Notice of Violation of IEEE Publication Principles

"A Novel Method for Protecting Sensitive Knowledge in Association Rules Mining"
by Tianding Chen
in the 2006 Proceedings of the Sixth International Conference on Intelligent Systems Design and Applications (ISDA'06)

After careful and considered review of the content and authorship of this paper by a duly constituted expert committee, this paper has been found to be in violation of IEEE's Publication Principles.

This paper contains significant duplication of original text from the paper cited below. The original text was copied without attribution (including appropriate references to the original author(s) and/or paper title) and without permission.

Due to the nature of this violation, reasonable effort should be made to remove all past references to this paper, and future references should be made to the following two articles:

"A Novel Method for Protecting Sensitive Knowledge in Association Rules Mining"
by En Tzu Wang, Gualin Lee and Yu Tzu Lin
in the 2005 Proceedings of the 29th Annual International Computer Software and Applications Conference (COMPSACC'05)

and

"A Novel Method for Protecting Sensitive Knowledge in Association Rules Mining"
by En Tzu Wang
in a Masters thesis for Dept. Of Computer Science and Information Engineering, National Dong Hwa University, Taiwan, June 2005

http://etd.lib.ndhu.edu.tw/theses//available/etd-0716105-171246/unrestricted/etd-0716105-171246.pdf

A Novel Method for Protecting Sensitive Knowledge in Association Rules Mining

Tianding Chen
Institute of Communications and Information Technology,
Zhejiang Gongshang University, Hangzhou, China 310035
chentianding@163.com

Abstract

In the researches of data mining, discovering frequent patterns from huge amounts of data is one of the most studied problems. The frequent patterns mined form databases can bring the users many commercial benefits. However, some sensitive patterns with security concerned may cause a threat to privacy. It investigates to find an appropriate balance between a need for privacy and information discovery on frequent patterns. It proposes a novel method for modifying databases to hide sensitive patterns. By multiplying the original database and a sanitization matrix together, a sanitized database with privacy concerns is obtained. Additionally, two probability policies are introduced to against the recovery of sensitive patterns and reduce the probability of hiding non-sensitive patterns in the sanitized database. The complexity analysis of our sanitization process is proved and a set of experiments is also performed to show the benefit of our approach.

1. Introduction

In recent years, more and more researches emphasize the seriousness of the problems about privacy. Data privacy problems focus on the privacy of sensitive data[1][2][3][4][5][6][7][8]. On the other hand, information privacy problems focus on hiding the association rules or frequent patterns which contain highly sensitive knowledge.

In order to avoid Forward-Inference Attack problems, according to the solution discussed in [9], at least one sup-pattern with length 2 of the pattern should be removed or the hiding pattern will be inferred recursively. In [10], the idea of using correlation matrix for hiding sensitive patterns is introduced. However, only the maximal patterns are considered in the approach, it doesn't apply to all the frequent patterns. In [11], the authors investigate confidentiality issues of a broad category of association rules and present three strategies and five algorithms for hiding sensitive rules. Although these algorithms ensure privacy preserving, they may modify true data values and relationships by adding new items into the original transactions. In [12], a matrix-based sanitization approach is proposed to hide the sensitive patterns.

In this paper, it proposes a novel method to extend [12] for modifying database to hide sensitive patterns. By observing the relationships between sensitive patterns and non-sensitive patterns, a sanitization matrix is defined. By setting the entries in sanitization matrix to appropriate values and multiplying the original transaction database to the sanitization matrix, a sanitized database which can resist Forward-Inference Attack is gotten. The sanitized database is the database that has been modified for hiding sensitive patterns with some privacy concerns. Moreover, we use some probability policies which are the first time to be introduced into this kind of approach with a level of confidence given by users to against the recovery of sensitive pattern and reduce the probability of hiding non-sensitive patterns in the sanitized database. The complexity analysis of our sanitization process is also proved in this paper.

2. Preliminary

The problem of discovering association patterns is defined as finding relationships between the occurrences of items within transactions.

