Privacy Preserving Association Rules by Using Greedy Approach

Chieh-Ming Wu, Yin-Fu Huang and Jian-Ying Chen

Graduate School of Engineering Science and Technology
National Yunlin University of Science and Technology
123 University Road, Section 3,
Touliu, Yunlin, Taiwan 640, R.O.C.
Tel: (+886)-5-5342601 Ext. 4314
Fax: (+886)-5-5312063
Email: huangyf@el.yuntech.edu.tw

Abstract

Data mining techniques have been developed in many applications. However, they also cause a threat to privacy. In this paper, we proposed a greedy method for hiding the number of sensitive rules. The experimental results showed that the undesired side effects can be avoided in the rule hiding process by use of our approach. The results also revealed that in most cases, all the sensitive rules are hidden without generating spurious rules. First, the good scalability of our approach in terms of database sizes is achieved by using an efficient data structure FCET to store solely maximal frequent itemsets rather than the entire frequent itemsets. Furthermore, we proposed a new framework for enforcing the privacy in mining association rules, that combine the techniques for efficiently hiding sensitive rules and the transaction retrieval engine based on the FCET index tree. In particular, four strategies are implemented in the sanitized procedure, for hiding a group of association rules characterized as sensitive or artificial rules.

Keywords—Greedy methods, FCET, maximal frequent itemsets, sanitized procedure, rule hiding

1. Introduction

In the data mining, the most well-known method is the Apriori algorithm. Since the Apriori algorithm is costly to find candidate itemsets, there are many variants of Apriori that differ in how they check candidate itemsets against the database [4]. For example, Zaki et al. [9] presented the algorithms MaxEclat and CHARM for identifying maximal frequent itemsets. Furthermore, in [8] employed an efficient data structure FCET and a novel algorithm GMAR to mine generalized association rules.

Recently, disclosure limitation of sensitive knowledge by data mining algorithms and the association rule retrieval has also been investigated [1][2][3][5]. The authors in [6] proposed preventing disclosure of sensitive knowledge by decreasing the significance of the rules using some heuristics which can be regarded as the precursors to the heuristics. They demonstrated that solving this problem by reducing the support of the large itemsets via removing items from transactions (also referred to as "sanitization" problem) is an NP-hard problem.

In the paper, we removed the assumption mentioned above and allowed users to select sensitive rules from all strong rules. Given a transactional database, MST, MCT, and a set of sensitive rules, how can we modify the database using the same MST, MCT, and another set of strong rules so as to satisfy all the constraints: 1) no sensitive rules, 2) no artificial rules, and 3) no missing rules?

Since violating the last two constraints results in the production of side effects, only focusing on the first constraint as the work done in [6][7][8] is inadequate for certain applications. For some applications, they are only interested in hiding certain sensitive rules containing given items. The reason is that, once a sensitive rule is mined from other rivals, it may cause serious loss in an enterprise. Furthermore, a supplier cannot enhance their goods supply if the corresponding rule is falsely hidden. In medical applications, a misleading rule falsely generated would threaten human lives. Therefore, in the paper, we aimed at avoiding the side effects in the rule hiding process. Afterwards, the experimental results showed that the undesired side effects are avoided by using our approach. Here, we proposed the greedy approach to reduce the side effects and integrate four lemmas with a cost and weight mechanism to achieve no hiding failure without artificial rules.

The remainder of the paper is organized as follows. In Section 2, the Privacy-Preserving problem resulted from

978-0-7695-3507-4/08 $25.00 © 2008 IEEE
DOI 10.1109/CSIE.2009.812

mining association rules was defined. Then, an efficient framework is proposed for privacy preservation in Section 3, where two stages are included. In Section 4, the greedy approach for rule hiding is described. Section 5 depicts several experimental results showing the superiority of the greedy approaches over other algorithms. Finally, conclusions are made in Section 6.

