Indonesian Journal of Electrical Engineering and Computer Science
Vol. 23, No. 2, August 2021, pp. 1137~1145
ISSN: 2502-4752, DOI: 10.11591/ijeecs.v23.i2.pp1137-1145

Privacy preserving association rule hiding using border based approach

Suma B., Shobha G.
Department of Computer Science and Engineering, RV College of Engineering, Bengaluru, India

Article Info

Article history:

Received Apr 1, 2021
Revised Jul 8, 2021
Accepted Jul 13, 2021

Keywords:

Data sanitization
Frequent item sets
Privacy preserving data mining
Sensitive rules

ABSTRACT

Association rule mining is a well-known data mining technique used for extracting hidden correlations between data items in large databases. In the majority of the situations, data mining results contain sensitive information about individuals, and publishing such data will violate individual secrecy. The challenge of association rule mining is to preserve the confidentiality of sensitive rules when releasing the database to external parties. The association rule hiding technique conceals the knowledge extracted by the sensitive association rules by modifying the database. In this paper, we introduce a border-based algorithm for hiding sensitive association rules. The main purpose of this approach is to conceal the sensitive rule set while maintaining the utility of the database and association rule mining results at the highest level. The performance of the algorithm in terms of the side effects is demonstrated using experiments conducted on two real datasets. The results show that the information loss is minimized without sacrificing the accuracy.

Corresponding Author:

Suma B.
Department of Computer Science and Engineering
RV College of Engineering
Mysore Road, Bengaluru, 560059, India
Email: sumab_rao@rvce.edu.in

1. INTRODUCTION

Data mining is the process of analyzing enormous amounts of data to identify hidden useful predictions and patterns that help in decision-making. Data mining is used in areas such as research, business, engineering, and government security. In collaborative data mining, organizations have to share information in order to shorten processing time and improve asset productivity, quality, and accuracy. Sharing of data enables policy-makers and researchers to analyze data in order to obtain useful knowledge benefiting the society as a whole. However, much of the shared information may be sensitive data that raises serious privacy concerns as a consequence, privacy preserving data mining (PPDM) techniques deal with the efficient conduction of data mining functionalities without sacrificing the privacy and usefulness of the data [1]. The analysis of PPDM techniques considers the effects on data mining results as well as in preserving the privacy of the original data. Verykios *et al*. [2] proposed a hierarchy for classifying the PPDM algorithms. Bertino *et al*. [3] proposed a method for comparing and measuring the PPDM algorithms of various types. The framework consists of a number of criteria for evaluating PPDM algorithms. Techniques of PPDM can be broadly classified into three categories: knowledge hiding, cryptography, inference control, and query auditing. The knowledge hiding technique sanitizes the sensitive knowledge before sharing data with a third party in order to ensure privacy [4]. The cryptography-based methods used in multiparty computation, where several sites cooperate to discover data mining findings without exposing the data at

their individual sites [5]. The inference control and query auditing methods preserve the privacy of sensitive information by modifying the results of the query [6].

Privacy preserving association rule mining (PPARM) research domain focuses on the security and privacy implications resulting from the applications of various data mining techniques. The goal of PPARM is to find useful relationships among items in the transaction database while protecting the privacy of the data owner. This paper focuses on the privacy concerns of data owners toward the knowledge extracted from the data, before sharing the data with third party. Data owners must identify the association rules which must be considered as sensitive knowledge.

The remainder of this paper is organized as follows: Section 2 provides a brief review of the literature on association rule hiding. Section 3 gives the problem formalization of sensitive association rule hiding, while in section 4, we expose our proposed border based sensitive association rule hiding technique. Section 5 contains the experimental evaluation and finally, we conclude the paper in section 6.

2. RELATED WORK

The aim of association rule hiding algorithms is to conceal sensitive data so that it cannot be discovered using an association rule mining (ARM) algorithm while still reaping the benefits of the ARM. Association rule hiding approaches are categorized into three major classes including, heuristic, reconstruction, and metaheuristic.