In this approach, a transactional database is represented as a binary matrix D where the rows represent transactions and the columns represent items. Entry D_{ij} is set to 1, if item j is purchased in transaction i and 0, otherwise. Moreover, the frequent patterns with length 1 (frequent items) are not taken into account. Therefore, the problem of hiding sensitive patterns can be formulated as follows, let P be the set of all frequent patterns mined from D except the patterns with length 1, P_H be the set of sensitive patterns, $\sim P_H$ be the set of remainder frequent patterns (non-sensitive patterns), i.e. $P_H \cup \sim P_H = P$. Our approach is to transform D into a sanitized database D', such that only the patterns belong to $\sim P_H$ can be mined from D'. Moreover, the patterns belong to P_H will never suffer from Forward-Inference Attacks discussed in the previous section.

In our approach, the original database D is multiplied by a sanitization matrix S to get a sanitized database D'. That is, $D'_{n\times m} = D_{n\times m} S_{m\times m}$. If S is an identity matrix (i.e., $S_{ij}=1$ if $i=j$, otherwise, $S_{ij}=0$), D' will be equal to D. By setting S_{ij} where $i \neq j$ to the appropriate value, a sanitized database will be gotten. In the following, the basic concept of our approach is discussed.

In order to fit the properties of the binary matrix which is the representation of the transactional database, some different definitions of the matrix multiplication are given as follows:

1. If $D_{ij}=0$, D'_{ij} is set to 0 directly. This is because our goal is to hide sensitive patterns by decreasing their supports, therefore, we only need to take care of how and when an entry with its value equal to 1 in D should be converted to 0 in D'. Moreover, it also guarantees that there are no new artificial patterns created by the sanitization process.

2. If the resulting value of $\sum_{k=1}^{m} D_{ik} \cdot S_{kj}$ is not smaller than 1, set D'_{ij} to 1.

3. If the resulting value of $\sum_{k=1}^{m} D_{ik} \cdot S_{kj}$ is not larger than 0, set D'_{ij} to 0.

3. Sanitization process

Fig. 1 shows the flowchart of our approach. There are several components in the sanitization process. Each of them will be discussed in this section.

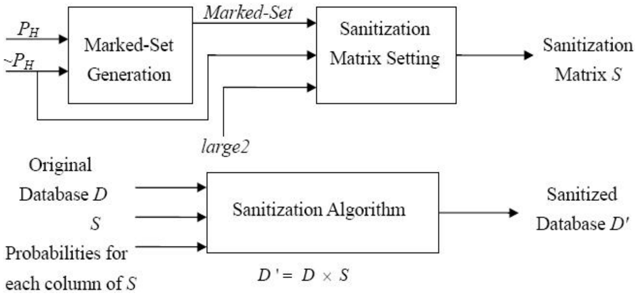


Figure 1. The flowchart of the sanitization process

3.1. Sanitization matrix setting

As discussed in section 1, to avoid Forward-Inference Attack, for each sensitive pattern P in P_H, at least one pattern belong to the pair-subset of P should be hidden or the attackers can infer from the subset with length 2 to the sensitive pattern recursively. The pair-subset is defined as follows:

Definition 1: Let F be a frequent pattern. The pair-subpattern of F is a sub-pattern of F with length 2. The set which includes all pair-subpatterns of F is called the pair-subset of F.

In our approach, a temporary set, Marked-Set, is used to store the victim pair-subpatterns. A victim pair-subpattern is the pair-subpattern selected from the pair-subset of a sensitive pattern and needed to be hidden. After setting up Marked-Set, all patterns in Marked-Set are used to set the corresponding entries to –1 in S.

3.2. Probability policies

3.2.1. Distortion probability ρ

By setting –1 in sanitization matrix, the supports of the pair-subpatterns belong to P_H can be decreased. However, refer to Fig. 2, it brings the following problem.

$$\underset{D}{\begin{pmatrix} 0 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 1 \end{pmatrix}}_{4\times 3} \times \underset{S}{\begin{pmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}}_{3\times 3} = \underset{D'}{\begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix}}_{4\times 3}$$

$$And \quad \underset{D}{\begin{pmatrix} 0 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 1 \end{pmatrix}}_{4\times 3} \times \underset{S}{\begin{pmatrix} 1 & -1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}}_{3\times 3} = \underset{D'}{\begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 1 \end{pmatrix}}_{4\times 3}$$