2. Problem Formulation

Let $\Gamma=\{i_1,i_2,i_3\ldots i_n\}$ be a set of literals, called items. Let $D=\{T_1,T_2,\ldots T_m\}$ be a set of transactions which is the database that is going to be disclosed. Each transaction $T_j \in D$ is a subset of Γ of itemsets, such that $T \subseteq \Gamma$. For each transaction T_j is associated with a unique identifier, which we call TID. We assume that the items in a transaction or an itemset are sorted in lexicographic order.

Denoting the set of transactions in the database by T and the set of items in the database by Γ, an association rule is a implication of the form X=>Y, where $X,Y \subset \Gamma$ and $X \cap Y = \varnothing$. A rule X=>Y is said to have a support factors iff at least s% of the transactions in T satisfy $X \cup Y$. A rule X=>Y is satisfied in the set of transactions T with a confidence factor c iff at least c% of the transactions in T that satisfy X also satisfy Y. Both support and confidence are fractions in the interval [0,1]. The support is a measure of statistical significance, whereas confidence is a measure of the strength of the rule. A rule is said to be "interesting" if its support and confidence are greater than user-defined thresholds minimum support threshold (MST) and minimum confidence threshold (MCT), respectively, and the objective of the mining process is to find all such interesting rules. In this paper, our goal is to hide a group of interesting rules which contains highly sensitive knowledge. We refer to these rules as restrictive rules and we define them as follows:

Definition 1： Let **D** be a transactional database, **R** be a set of all the strong rules that can be mined from **D**, and Rules **S** be a set of sensitive rules that need to be hidden according to some security policies. A set of rules, denoted by **R'**, is all the strong rules after rule hiding. **SR** is all the sensitive rules that fail to be hidden. **MR** is all the missing rules after rule hiding and **AR** is all the artificial rules after rule hiding.

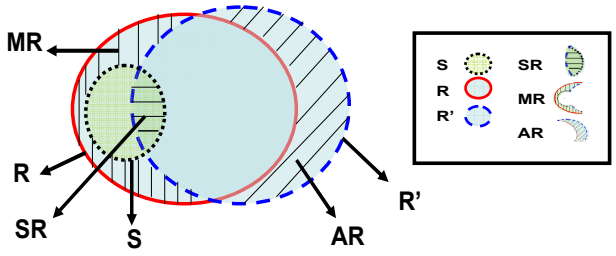


Fig.1 Visual representation of restrictive and nonrestrictive rules and the rules effectively discovered after sanitization.

Figure 1 illustrates the relationship between the rules R of all association rules in the database D, the restrictive and non-restrictive rules, as well as the set R' of strong rules discovered from the sanitized database D'. SR, MR, and AR are sets of the potential rules that represent the restrictive rules that were failed to be hidden, the legitimate rules accidentally missed, and the artificial riles created by the sanitization process. A group of restrictive rules is mined from a database D based on a special group of transactions. We refer to these transactions as sensitive transactions and define them as follows.

Definition 2： Let **T** be a set of all transactions in a transactional database **D** and **S** be a set of restrictive rules mined from **D**. A set of transactions is said to be sensitive, as denoted by **ST**, if **ST** $\subset$ **T** and only if all restrictive patterns can be mined from **ST** .

If D is the source database of transactions and R is a set of relevant strong rules that could be mined from D, the goal is to transform D into a database D' so that the most strong rules in {R-S} can still be mined from D' while the sensitive rules are all hidden (no SR) and no missing rules MR and no artificial rules AR are generated as many as possible. In this case, D' becomes the released database.

3. The Framework for Privacy Preservation

As depicted in Figure 2, our framework consists of an original database, mining algorithm, FCET index tree, sanitized algorithm used for hiding restrictive association rules from the database, and a transaction retrieval procedure which is included in sanitizing algorithm for fast retrieval of transactions.