In order to find a successful solution, a heuristic method searches the solution space. Heuristic based algorithms use perturbation or distortion technique to find sensitive transactions and items for sanitization. A perturbation-based method modifies the values of some item to bring the support or confidence of the sensitive rules below the threshold. The blocking sanitization approach replaces the values of selected items with a question mark (an unknown value) to shield the confidential rules from disclosure. An itemsets-based association rule hiding method [7] uses a perturbation technique that selectively hides sensitive frequent itemsets by reducing the support count of the itemsets below the user specified minimum threshold. However, this approach has the drawback of hiding one pattern at a time. Dasseni *et al*. [8] presented an association rule hiding technique that conceals sensitive rules by lowering their support or confidence below user-specified thresholds. However, this algorithm generates undesired artificial rules that reduce the utility of the sanitized database. Modi *et al*. [9] presented a rule hiding algorithm that groups the sensitive rules on the basis of consequent itemsets of the sensitive rules. The algorithm hides as many rules as possible by sanitizing a small number of transactions, the. Hong *et al*. [10] proposed SIF-IDF technique that utilizes TF-IDF measure to determine the similarity between transactions and sensitive itemsets. This greedy approach assigns a SIF-IDF value to each transaction in order to determine the degree of correlation between the transactions. Cheng *et al*. [11] presented a multi-objective optimization method that considers multiple parameters for concealing sensitive frequent itemsets. In [12], the item-based hybrid algorithm is discussed that minimizes the side effects on the dense and sparse databases. Telikani *et al*. [13] devised an algorithm to hide sensitive association rules by combining heuristic and border strategies. Wang *et al*. [14] devised a blocking based rule hiding that replaces data by unknowns such that patterns containing identified items on the antecedent part of the rule are not generated during the ARM process. Homomorphic encryption algorithm [15] conceals sensitive association rules in the outsourced data that is uploaded by multiple data owners. The key disadvantage of the heuristic technique is that it fails to provide an optimal solution to the data hiding problem in the vast majority of circumstances.

Data reconstruction approach keeps the input database aside and accomplishes data sanitization on an itemsets lattice called a knowledge base. The sanitized knowledge base is then used to restore a new published database using a reconstruction technique. Chen *et al*. [16] a devised privacy preserving data sharing framework based on dataset reconstruction. The framework performs knowledge base modification rather than transaction modification and a practical balance between data sharing and privacy preservation. Database reconstruction algorithm based on FP tree mines inverse frequent itemsets [17]. Database reconstruction-based algorithm for frequent itemsets hiding is proposed in [18] that achieves a reasonable data utility and a high degree of privacy. The reconstruction strategy integrates the concepts of database extension and inverse frequent itemsets mining. The drawback of the reconstruction-based approach is that number of transactions is restricted in the sanitized database.

Metaheuristic based solutions include evolutionary algorithms, use iterative evolutionary mechanisms for exploring state-space to find a global optimal solution for the rule hiding problem. The genetic algorithm-based approach [19] utilizes four fitness strategies for defining the fitness function. The weighted sum function minimizes the side effects on the transformed database. Khan *et al*. [20] devised a genetic algorithm framework that reduces the loss of information when compared to the algorithm in [19]. Lin *et al*. [21] presented cpGA2DT, sGA2DT, and pGA2DT [22] algorithms for concealing sensitive

frequent itemsets by deleting victim transactions using genetic algorithms. Lin *et al*. [23] developed a multi-objective scheme by deleting transactions to hide the sensitive association rules. Afshari *et al*. [24] proposed a Cuckoo optimization algorithm for concealing sensitive rules. The algorithm achieves the solution with the reduced side effects using three fitness functions. Doan *et al*. [25] enhanced the method in [24] to minimize the side effect in terms of lost non-sensitive rules. Ant colony-based solutions to conceal the frequent itemsets obtained improvement in the performance in terms of side effects [26]. Genetic encoding scheme proposed in [27] utilizes objective function to estimate the effect on non-sensitive rules and provides recursive computation to reduce the side effects. ABC4ARH [28] rule hiding algorithm, selects sensitive transactions using neighborhood generation mechanism that balances between exploitation and exploration. Metaheuristic based approaches do not define the strategies for identifying victim item for deletion or insertion in the selected database transactions.

3. PROBLEM STATEMENT

This section provides some basic notations and definitions used in our problem statement. The ARM algorithm plays a vital role in the analysis and decision-making process to discover relationships among items in a transactional database.

3.1. Definitions

Let $I = \{I_1, I_2, ., I_k, ., I_m\}$ be a set of literals known as items and $D = \{T_1, T_2, \ldots, T_i, \ldots, T_N\}$ be a transactional database. Each transaction T_i represents a list of items from I such that $T_i \subseteq I$, referred to as an itemsets in the transaction. If $Y \subseteq T_i$, then the transaction T_i is said to have an itemsets Y. An association rule $A \rightarrow B$ is an association between sets of items A and B such that A, B disjoint itemsets. The support and confidence metrics are utilized for measuring the strength of an association rule. The matric support defines how often a rule applies to a given data set, while the metric confidence determines the reliability of a rule's inference. The definitions of support and confidence are formulated in (1) and (2).

$$support\ (A \rightarrow B) = \frac{|A \cup B|}{|D|} \tag{1}$$

$$confidence\ (A \rightarrow B) = \frac{|A \cup B|}{|A|} \tag{2}$$

The ARM algorithm is executed in two phases. The first phase of the mining terminates when all itemsets that occur at least as frequently as a predetermined minimum support count are discovered. Then the second phase is executed to discover the association rules that satisfy minimum confidence. Let σ, δ be the user specified minimum support threshold and the minimum confidence threshold, respectively. Rules that satisfy both σ and δ are called strong association rules. Sensitive association rules are strong rules that must be concealed before data is shared or published. The goal of the rule hiding problem is to protect sensitive association rules by modifying the original dataset.