Figure 2. Over Hiding problem of setting –1 in S

No matter the left-hand or right-hand equation, the support of {1, 2} in D' is 0. That is, item 1 and item 2 never appear together, and they are mutual exclusive! This situation almost never happens in the database which doesn't hide any information. The attackers may interest in this situation and infer that {1, 2} is hidden deliberately. To hide the sensitive patterns, we only need to make their supports smaller than minimum support and need not to decrease their support to 0. To solve the problem, we inject a probability ρ which is called Distortion probability into this approach. Distortion probability is used only when the column j of the sanitization matrix S contains only one "1". (i.e. $S_{jj}=1$), and it works as follows: if $\sum_{k=1}^{m} D_{ik} \cdot S_{kj} \le 0, \forall i, j, 1 \le i \le n, 1 \le j \le m$, D'_{ij} has ρ_j

probability to be set to 1 and $1-\rho_j$ probability to be set to 0.

Lemma 1: Given a minimum support σ, and a level of confidence c. Let $\{i, j\}$ be a pattern in Marked-Set, n_{ij} be the support count of $\{i, j\}$. ρ is the Distortion probability of column j. Without loss of generality, we assume that $S_{ij} = -1$. If ρ satisfies $n_{ij} \times \rho < \sigma \times |D|$ and $\sum_{x=0}^{\lceil \sigma \times |D| \rceil - 1} \binom{n_{ij}}{x} \rho^{x} (1-\rho)^{n_{ij}-x} \geq c$, where $|D|$ is the number of transactions in D, we can say that we are c confident that $\{i, j\}$ isn't frequent in D'.

3.2.2. Conformity probability μ

Setting 1 to enhance the relation of non-sensitive patterns may cause failure in hiding some sensitive patterns. Consider the following example:

$$\underset{D \quad 4\times 3}{\begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 1 \end{pmatrix}} \times \underset{S \quad 3\times 3}{\begin{pmatrix} 1 & -1 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}} = \underset{D' \quad 4\times 3}{\begin{pmatrix} 1 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 1 \end{pmatrix}}$$

Let minimum support = 0.5, {1, 2} be the sensitive pattern and {1, 3}, {2, 3} be the non-sensitive patterns. S_{13}, S_{31}, S_{23} and S_{32} are set to 1 to enhance the relationships of {1, 3} and {2, 3}. As a result, it fails to hide {1, 2}. To avoid this situation, the idea of Conformity probability is proposed. Conformity probability is used when the column j of the sanitization matrix S contains at least two "1"s. It works as follows, if $\sum_{k=1}^{m} D_{ik} \cdot S_{kj} \geq 1$, and at least one entry in column j with value –1 is multiplied by the entry with value 1 in D, $D'ij$ is set to 1 with probability μ_j and 0 with probability 1-μ_j. When all "–1" in column j of S exactly multiply by all entries with value 0 in raw i of D, it means that the row i of D does not contain any entries act on the column j of S, and it only needs to follow the matrix multiplication rules discussed in the previous section.

Lemma 2: Given a minimum support σ , and a level of confidence c. Let $\{i, j\}$ be a pattern in Marked-Set, and $\{k, j\}$ be a pattern in {large2–Marked-Set}, n_{ikj} be the support count of $\{i, k, j\}$. Without loss of generality, we assume that S_{ij}=-1. μ is the Conformity probability of column j. If μ is set according to the following rule,

$$\mu = \begin{cases} 1 & , n_{ijk} < \lceil \sigma \times |D| \rceil \\ n_{ijk} \times \mu < \sigma \times |D| & \textit{and} \ \sum_{y=0}^{\lceil \sigma \times |D| \rceil - 1} \binom{n_{ikj}}{y} \mu^{y} (1-\mu)^{n_{ikj}-y} \geq c \end{cases}$$

we can say that we are c confident that $\{i, j\}$ isn't frequent in D'.

