule	Lost Rule
DSR	×	✓	✓	×	×	✓	✓	✓	✓
ISL	×	✓	×	✓	×	✓	✓	✓	×
RRLR	×	✓	×	✓	×	✓	✓	×	✓
ADSRRC	×	✓	✓	×	×	✓	✓	×	✓
DSRRC	×	✓	✓	×	×	✓	✓	✓	✓
HF	✓	×	-	-	×	✓	✓	×	✓
NHF	✓	×	-	-	×	✓	✓	×	✓
HPCME	✓	×	-	-	×	✓	✓	×	✓
CR2	×	✓	✓	×	✓	×	×	×	✓
CR	×	✓	✓	×	✓	×	×	×	✓
GIH	×	✓	✓	×	✓	×	✓	×	✓
RR	×	✓	-	-	×	✓	✓	×	✓
RA	×	✓	-	-	×	✓	✓	×	✓
2.b	✓	×	-	-	×	✓	✓	×	✓
2.c	✓	×	-	-	×	✓	✓	×	✓
MinFia	×	✓	-	-	×	✓	×	✓	✓
MaxFia	×	✓	-	-	×	✓	×	✓	✓
Naïve	×	✓	-	-	×	✓	×	✓	✓
IGA	✓	×	-	-	×	✓	✓	×	✓
HARFI	✓	×	-	-	×	✓	✓	×	✓
DIS	×	✓	×	✓	×	✓	✓	✓	✓
DCDS	×	✓	✓	✓	×	✓	×	✓	✓
SWA	✓	✓	-	-	×	✓	✓	×	✓

Cheng et al., [18] proposed a new distortion based method which hides sensitive rules by removing some items in a database to reduce the support and confidence of sensitive rules below specified thresholds. In order to minimize side effects on knowledge, the information on non-sensitive itemsets contained by each transaction is used to sort the supporting transactions. The candidates that contain fewer non-sensitive itemsets are selected for modification preferably. In order to reduce the distortion degree on data, the minimum number of transactions that need to be modified to conceal a sensitive rule is derived. Comparative experiments on real datasets showed that the new method can achieve satisfactory results with fewer side effects and data loss.

8. CONCUSIONS AND REMARKS

As shown briefly in Table 1 many algorithms and methods have been recently presented for privacy preserving data mining. The fundamental notions of the existing privacy preserving data mining methods, their merits, and shortcomings are presented. The current privacy preserving data mining techniques are classified based on distortion, association rule, hide association rule, Blocking techniques, distortion methods and their side effects, where their notable advantages and disadvantages are emphasized. However there is an opportunity for further study, research and development in this issue. In this article privacy preserving techniques were introduced and discussed. Distortion and blocking techniques have been more concentrated on privacy preserving and have been more emphasized on hiding rules or preventing from making sensitive rules. These methods are simple and have many side effects. Side effects involve losing non sensitive rules, making ghost rules that are dangerous for sensitive database such as medical science and leads to failure in hiding. Another challenge in this issue is about inference sensitive rules by using non sensitive ones.

REFERENCES

[1] Lee, G., Chang, C.-Y., & Chen, A. L. **"Hiding sensitive patterns in association rules mining"**, *Computer Software and Applications Conference. COMPSAC*, Proceedings of the 28th Annual International, 2004.

[2] Li, X., Liu, Z. and Zuo, C., **"Hiding association rules based on relative-non-sensitive frequent itemsets"**, *In Cognitive Informatics. ICCI'09. 8th IEEE International Conference*, pp. 384-389, 2009.

[3] Natarajan, R., Sugumar, R., Mahendran, M., & Anbazhagan, K. **"Design and Implement an Association Rule hiding Algorithm for Privacy Preserving Data Mining"**, *International Journal of Advanced Research in Computer and Communication Engineering*, 2012, Vol. 1(7).

[4] Oliveira, S. R., & Zaiane, O. R. **"Privacy preserving frequent itemset mining"**, *Paper presented at the Proceedings of the IEEE international conference on Privacy*, security and data mining, 2002, Vol. 14.

[5] Oliveira, S. R., & Zaiane, O. R. **"Algorithms for balancing privacy and knowledge discovery in association rule mining"**, *Paper presented at the Database Engineering and Applications Symposium, 2003. Proceedings*, Seventh International.

[6] Saygin, Y., Verykios, V. S., & Clifton, C. **"Using unknowns to prevent discovery of association rules"**, *ACM Sigmod Record*, Vol. 30(4), 2001, pp. 45-54.

[7] Shah, K., Thakkar, A., & Ganatra, A. **"Association Rule Hiding by Heuristic Approach to Reduce Side Effects & Hide Multiple RHS Items"**, *International Journal of Computer Applications*, 2012, Vol. 45(1).

[8] Verykios, V. S., Elmagarmid, A. K., Bertino, E., Saygin, Y., & Dasseni, E. **"Association rule hiding. Knowledge and Data Engineering"**, *IEEE Transactions on*, Vol. 16(4), 2004, pp. 434-447.

[9] Wang, E. T., & Lee, G. **"An efficient sanitization algorithm for balancing information privacy and knowledge discovery in association patterns mining"**, *Data & Knowledge Engineering*, Vol. 65(3), 2008, pp. 463-484.

[10] Atallah, M., Bertino, E., Elmagarmid, A., Ibrahim, M., & Verykios, V. **"Disclosure limitation of sensitive rules"**, *In Knowledge and Data Engineering Exchange, (KDEX'99) Proceedings. Workshop*, 1999, pp. 45-52. IEEE.

[11] Dasseni, E., , Vassilios S. Verykios, Ahmed K. Elmagarmid, and Elisa B., **"Hiding association rules by using confidence and support"**, *In Information Hiding. Springer Berlin Heidelberg*, 2001, pp. 369-383.

[12] Ganatra, K. S. a. A. T. a. **"Association Rule Hiding by Heuristic Approach to Reduce Side Effects and Hide Multiple R.H.S. Items"**, *International Journal of Computer Applications*, 2012, 45, pp. 1-7.

[13] Oliveira, S. R. **"Protecting sensitive knowledge by data sanitization"**, *13th IEEE International Conference on Data Mining*, 2003.

[14] Wang, S.-L., Parikh, B., & Jafari, A. **"Hiding informative association rule sets. Expert Systems with Applications"**, Vol. 33(2), 2007, pp. 316-323.

[15] Wang, S.-L., Patel, D., Jafari, A., & Hong, T.-P. **"Hiding collaborative recommendation association rules. Applied Intelligence"**, Vol. 27(1), 2007, pp. 67-77.

[16] Yeh, J.-S., Hsu, P.-C., & Wen, M.-H. **"Novel Algorithms for Privacy Preserving Utility Mining. Paper presented at the Intelligent Systems Design and Applications"**, *ISDA'08. Eighth International Conference on Intelligent Systems Design and Applications*, Vol. 1, 2008, pp. 291-296.

[17] C-W. Lin, T-P. Hong, H-C. Hsu, **"Reducing Side Effects of Hiding Sensitive Itemsets in Privacy Preserving Data Mining"**, *The Scientific World Journal*, Vol. 2, 2014.

[18] Cheng, P., Roddick, J. F., Chu, S. C., & Lin, C. W. **"Privacy preservation through a greedy, distortion-based rule-hiding method"**, *Applied Intelligence*, pp. 1-12, 2015.