3.2. Problem description

The sensitive association rule hiding problem discussed in this paper is defined as follows: Given a database D, minimum support σ, minimum confidence δ, a set of strong rules R mined from D and a set of sensitive rules $R_s \subseteq I$ to be hidden, transform D into a transformed database D′. The hiding process accomplishes to secure sensitive rules R_s from being disclosed while keeping the side effects at a minimum level. Finding an optimal solution to the sensitive rule hiding problem is NP-hard. In this paper, we propose a border based approach to reduce the support of sensitive rules by deleting the items from selected transactions such that no sensitive rule is discovered from the sanitized database D′. The proposed hiding approach aims to transform D into D′ that maximizes the number of non-sensitive rules in D that can still be mined.

4. PROPOSED SOLUTION

This section presents the devised border-based rule hiding (BBRH) algorithm to conceal sensitive association rules by reducing the support of sensitive itemsets below σ. A sensitive rule $A \rightarrow B$ is hidden if $\text{support}(A \rightarrow B) < \sigma$) or confidence($A \rightarrow B$) $< \delta$.The proposed method aims to minimize the side effects in terms of the number of missing rules and artificial rules while hiding sensitive rules.

The proposed hiding strategy employs the concept of border presented in [29] to identify the positive border of non-sensitive frequent itemsets and the negative border of sensitive frequent itemsets i.e.,$^{+}$(R-R_s), $Bd^{-}(R_s)$. Given a set of itemsets Z the positive border(respectively negative border) of $Bd^{+}(Z)$ (respectively $Bd^{-}(Z)$), is a subset of Z with the following two properties: 1) $Bd^{+}(Z)$ (respectively $Bd^{-}(Z)$) is an antichain collection of sets 2) $\forall U \in Z$, there exists an itemsets $V \in Bd^{+}(Z)$ (respectively $V \in Bd^{-}(Z)$) such that $U \subseteq V$ (respectively $U \supseteq V$). A border itemsets is an itemsets that is a member of the positive border or the negative border.

The objective of the hiding strategy is to reduce the support of each element in the set $Bd^{-}(R_s)$ below σ with a minimal effect on Bd^{+}(R-R_s). During the hiding process, each border element E in Bd^{+} is given a weight that is calculated based on its current support. The weight of a border element E in Bd^{+} is defined by the border based approach as follows:

Definition 1: Let frq(E) be the number of transactions that contain the itemsets E in D, given a database D and a border itemsets $E \in Bd^{+}$. Let $\tilde{D}$ be the database during the transformation process, and $\widetilde{frq(E)}$ be the number of transactions in $\tilde{D}$ that contains the itemsets E. The transformation begins with the initialization of $\tilde{D}$=D and $\widetilde{frq(E)}$ = frq(E). The weight of a border itemsets E in Bd^{+} is formulated in (3).

$$w(E) = \begin{cases} \dfrac{frq(E) - \widetilde{frq(E)} + 1}{frq(E) - \sigma * N} & , \widetilde{frq(E)} \geq \sigma * N + 1 \\ \lambda + \sigma * N - frq(E) & , 0 \leq \widetilde{frq(E)} \leq \sigma * N \end{cases} \quad (3)$$

If the border element E is more vulnerable, then a large weight assigned is to E. In the above dentition, the following points can be observed: 1) For a border itemsets E, when $\widetilde{frq(E)} > \sigma * N$, w(E) is no more than one. If $\widetilde{frq(E)} = \sigma * N$, a large integer λ is assigned to w(E), where $\infty > \lambda > |Bd^{+}|$. When the border itemsets E is about to become infrequent, w(E) is given a high value, indicating a low priority of being affected. The border element E must be avoided for further alteration if it is over-hidden $\widetilde{frq(E)} \leq \sigma * N$. Therefore, w(E) is decided by λ and the amount of frq(E) less than $\sigma * N$. 2) If $\widetilde{frq(E)} > \sigma * N + 1$, w(E) increases as frq(E) decreases with a the rate of $1/(frq(E) - \sigma * N)$. This checks the risk of losing the positive border itemsets and maintains the relative count among the border itemsets.