3.2.3. Sanitization algorithm

According to the probability policies discussed above, the sanitization algorithm $D'_{n\times m}$=$D_{n\times m}$×$S_{m\times m}$ is as follows:

```
    Sanitization algorithm
Input: D, S, probabilities for each column of S
Output: D'
for ( i = 1 to n )
{          /*i is a row of D*/
   for ( j = 1 to m )
   {          /*j is a column of S*/
      if ( $D_{ij}$ = 0 )
         $D'_{ij}$ = 0
      else
         { temp = $\sum_{k=1}^{m} D_{ik} \cdot S_{kj}$
      if (column j in S contains some entries= –1, only $S_{jj}$ =1 and temp ≤ 0 )
      $D'_{ij}$ is set to 1 with probability $\rho_j$ and 0 with probability $1-\rho_j$.
      else if (column j in S contains some entries= –1, more than one entry=1, temp ≥ 1 and at least one entry in column j with value –1 is multiplied by the entry with value 1 in D )
      $D'_{ij}$ is set to 1 with probability $\mu_j$ and 0 with probability $1-\mu_j$.
      else if ( temp ≤ 0 )
         $D'_{ij}$ is set to 0.
      else if ( temp ≥ 1)
         $D'_{ij}$ is set to 1.
         }
   }
}
```

Notice that, if the sanitization process causes all the entries in row r of D' equal to 0, randomly choose an entry from some j where D_{rj} =1, and set D'_{rj} =1. This movement is to guarantee that the size of the sanitized database D' is equal to the size of the original database D.

4. Performance evaluation analysis

4.1. Performance quantifying

There are three potential errors in the problem. First of all, some sensitive patterns are hidden unsuccessfully. It means that some sensitive patterns can still be mined from *D*'. Secondly, some non-sensitive patterns cannot be mined from *D*'. And the third, new artificial patterns may be produced after transforming *D* into *D*'.

Besides the three conditions discussed above, we also introduce the other two criteria, dissimilarity and weakness. All of the criteria are discussed as follows:

Criterion 1: some sensitive patterns can still be frequent in *D*'. This condition is denoted as Hiding Failure[13], and measured by $HF = \frac{|P_H(D')|}{|P_H(D)|}$, where $P_H(X)$ represents the number of patterns contained in P_H which is mined from database *X*.

Criterion 2: some non-sensitive patterns are hidden in *D*'. This condition is denoted as Misses Cost[13], and measured by $MC = \frac{|\sim P_H(D)| - |\sim P_H(D')|}{|\sim P_H(D)|}$, where $|\sim P_H(X)|$ represents the number of patterns contained in $\sim P_H$ which is mined from database *X*.

Criterion 3: some artificial patterns are generated after the sanitization process. The error is denoted as Artificial Patterns[13], and it is measured by $AP = \frac{|P(D')| - |P(D) \cap P(D')|}{|P(D')|}$, where $|P(X)|$ represents the number of patterns mined from database *X*. *AP* of our approach and SWA are always 0%. Because both of SWA and our sanitization process are trying to erase the original items in the transactions and are not going to put new items in the transactions.

Criterion 4: the dissimilarity between the original and the sanitized database is also concerned, and it is measured by $Dis = \frac{\sum_{i=1}^{n}\sum_{j=1}^{m}(D_{ij} - D'_{ij})}{\sum_{i=1}^{n}\sum_{j=1}^{m}D_{ij}}$.

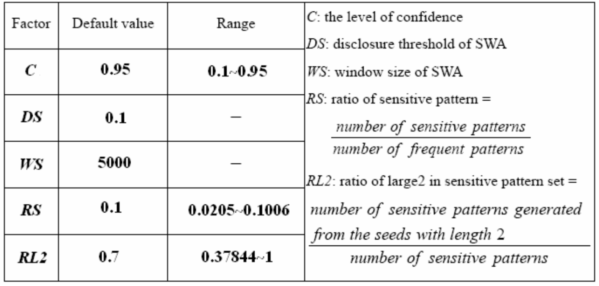
Criterion 5: according to the previous chapter, Forward-Inference Attack is avoided while at least one pair-subpattern of a sensitive pattern is hidden. Forward-Inference Attack is quantified by $Weskness = \frac{|(P_H(D) - P_H(D')) \cap PairS(D')|}{|P_H(D) - P_H(D')|}$, where $PairS(D')$ is the set of sensitive patterns whose pair-subsets can be completely mined from *D*'.

4.2. Experimental results

Two series of experiments are performed in this chapter: the first one is to investigate a balance between the level of confidence and the other criteria in our sanitization process. The second one is to measure the effectiveness of our sanitization process and SWA which has been compare with Algo2a[11] and IGA[13] is so far the algorithm with best performances published and presented in [14] as we know. All the experiments are performed on a PC with Intel Pentium4 2.4GHz, 512M of main memory, under Windows *XP* operating system. To measure the effectiveness of these algorithms and ensure the right of the test dataset, the test dataset is generated by the IBM synthetic data generator. The dataset contains 1000 different items, with 100K transactions where the average length of each transaction is 15 items. The Apriori algorithm with minimum support= 1% is used to mine the dataset and 52964 frequent patterns are gotten.