There are two stages in the framework. In the stage1, it is supposed that the components shown in the stage 1, such as mining algorithms, frequent itemsets, and FCET index tree[8], were generated before our sanitize algorithm is executed. Rather than scanning the original database, we make use of the FCET index tree transformed from the frequent itemsets to speed up retrieval of association rules and transactions. In the stage 2, the sanitizing algorithm for transactional databases also includes some lemmas to remove or add information with the least impact on the database. In order to search for sensitive transactions in the transactional database, it is necessary to access, manipulate, and query transaction IDs and items. The retrieval procedure of transactions [8] performs theses tasks included in the sanitizing algorithm. We describe the stage 1 in this section and the stage 2 in Section 4.

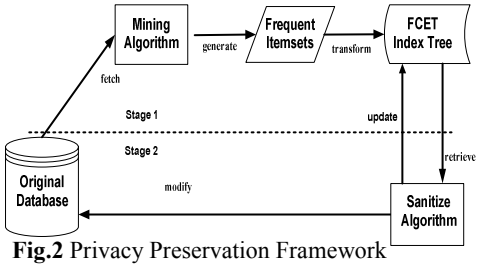


Fig.2 Privacy Preservation Framework

4. The Greedy Approach

As depicted in Figure 3, our greedy approach include the sanitize procedure, we also use extra two sets that are artificial set and missing set where the values in sets store for artificial rules and missing rules, respectively. Initially, the sensitive rules are stored in the sensitive set. The missing set and artificial set are empty. First, we use the sanitized procedure to let the sensitive set to empty and once a sensitive rule was

hidden then it will never appears. The sanitized procedure will execute repeat until the sensitive set is empty. Second, if the artificial set is not empty, then the sanitized will execute repeat until the artificial set is empty. Finally, if the sensitive set and artificial set are all empty then judge whether the missing set is empty. If the missing set is empty then stop the approach execution and get the perfect solution (sensitive set, artificial set and missing set are all empty). We also define the NTH ratio=|LR|/|U-H| is the percentage of the nonsensitive rules falsely hidden, where LR is all the loss rules, H and U be the sets of all sensitive rules and all strong rules in the original database, respectively. If the NTH ratio is small than an error constant ($0 \le \varepsilon \le 1$) it will also stop the approach execution.

Given a rule r, for each sanitized procedure we have four lemmas to select. How much transactions are chosen to hide sensitive rule then based on the lemmas.

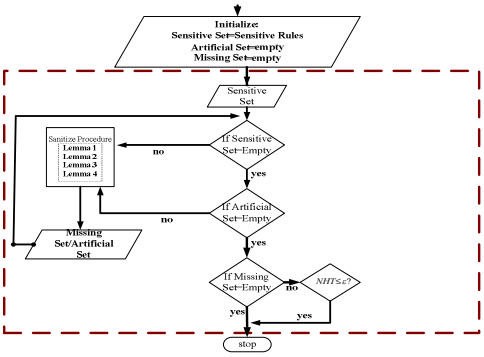


Fig. 3 Greedy Approach

4.1 Sanitize Procedure

The sanitize procedure uses the four lemmas to sanitize the sensitive rules and artificial rules, for example, given a rule $l => s$, for different lemmas, we will have corresponding strategies. In the lemma 1, the strategy is to find the transactions that contain the $\{ l \cup s \}$ itemsets, then we remove the subset of $\{ s \}$ itemsets from those transactions to decrease the support count of $\{ l \cup s \}$ itemsets. In the lemma 2, the strategy is to find the transactions that support the $\{ l \cup s \}$ itemsets and remove the $\{ l \}$ item from those transactions to decrease the support count of $\{ l \cup s \}$ and $\{ l \}$ itemsets. In the lemma 3, the strategy is to insert a new transaction into the database to increase the support count of $\{ l \}$ itemset. In the lemma 4, we are not only decrease the support count of $\{ l \cup s \}$, but also increase the support count of $\{ l \}$ itemset.

Lemma 1. Given a rule r, Greedy approach performs $\min\left(\lceil N_r - N_{lr} \times \min\_conf \rceil, \lceil N_r - |D| \times \min\_supp \rceil\right)$ executions of the count $\mathbf{C}_i$, where i is determined by which lemma is used.