Given a sensitive transaction T_i, let U= {nsi_1, nsi_2, …, nsi_j, …, nsi_q} be a set of positive border elements such that $\forall nsi_j \in U \mid nsi_j \cap T_i \neq \emptyset$. Let U_x = {nsx_i, nsx_2, …, nsx_j, …, nsx_q} where $nsx_j = nsi_j \cap T_i$. The transaction weight of a sensitive transaction T_i is defined by (4).

$$TWeight(T_i) = \sum_{j=1}^{q} w(nsi_j) \times \left[log\left(\frac{|T_i| + 1}{|T_i| - |nsx_j| + 1} \right) \right] \quad (4)$$

The TWeight(T_i) determines the transaction's quality, which has a direct impact on the quality of the transformed database generated by the hiding process. The multiplicative factor $\log((|T_i| + 1)/(|T_i| - |nsx_j| + 1))$ increases or decreases the weight of the positive border element nsi_j while computing transaction weight based on the number of positive border elements present in the transaction. The greater the transaction's weight TWeight(T_i), the more susceptible it is to sanitization, and therefore the lower the priority of getting T_i impacted. The algorithm calculates the weight of each sensitive transaction and utilizes it to measure the possible side effects of transaction sanitization. The transaction with the lowest weight is considered for sanitization. Given the victim transaction T_v containing sensitive itemsets {si_1, si_2,.. si_j,…,si_p}, then hiding candidates set $C(T_v)$ is computed as $C(T_v) = \bigcup_{j=1}^{p} si_j$. Deleting a candidate item c affects only the border elements of T_v, which includes item c. For each hiding candidate item c in the victim transaction, the weights of the affected positive border elements added together to calculate the effect on the border. We assign a weight for each candidate item c of Tv which is defined by (5).

$$IWeight(c) = \sum_{E_i \in Bd^{+}|c} w(E_i) \quad (5)$$

If a candidate item c belongs to many sensitive itemsets, deletion of c results in few database modifications. Each time a candidate item with the smallest weight is chosen as the victim item for deletion. The deletion of selected victim item has minimal impact on border Bd^{+}. The transaction selection and victim item deletion steps are repeated while the set of sensitive itemsets is non-empty.

The proposed solution border-based rule hiding (BBRH) for hiding sensitive association rules is presented in Algorithm 1. The while loop in the algorithm iterates until the support of all elements in Bd^- are below the minimum support threshold σ. During each iteration of the while loop, for each sensitive transaction in the database, the transaction weight is computed using (4) and a transaction with the highest weight is selected for modification. If there are several transactions with the highest weight, they are sorted in increasing order of size and number of sensitive items, and the first transaction is chosen for alteration. The hiding candidate set C is initialized to sensitive items present in the chosen transaction for sanitization. The algorithm employs (5) to compute the possible impact of deleting hiding candidate c on positive border and the candidate item with the lowest weight is selected as victim item for deletion. The supports of border elements, weights of positive border elements that are affected by deletion of victim item are updated and the victim item is deleted from the database.

Algorithm 1. BBRH Algorithm

Compute Bd^+ and Bd^- from R and R_s repectively
Compute $w(E)$ $\forall E \in Bd^+$
while $Bd^- \neq \emptyset$
  $T_s = \{t \mid t \in D \wedge \exists si_j \in S \wedge si_j \subseteq t\}$
  for each transaction T_i *in* T_s

Compute transaction weight $TWeight(T_i)$
  end for
  $T_v = \{ t \mid \min_{t \in T_s} TWeight(t)\}$
  initialize C the set of hiding candidates of T_v
    for each hiding candidate c in C
      compute item weight $IWeight(c)$
    end for
    $I_v = \{I \mid \min_{I \in C} IWeight(I)\}$
  for each sensitive itemset si_j *in* Bd^-
      if $si_j \subseteq T_v$ *and* $I_v \in si_j$
        update the support of si_j
          if $support(si_j) < \sigma$
            delete si_j *from* Bd^-
    end for
    for each non − sensitive itemset nsi_j *in* Bd^+
      if $nsi_j \subseteq T_v$ *and* $I_v \in nsi_j$
        update the support of nsi_j
          update $w(nsi_j)$
    end for
  Delete the I_v *from transaction* $T_v \in D$
end while
output $D' = D$

5. PERFORMANCE EVALUATION AND RESULTS DISCUSSION

This section presents the results of experiments that we carried out on different real-world datasets to analyze the performance of our proposed hiding algorithm.