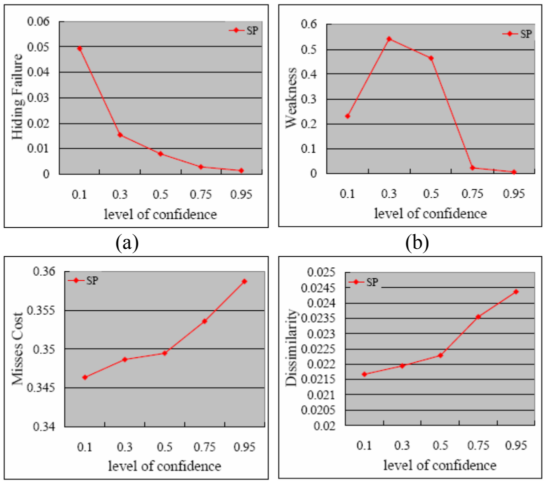
Several sensitive patterns are randomly selected from the frequent patterns with length two to three items to be seeds. With the several seeds, all of their supersets are included into the sensitive patterns set since any pattern which contains sensitive patterns should also be sensitive. All the experimental factors are defined in Table 1.

Table 1. Experiment factor



Factor	Default value	Range	*C*: the level of confidence *DS*: disclosure threshold of SWA *WS*: window size of SWA *RS*: ratio of sensitive pattern = $\frac{\textit{number of sensitive patterns}}{\textit{number of frequent patterns}}$ *RL2*: ratio of large2 in sensitive pattern set = $\frac{\textit{number of sensitive patterns generated from the seeds with length 2}}{\textit{number of sensitive patterns}}$
C	**0.95**	**0.1~0.95**	
DS	**0.1**	–	
WS	**5000**	–	
RS	**0.1**	**0.0205~0.1006**	
RL2	**0.7**	**0.37844~1**	

Fig. 3(a), (b), (c)and (d) show the effect of the level of confidence *C* on hiding failure, weakness, misses cost and dissimilarity for our sanitization process, considering *RS*=0.1. The larger the *C* is, the higher the Distortion and Conformity probabilities are. And as a result, the more entries in *D* are selected to be modified. Therefore, the hiding failure decreases as *C* increases. Moreover, the misses cost and dissimilarity increase as *C* increases. Ideally, the weakness should be getting lower while *C* is getting higher, however, there are some exceptions when *C*=0.1. This is because when the hiding failure problems occur in some patterns which are seeds with length not equal to two, the weakness will not get high even get less.

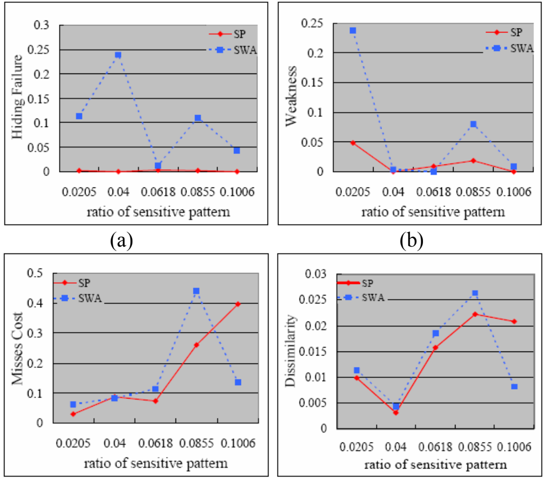


(a) (b) (c) (d)

Figure 3. Relations between *HF*(*Weakness*, *MC*, *Dis*) and *C*

Fig. 4(a), (b), (c) and (d) show the effect of *RS* by comparing our approach to SWA. Refer to Fig. 4(a), because the level of confidence in our sanitization process takes the minimum support into account, no matter how the distribution of the sensitive patterns, we still are *C* confident to avoid the hiding failure problems. However, there is no correlation between the disclosure threshold in SWA and the minimum support. Under the same disclosure threshold, if the frequencies of the sensitive patterns are high, the hiding failure will get high too. In Fig. 4(b), because our approach is to hide the sensitive patterns by decreasing the supports of the pair-subpatterns of the sensitive patterns, the value of weakness is related to the level of confidence. However, according to the disclosure threshold of SWA, when all the pair-subpatterns of the sensitive patterns have large frequencies, it may cause the serious Forward-Inference Attack problems. The hiding failure and the weakness of SWA change with the distributions of the sensitive patterns.