Lemma 2. Given a rule r, Greedy approach performs

$$\min\left(\left\lceil \frac{N_r - \min\_conf \times N_{lr}}{1 - \min\_conf} \right\rceil, \lceil N_r - |D| \times \min\_supp \rceil\right)$$

executions of the count $\mathbf{C}_i$, where i is determined by which lemma is used.

Lemma 3. Given a rule r, Greedy approach performs $\left\lceil \frac{N_r}{\min\_conf} - N_{lr} \right\rceil$ executions of the count $\mathbf{C}_i$, where i is determined by which lemma is used.

Lemma 4. Given a rule r, Greedy approach performs

$$\min\left(\left\lceil \frac{N_r - \min\_conf \times N_{lr}}{1 + \min\_conf} \right\rceil, \lceil N_r - |D| \times \min\_supp \rceil\right)$$

executions of the count $\mathbf{C}_i$, where i is determined by which lemma is used.

4.2 Greedy Approximation Approach

The sanitized problem is from the some selected transactions (**T**), removed or added some itemsets (**S**) to hide all sensitive rules and make the number of missing rules and artificial rules to be minimum as much as possible. Those itemsets of removed or added come according to lemmas that we have proposed and the cost and weight calculation is defined as follows.

Definition 3：cost=r*artificial cost+(1-r)*missing cost, where the artificial cost is the number of artificial rules and the missing cost is the number of missing rules after the execution of lemmas, and r is a ratio of artificial cost and missing cost, where $0 \le r \le 1$.

Definition 4：weight=(# of hided artificial rules+# of exposed of missing rules)/cost, where the # of hided artificial rules is the number of hided artificial rules from the artificial set and # of exposed of missing rules is the number of exposed missing rules from the missing set after the execution of lemmas.

Now, for each transaction t_i in **T,** where $0 < i \le |T|$ and for each item s_j in **S** where $0 < j \le |S|$, **T** is the set of some selected transactions based on the selected lemma and **S** is a set of itemsets which will be removed or added based on the selected lemma. We are given the number $\cos t_{i,j} X_{i,j}$ is $|\bigcup_{i=1}^{|T|} T_i Y_i| = C$ minimal subject to, where $X_{i,j} = 0$ or 1 and Y_i =0 or 1, finally, we select the maximal weight among of lemmas execution to hide a sensitive rule.

In the greedy approximation approach, some items need to add/remove from the transactions, is based on the different lemmas with the different strategies adopted.

5. Performance Evaluations

5.1 Simulation Model

In this section, we evaluate the performances of five algorithms including greedy approximation, greedy exhausted, random, algorithm 1 [6] and algorithm 2 [6] on a DELL GX 270 with Intel Pentium 4 3.2Ghz and 1 GB main memory running Windows XP. All the experimental data are generated from the IBM synthetic data generator [10]. The performance of the developed algorithms has been measured according to two criteria: time requirements and side effects produced. For the former criteria, we considered the time needed by each algorithm to hide a specified set of rules, while, for the latter one, the number of "missing" rules(MR) and the number of "artificial" rules(AR) indicated in Figure 1. Hiding rules

produce some changes in the original database as well as changes affecting the set of rules mined.

5.2 Experimental Results

In order to explain the experimental results with ease, we explore the required association rules of Apriori-like algorithm in the dataset, as shown in Fig. 4. By looking the trend of the figure, the distribution of the strong rules can be found. Then, we encode the strong rules into a bit-vector table and apply the cluster algorithm to divide the strong rules into twenty clusters. Furthermore, for the purpose of dealing with the problems of the sensitive rules with high correlation, some definitions are proposed and described as follows.