5.1. Data sets and performance metrics

We examined the performance of the proposed on two different real-world transaction datasets Mushroom and Chess form FIMI repository that are publicly available. Table 1 depicts characteristics of the overall datasets, where |D|, |I| and AvgSize respectively indicate the number of transactions, maximum size of an itemsets and, the average size of transactions.

Table 1. Dataset characteristics

Dataset	|I|	|D|	AvgSize
Chess	75	3,196	37.0
Mushroom	119	8,124	23.0

The algorithms used for experimentation conceals all the sensitive rules; hence, all the algorithms achieve 0% hiding failure. In order to evaluate effectiveness, we compare the results of the proposed algorithm with the results obtained using TF-IDF [10] and BRDA [11] algorithms. Algorithms BBRH, TF-

IDF, and BRDA were coded in R and executed on an Intel Pentium 4, 2.50 GHz processor with 4GB RAM running a 64-bit version of Windows 10. A series of experiments are carried out to measure the effectiveness of the hiding strategy using the following parameters i) dataset dissimilarity; ii) missing rules; iii) artificial rules.

Dataset Dissimilarity: The usefulness of the transformed database is measured by *Dataset Dissimilarity* as the difference between the input (D) and transformed database (D′) and defined by (6).

$$Dataset\ Dissiilarity = \frac{\sum_{i=1}^{m}(frq_D(i) - frq_{D'}(i))}{\sum_{i=1}^{m} frq_D(i)} \quad (6)$$

where $\mathrm{frq_D(i)}, \mathrm{frq_{D'}(i)}$ are the frequency of item *i* in the input and transformed database, respectively.

Missing Rules: Non-disclosure of some non-sensitive rules from the transformed database indicates that hiding of sensitive rules also hides few non-sensitive rules. The parameter *Missing Rule* estimates the percentage of non-sensitive rules that were accidentally hidden during the sanitization and is defined by (7).

$$Missing\ Rules = \frac{|R_{NS}(D)| - |R_{NS}(D')|}{|R_{NS}(D)|} \quad (7)$$

where $\mathrm{R_{NS}(D), R_{NS}(D')}$ are the set of non-sensitive rules discovered from the input and transformed database, respectively.

Artificial Rules: The mining of association rules on a transformed database may uncover rules that were not discovered from the original database. The parameter *Artificial Rules* measures the percentage of new rules that are discovered from the transformed database and is defined by (8).

$$Artificial\ Rules = \frac{|R(D') - (R(D) \cap R(D'))|}{|R(D')|} \quad (8)$$

where $\mathrm{R(D), R(D')}$ are the set of rules discovered from the input and transformed database, respectively.

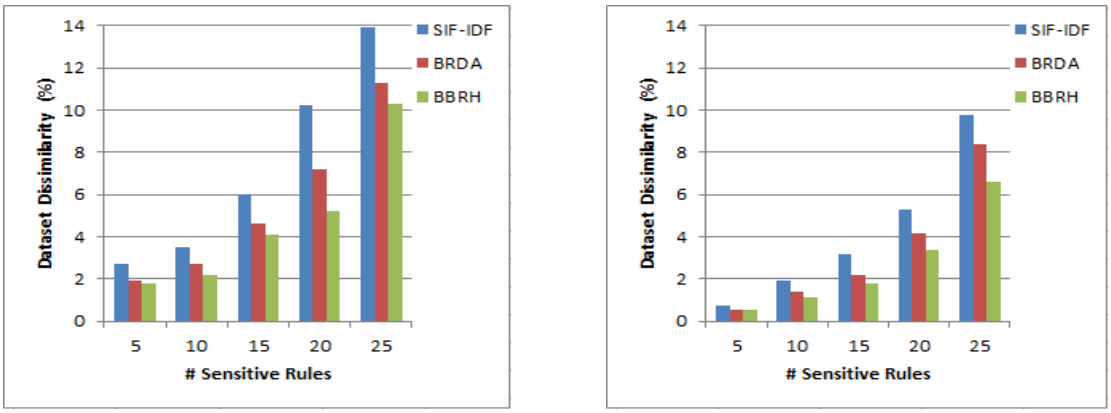
5.2. Data sets and performance metrics

The effectiveness of the sanitizing algorithm is measured in terms of side effects incurred on the dataset and association rule mining results include: *dataset dissimilarity*, *missing rules*, and *artificial rules*. The association rules were obtained with the specified threshold parameters as depicted in Table 2 using the ARM algorithm and as sensitive rules, a certain percentage of these association rules were chosen at random. The threshold values σ, δ, and percentage of sensitive association rules values were set for each data set to ensure that the number of association rules and sensitive association rules is adequate for experimentation.