In Fig. 4(c) and Fig. 4(d), ideally, the misses cost and dissimilarity increase as the *RS* increases in our approach and SWA. However, if the sensitive pattern set is composed of too many seeds, a lot of victim pair-subpatterns will be contained in Marked-Set. And as a result, cause a higher misses cost, such as *x*=0.04 in Fig. 4(c). Moreover, refer to the turning points of SWA under *x*=0.0855 to 0.1006 in Fig. 4(c) and Fig. 4(d). The reason of violation is that, the result of the experiment has strong correlation with the distribution of the sensitive patterns. Because the sensitive patterns are chosen randomly, several variant factors of the sensitive patterns are not under control such as the frequencies of the sensitive patterns and the overlap between the sensitive patterns; if the overlap between the sensitive patterns is high, some sensitive patterns can be hidden by removing a common item in SWA. Therefore, decrease the misses cost and the dissimilarity.

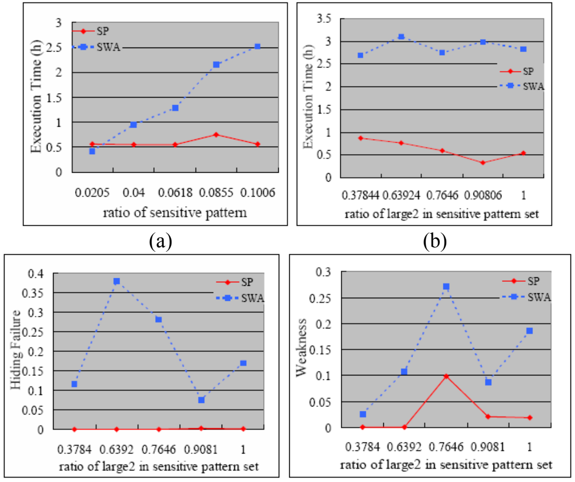


(a) (b) (c) (d)

Figure 4. Relations between *RS* and *HF*(*Weakness*, *MC*, *Dis*)

Fig. 5(a) shows the execution time of SWA and our approach. As shown in the result, the execution time of SWA increases as *RS* increases. On the other hand, our approach can be separated roughly into two parts, one is to get Marked-Set which is strong dependent on the data, the other is to set sanitization matrix and execute multiplication whose execution time is dependent on the numbers of transactions and items in the database. In the experiment, because the items and transactions are fixed, therefore, the execution time of our approach is decided by the setting of Marked-Set which makes the execution time changing slightly.

Fig. 5(b), Fig. 5(c), Fig. 5(d) show the effect of *RL2*. In Fig. 5(b), the execution time of our approach decreases as *RL2* increases. This is because when *RL2* increases, our approach takes less time to generate the Marked-Set ideally.

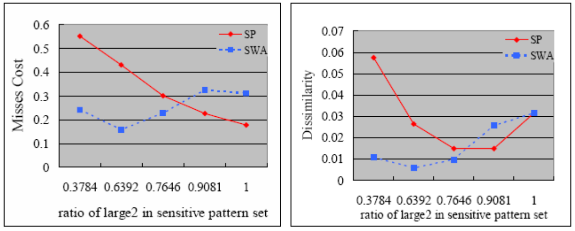


(a) (b)

(c) (d)

Figure 5. (a), (b) isRelations bet. *RS*(*RL2*) and time. (c), (d) is Relations between *RL2* and *HF*(*weakness*)

Refer to Fig. 5(c) and Fig. 5(d), our approach outperforms SWA no matter what the *RL2* is. And our sanitization process approach almost 0% hiding failure. The reason of the hiding failure of SWA is the same in Fig. 4(a), because the frequencies of the sensitive patterns are large. Notice the result at x=0.7646 in Fig. 5(d), because the hiding failure is occurred at the seeds of the sensitive patterns, a high weakness is produced. As shown in Fig. 6(a) and Fig. 6(b), the misses cost and dissimilarity of our approach decreases as *RL2* increases. This is because the larger the *RL2* is, the less the effect on non-sensitive patterns. On the other hand, the weakness and the dissimilarity of SWA are independent of *RL2*.