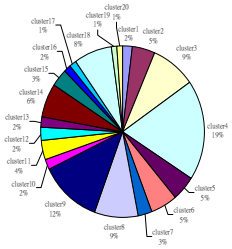
Definition 5: Intra correlation degree. The intra correlation of a set of sensitive rules S means that the item occurring in the precedent or the consequent of the sensitive rules appears in the same cluster. We define the intra correlation degree of a rules **r** as

$$\log_w \Big(\sum_{item \in r} occurence_{item\ in\ same\ cluster} \Big)$$

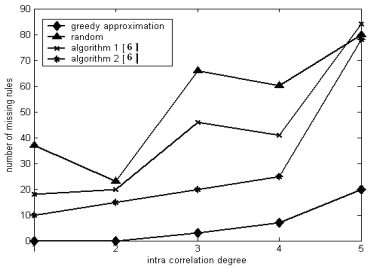
Definition 6: Inter correlation degree. The inter correlation of a set of sensitive rules S means that the item occurring in the precedent or the consequent of the sensitive rules appears in the different cluster. We define the inter correlation degree of a rules r as

.

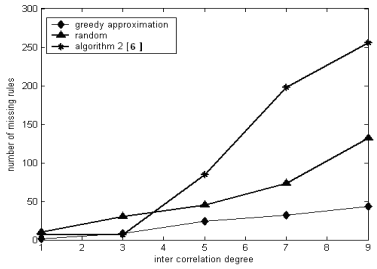
$$\log_w \Big(\sum_{item \in S} occurence_{item\ in\ different\ cluster} \Big)$$



(a)



(b)

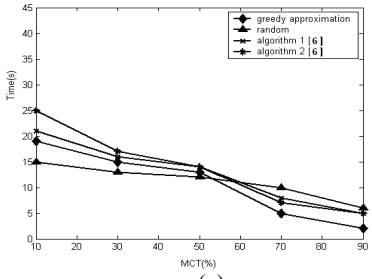


(c)

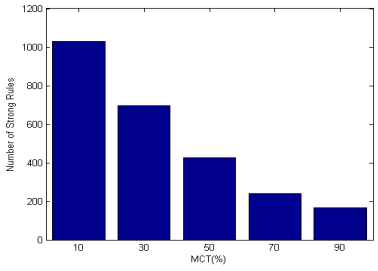
Fig.4 Performance on the intra/inter correlation degree.

To more precisely investigate the relation of intra/inter correlation degree with number of missing rules from Fig. , we use the database with size of 10K producing more than 1000 strong rules. The strong rules generated are then divided into 20 clusters as shown in (a) of Fig. 4. The number of missing rules over inter/intra correlation degree are shown in (b) and (c) of Fig. 4 with ω=2. The results reveal that higher intra/inter correlation degree will get higher number of missing rules.

In Fig.4, only one sensitive rule is used to join the hiding process. The algorithm 1 [6] hides the sensitive rules according to the increasing trends of the support of rule's antecedent until the rule confidence decreases below MCT. The algorithm 2 [6] hides the sensitive rules by decreasing the support of their generating itemsets until their support is below MST. Because both the algorithm 1 [6] and algorithm 2 [6] neglect the intra/inter correlation and assume that the sensitive rules are supported by disjoint frequent itemsets. In our experiments, each set is not disjoint.



(a)



(b)

Fig.5 Scalability on the number of strong rules.

To observe the impact of the number of strong rules, the results on the CPU time of our approach under various MCT are plotted in Fig.5. (a). Since applying more strong rules leads to higher cost on checking the constraints of all the algorithms, the greedy approximations utilize the four lemmas in sanitized process to hide sensitive rules with a greedy approach. To choose the best lemma to execute, the greedy approximation algorithm will spend less time than the other

algorithms, since we use an efficient data structure (FCET & IFCET) [8] and a hash function [4] to store and get the relative information to reduce the time required. In Fig.5 (b), we also show how the number of strong rules varies as the growth of MCT.