Table 2. ARM results

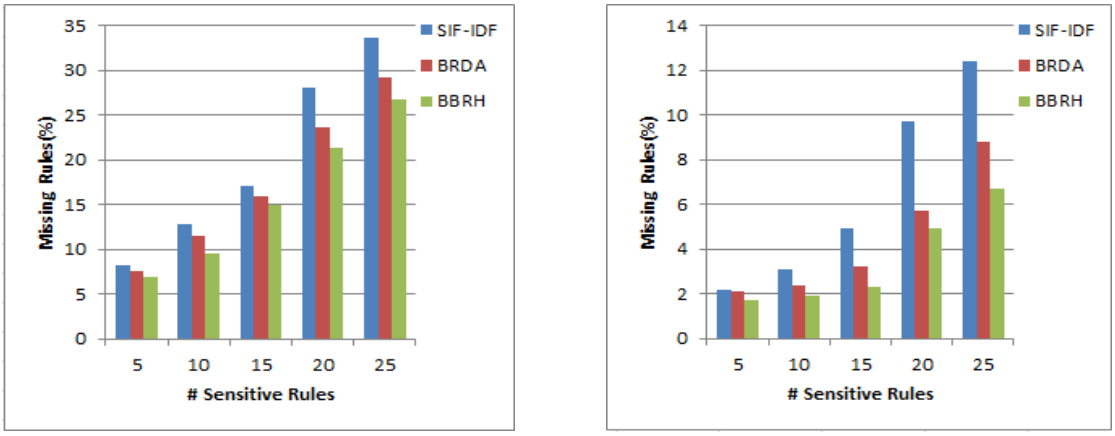
Dataset Name	σ	δ	# Association Rules
Chess	0.95	0.98	303
Chess	0.88	0.92	22085
Mushroom	0.40	0.70	3828
Mushroom	0.40	0.60	4570

From Figure 1, it is observed that improved results are obtained by the proposed algorithm in terms of database dissimilarity compared to TF-IDF and BRDA algorithms. Figure 2 depicts the results of Missing Rules of three algorithms by considering a fixed σ and δ for two different datasets. Figure 2 shows that as compared to the TF-IDF and BRDA algorithms, the proposed algorithm achieves better results in terms of missing rules. The results demonstrate that as the number of sensitive rules increases the percentage of dataset dissimilarity and missing rules also increases. The reason for this is as the number of sensitive rules increases, the number of dataset modifications also increases, and as a side effect of this is the increase in the loss of non-sensitive rules. The graphs in Figure 3 show that the number of artificial rules introduced by the BBRH algorithm less when compared to the TF-IDF algorithm and the number of artificial rules introduced by BBRH and BRDA algorithms are almost the same.



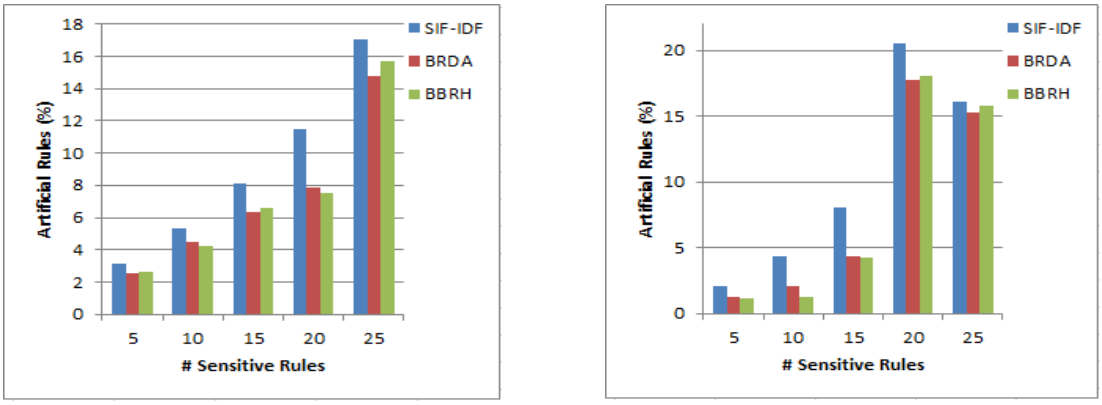
(a) (b)

Figure 1. Dataset dissimilarity for; (a) chess dataset and (b) mushroom dataset



(a) (b)

Figure 2. Missing rules for; (a) chess dataset and (b) mushroom dataset



(a) (b)

Figure 3. Artificial rules for; (a) chess dataset and (b) mushroom dataset

6. CONCLUSION

In this paper, we presented an algorithm that relies on border criterion for protecting sensitive association rules from disclosure. The proposed algorithm gradually reduces the amount of support for sensitive rules until they are hidden. The algorithm selects the transactions and items for sanitization such

that the modification of items in the identified transactions results in minimal side effects on the database. The proposed algorithm utilizes the idea of the border theory in order to minimize the impact on the positive border of itemsets which is produced while hiding the itemsets of sensitive association rules. We demonstrated through experiments that the proposed algorithm's results are of higher quality in terms of database side effects than those produced by other similar approaches.