(a) (b)

Figure 6. (a)Relations between *RL2* and *MC*. (b): Relations *RL2* and *Dis*

5. Conclusion

In this paper, a novel sanitization process which improves the balance between sensitive knowledge protecting and information discovery on frequent patterns has been introduced. By setting the entries of a sanitization matrix to the appropriate values and multiplying the original database by the sanitization matrix with some probability policies, a sanitized database is gotten. Furthermore, it can avoid the Forward-Inference Attack absolutely while the level of confidence which is controlled by the administrator of the database approximates to 1.

The experimental results revealed that although the misses cost and the dissimilarity between the original and sanitized database of our sanitization process are little more than SWA in some conditions, our sanitization process provide more safely protection than SWA. Unlike SWA, our sanitization process could not suffer from Forward-Inference Attack and the probability policies in our approach also take the minimum support into account, the users only need to decide the level of confidence which affects the degree of patterns hiding and don't need to investigate the balance between the disclosure threshold and the minimum support. In the near future, we will investigate how to decrease the misses cost and dissimilarity and apply this method to hiding association rules.

References

[1]R. Agrawal and R. Srikant. Privacy preserving data mining. In ACM SIGMOD Conference on Management of Data, pp: 439–450, Dallas, Texas, May 2000.

[2]A. Evfimievski, J. Gehrke, and R. Srikant. Limiting Privacy Breached in privacy preserving data mining. In Proceedings of the ACM SIGMOD/PODS Conference, 2003.

[3]A. Evfimievski, R. Srikant, R. Agrawal, and J. Gehrke. Privacy preserving mining of association rules. In Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery in Databases and Data Mining, pages 217–228, Edmonton, Alberta, Canada, July 23–26 2002.

[4]S. J. Rizvi and J. R. Haritsa. Maintaining data privacy in association rule mining. In Proceedings of the 28th International Conference on Very Large Data Bases, 2002.

[5]M. Kantarcioglu and C. Clifton. Privacy-preserving distributed mining of association rules on horizontally partitioned data. In ACM SIGMOD Workshop on Research Issues on Data Mining and Knowledge Discovery, June 2002.

[6]Y. Lindell and B. Pinkas. Privacy Preserving Data mining. In CRYPTO, pp: 36-54, 2000.

[7]Benny Pinks. Cryptographic Techniques For Privacy-Preserving Data Mining. ACM SIGKDD Explorations Newsletter, 4(2), Dec 2002

[8]J. Vaidya and C. W. Clifton. Privacy preserving association rule mining in vertically partitioned data. In Proc. of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Edmonton, Canada, July 2002.

[9]S. R. M. Oliveira, O. R. Zaïane and Yücel Saygin. Secure Association Rule Sharing The 8th Pacific-Asia Conference on Knowledge Discovery and Data Mining 2004(PAKDD-04).

[10]Guanling Lee, Chien-Yu Chang and Arbee L.P Chen. Hiding sensitive patterns in association rules mining. The 28th Annual International Computer Software and Applications Conference (COMPSAC 2004)

[11]Verykios, V.S.; Elmagarmid, A.K.; Bertino, E.; Saygin, Y.; Dasseni, E. Association rule hiding. IEEE Transactions On Knowledge And Data Engineering, 16(4), April 2004.

[12]En Tzu Wang, Guanling Lee and Yu Tzu Lin. A Novel Method for Protecting Sensitive Knowledge in Association Rules Mining. To appear at the IEEE, 29th Annual International Computer Software and Applications Conference (COMPSAC 2005)

[13]S. R. M. Oliveira and O. R. Zaïane. Privacy Preserving Frequent Itemset Mining. In Proc. of the IEEE ICDM Workshop on Privacy, Security, and Data Mining Japan, December 2002.

[14] S. R. M. Oliveira and O. R. Zaïane. Protecting Sensitive Knowledge By Data Sanitization. In Proc. of the 3rd IEEE International Conference on Data Mining (ICDM'03).