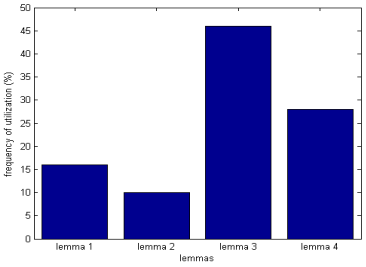


Fig.6 the frequency of utilization of lemmas

To evaluate the frequency of utilization shown in Fig.6, we execute the greedy approximation algorithm 1000 times and count the rate of utilization of each lemma on randomly selecting one sensitive rule for each execution. We found the lemma 3 has the highest rate of utilization in the sanitized procedure.

6. Conclusions

In this paper, we present a greedy approach in order to protect sensitive rules from disclosure. The approach prevents rules by use of a greedy approximation algorithm generated by hiding the rules and setting their confidence or support below a user-specified threshold. The selection of the lemmas to be adopted depends on which weight is the largest. The lemma with the largest weight will cause the lowest side effects. No assumptions have been made for the development of the proposed algorithms. Therefore, the proposed algorithm is executed under a condition of no limitations and assumptions that is different from the works of the other researches [6] [7].

The experimental results show that our approach is scalable in terms of database size. In all cases, all the sensitive rules are hidden without false rules generated. In addition, it is observed that the intra-correlation degree and the inter-correlation degree among sensitive rules have a significant impact on the performance of rule hiding. We also use an efficient data structure (FCET) [8] to speed up the time of hiding processing. FCET stores all the information to avoid rescanning the database so that the transactions or itemsets from desired procedure can be easily retrieved.

As confirmed by the experimental results, the proposed algorithm possesses a desirable feature of producing no artificial rules and causing no hiding failure. Moreover, the experimental results have shown that our algorithm always generates lower number of missing rules than the other algorithms. Hence, the future development will focus on reducing the number of missing rules to further enhance the performance of the proposed algorithms.

6. References

[1]. M. Atallah, E. Bertino, A. Elmagarmid, “Disclosure Limitation of Sensitive Rules” Proc. of IEEE Knowledge and Data Engineering Workshop, 1999, pp.45-52.

[2]. R.Agrawal, R. Srikant, "Privacy-preserving data mining", ACM SIGMOD Record archive Volume 29 , Issue 2, 2000, pp.439-450.

[3]. Wai-Chee Fu, Raymond Chi-Wing Wong, Ke Wang, "Privacy-Preserving Frequent Pattern Mining across Private Databases," Fifth IEEE International Conference on Data Mining (ICDM'05), 2005, pp. 613-616.

[4]. Yin-Fu Huang and Chieh-Ming Wu, “Mining generalized association rules using pruning techniques,” Proc. IEEE International Conference on Data Mining, 2002, pp. 227-234.

[5]. Y. Saygin, V.S. Verykios, and C. Clifton, “Using Unknowns to Prevent Discovery of Association Rules,” *ACM SIGMOD Record,* vol. 30, no. 4,, 2001, pp. 45-54.

[6]. V.S. Verykios, A.K. Elmagarmid, E. Bertino, Y. Saygin, and E. Dasseni, “Association Rule Hiding,” IEEE Trans. Knowledge and Data Eng., vol. 16, no. 4, 2004, pp. 434-447.

[7]. Yi-Hung Wu, Chia-Ming Chiang, Arbee L.P. Chen, "Hiding Sensitive Association Rules with Limited Side Effects," IEEE Transactions on Knowledge and Data Engineering, vol. 19, no. 1, pp. 29-42, Jan., 2007

[8]. Chieh-Ming Wu, Yin-Fu Huang, "An Efficient Data Structure for Mining Generalized Association Rules", The 5th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD'08), volume 2,2008,pp.565-571

[9]. M.J. Zaki, C.-J. Hsiao, “Efficient algorithms for mining closed itemsets and their lattice structure” IEEE Transactions on Knowledge and Data Engineering, Volume 17, Issue 4, April ,2005, pp. 462-478.

[10]. http://www.almaden.ibm.com/software/quest/Resources