REFERENCES

[1] R. Agrawal and R. Srikant, "Privacy-Preserving Data Mining," in *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, 2000, pp. 439-450, doi: 10.1145/342009.335438.

[2] V. S. Verykios, E. Bertino, I. N. Fovino, L. P. Provenza, Y. Saygin, and Y. Theodoridis, "State-of-the-art in Privacy Preserving Data Mining," *ACM SIGMOD Record*, vol. 33, no. 1, pp. 50-57, 2004.

[3] E. Bertino, I. N. Fovino, and L. P. Provenza, "A Framework for Evaluating Privacy Preserving Data Mining Algorithms," *Data Mining and Knowledge Discovery*, vol. 11, no. 2, pp. 121-154, 2005, doi: 10.1007/s10618-005-0006-6.

[4] A. Evfimievski, "Randomization in privacy preserving data mining," *ACM SIGKDD Explorations Newsletter*, vol. 4, no. 2, pp. 43-48, 2002, doi: 10.1145/772862.772869.

[5] B. Pinkas, "Cryptographic techniques for privacy-preserving data mining," *ACM SIGKDD Explorations Newsletter*, vol. 4, no. 2, pp. 12-19, 2002, doi: 10.1145/772862.772865.

[6] C. C. Aggarwal and P. S. Yu, "On Variable Constraints in Privacy Preserving Data Mining," in *Proceedings of the 2005 SIAM International Conference on Data Mining*, 2005, doi: 10.1137/1.9781611972757.11.

[7] M. Atallah, E. Bertino, A. Elmagarmid, M. Ibrahim, and V. Verykios, "Disclosure limitation of sensitive rules," in *Proceedings 1999 Workshop on Knowledge and Data Engineering Exchange (KDEX'99) (Cat. No.PR00453)*, 1999, pp. 45-52, doi: 10.1109/KDEX.1999.836532.

[8] E. Dasseni, V. S. Verykios, A. K. Elmagarmid, and E. Bertino, "Hiding Association Rules by Using Confidence and Support," *Information Hiding Lecture Notes in Computer Science*, pp. 369-383, 2001, doi: 10.1007/3-540-45496-9_27.

[9] C. N. Modi, U. P. Rao, and D. R. Patel, "Maintaining privacy and data quality in privacy preserving association rule mining," in *2010 Second International conference on Computing, Communication and Networking Technologies*, 2010, pp. 1-6, doi: 10.1109/ICCCNT.2010.5592589.

[10] T.-P. Hong, C.-W. Lin, K.-T. Yang, and S.-L. Wang, "Using TF-IDF to hide sensitive itemsets," *Applied Intelligence*, vol. 38, no. 4, pp. 502-510, 2012, doi: 10.1007/s10489-012-0377-5.

[11] P. Cheng, I. Lee, J.-S. Pan, C.-W. Lin, and J. F. Roddick, "Hide Association Rules with Fewer Side Effects," in *IEICE Transactions on Information and Systems*, vol. E98.D, no. 10, pp. 1788-1798, 2015, doi: 10.1587/transinf.2014edp7345.

[12] N. J. Ghalehsefidi and M. N. Dehkordi, "A Hybrid Algorithm based on Heuristic Method to Preserve Privacy in Association Rule Mining," *Indian Journal of Science and Technology*, vol. 9, no. 27, pp. 1-10, 2016, doi: 10.17485/ijst/2016/v9i27/97476.

[13] A. Telikani and A. Shahbahrami, "Optimizing association rule hiding using combination of border and heuristic approaches," *Applied Intelligence*, vol. 47, no. 2, pp. 544–557, 2017, doi: 10.1007/s10489-017-0906-3.

[14] S. -. Wang and A. Jafari, "Using unknowns for hiding sensitive predictive association rules," *IRI -2005 IEEE International Conference on Information Reuse and Integration, Conf, 2005.*, 2005, pp. 223-228, doi: 10.1109/IRI-05.2005.1506477.

[15] H. Pang and B. Wang, "Privacy-Preserving Association Rule Mining Using Homomorphic Encryption in a Multikey Environment," *IEEE Systems Journal*, vol. 15, no. 2, pp. 3131-3141, June 2021, doi: 10.1109/JSYST.2020.3001316.

[16] X. Chen, M. Orlowska, and X. Li, "A new framework for privacy preserving data sharing," in *4th IEEE ICDM Workshop on Privacy and Security Aspects of Data Mining, IEEE Computer Society*, 2004, pp. 47-56.

[17] Y. H. Guo, "Reconstruction-based association rule hiding," in *SIG-MOD Workshop IDAR*, Beijing, China, 2007, pp. 51-56.

[18] S. Li, N. Mu, J. Le, and X. Liao, "Privacy preserving frequent itemset mining: Maximizing data utility based on database reconstruction," *Computers & Security*, vol. 84, pp. 17-34, 2019, doi: 10.1016/j.cose.2019.03.008.

[19] M. N. Dehkordi, K. Badie, and A. K. Zadeh, "A Novel Method for Privacy Preserving in Association Rule Mining Based on Genetic Algorithms," *Journal of Software*, vol. 4, no. 6, 2009, doi: 10.4304/jsw.4.6.555-562.

[20] A. Khan, M. S. Qureshi, and A. Hussain, "Improved genetic algorithm approach for sensitive association rules hiding," *World Appl. Sci. Journal*, vol. 31, no. 12, pp. 2087-2092, 2014.

[21] C.-W. Lin, B. Zhang, K.-T. Yang, and T.-P. Hong, "Efficiently Hiding Sensitive Itemsets with Transaction Deletion Based on Genetic Algorithms," *The Scientific World Journal*, vol. 2014, pp. 1-13, 2014, doi: 10.1155/2014/398269.

[22] C.-W. Lin, T.-P. Hong, K.-T. Yang, and S.-L. Wang, "The GA-based algorithms for optimizing hiding sensitive itemsets through transaction deletion," *Applied Intelligence*, vol. 42, no. 2, pp. 210-230, 2014, doi: 10.1007/s10489-014-0590-5.

[23] J. C.-W. Lin, Y. Zhang, P. Fournier-Viger, Y. Djenouri, and J. Zhang, "A Metaheuristic Algorithm for Hiding Sensitive Itemsets," *Lecture Notes in Computer Science Database and Expert Systems Applications*, pp. 492–498, 2018, doi: 10.1007/978-3-319-98812-2_45.
[24] M. H. Afshari, M. N. Dehkordi, and M. Akbari, "Association rule hiding using cuckoo optimization algorithm," *Expert Systems with Applications*, vol. 64, pp. 340–351, 2016, doi: 10.1016/j.eswa.2016.08.005.
[25] K. Doan, M. N. Quang, and B. Le, "Applied Cuckoo Algorithm for Association Rule Hiding Problem," in *Proceedings of the Eighth International Symposium on Information and Communication Technology*, 2017, doi: 10.1145/3155133.3155150.
[26] P. T. Selvan and S. Veni, "Social Ant based Sensitive Item Hiding with Optimal Side Effects for Data Publishing," *Indian Journal of Science and Technology*, vol. 9, no. 2, 2016, doi: 10.17485/ijst/2016/v9i2/81159.
[27] N. K. Bux, M. Lu, J. Wang, S. Hussain, and Y. Aljeroudi, "Efficient Association Rules Hiding Using Genetic Algorithms," *Symmetry*, vol. 10, no. 11, p. 576, 2018, doi: 10.3390/sym10110576.
[28] A. Telikani, A. H. Gandomi, A. Shahbahrami, and M. N. Dehkordi, "Privacy-preserving in association rule mining using an improved discrete binary artificial bee colony," *Expert Systems with Applications*, vol. 144, p. 113097, 2020, doi: 10.1016/j.eswa.2019.113097.
[29] H. Mannila and H. Toivonen, "Levelwise search and borders of theories in knowledge discovery," *Data Mining and Knowledge Discovery*, vol. 1, pp. 241-258, 1997.

BIOGRAPHIES OF AUTHORS

Suma B is working as Assistant Professor at Computer Science and Engineering Department, RV College of Engineering, Bengaluru, since 2010. She has received her master's degree from NITK, Surathkal and pursuing her Ph.D. (CSE) from VTU. Her research areas of interest are Algorithm Design, Data Mining.

Shobha G. is a professor in Computer Science and Engineering Department and associated RV College of Engineering, Bengaluru, since 1995. She has received her master's degree from BITS, Pilani and Ph.D. (CSE) from Mangalore University. Her research areas of interest are database management systems, data mining, data warehousing, image processing and information and network security.