Protecting Privacy while Discovering and Maintaining Association Rules

Tran Khanh DANG
Faculty of CSE, HCMUT
Ho Chi Minh City, Vietnam
khanh@cs.hcmut.edu.vn

Josef KÜNG
FAW, Johannes Kepler University
Linz, Austria
josef.kueng@faw.jku.at

Huynh V.Q. PHUONG
Faculty of CSE, HCMUT
Ho Chi Minh City, Vietnam
huynhvq.phuong@gmail.com

Abstract—The k-anonymity is an efficient model to preserve data privacy. Of late, this model has been applied to the area of privacy-preserving data mining but the state-of-the-arts are still far from practical needs. In this paper, we propose a new approach that preserves privacy and maintains data utility in data mining. Concretely, we use a k-anonymity model to preserve privacy while discovering and maintaining association rules through a novel algorithm, M3AR-member migration technique for maintaining association rules. We do not use the existing generalization and suppression techniques to achieve a k-anonymity model. Instead, we propose a member migration technique that is more appropriate for the requirements of maintaining association rules. Experimental results establish the practical value and theoretical analyses of our new technique.

Keywords—privacy preservation; k-anonymity; data mining; member migration technique; association rules

I. INTRODUCTION

The advance of information technology has brought many benefits to many organizations such as the ability of storing, sharing, mining data by data mining techniques. However, this bears a big obstacle of leaking out and abusing privacy. So, as a vital need, privacy preservation (PP) was born to undertake great responsibility of preserving privacy and maintaining data quality for data mining techniques. Concurrently, focusing on privacy and data quality is a trade-off in PP-related researches.

A. k-Anonymity Model and Techniques

In order to preserve privacy, identification attributes such as id, name, etc. must be removed. However, this does not ensure privacy since combinations of remaining attributes such as gender, birthday, postcode, etc. can uniquely or nearly identify some individuals. Therefore, sensitive information of the individuals will be exposed. Such remaining attributes are called quasi-identifier attributes [1]. The k-anonymity model [1,2,3] is an approach to protect data from individual identification. The model requires that a tuple in a table, representing an individual with respect to quasi-identifier attributes, has to be identical to at least (k-1) other tuples. The larger the value of k, the better the protection of privacy. To obtain a k-anonymity model, there are various techniques classified into two types: Generalization and Suppression. The Generalization technique builds a hierarchical system for values of an attribute value domain based on the generality of those values and replaces specific values with more general ones. This technique is classified into two generalization levels: attribute and cell levels. The attribute level replaces the current value domain of a quasi-identifier attribute with a more general one. For example, the domain of attribute age is mapped from years to 10-year intervals. The cell level just replaces current values of some essential cells with more general ones. Both levels obtain the k-anonymity model. However, the cell level has an advantage over the attribute level of losing less information as it does not unnecessarily replace many general values, but it has the disadvantage of creating inconsistent values for attributes because the general values co-exist with the original ones. The attribute level has consistency of attribute values but loses much information as there are many changes in original data. It therefore easily falls into a too general state. Generally, the cell level generalization is preferred for its upper hand of less information loss though it is more complicated than the attribute level one. Optimal k-anonymity by the cell level generalization is NP-hard [16]. Many proposed algorithms [5,6,7,10,11,13,14] have used this cell level generalization. The Suppression technique executes suppression on original data table. The Suppression can be applied to a single cell, to a whole tuple or column [4].

B. Problems and Main Contributions

Trade-off between data privacy and data quality is important in PP problem. Traditional approaches have used a variety of metrics as the basis for algorithm operations to minimize data information loss such as Precision [2], WHD, Distortion [5], IL [6], and NCP [7]. Those metrics are too general and do not concentrate on maintaining data quality towards any specific data mining technique. However, in reality, data after being modified will be mined by a specific mining technique. Therefore, modified data may not bring high data quality to any mining technique. In this paper, we propose a new approach to preserve data privacy while concentrating on maintaining data quality towards any specific data mining technique. The output data in our approach will have better data quality (towards data mining techniques that algorithms direct to) than that of the traditional approach. According to this approach, with the same original dataset D, if receiver wants to mine data by a mining technique DM, he will receive a ‘tailored’ dataset D’ modified by a PP algorithm which focuses on maintaining data quality towards DM.

In this paper, we do not aim at embracing all data mining techniques. We just focus on association rule mining techniques and concurrently preserve data privacy by a k-anonymity model. As discussed above, there are various...
techniques to obtain a k-anonymity model. However, we do not use those techniques for some reasons: the attribute level generalization has the disadvantage of creating a lot of data distortion. Many replacements of old values of an attribute with new ones will contribute to direct impact on making wrong in many association rules sets. The cell level generalization has less data distortion but it can increase the number of distinctive values of an attribute and its inconsistency is not a good choice to maintain association rules. The cell Suppression technique is not a desired solution as well since null or unknown values have to be pre-processed before mining. The tuple Suppression has two drawbacks. First, it directly has influence on min_sup of the association rule data mining technique. Second, it can lose many tuples of original data. Also, if the values of an attribute A that needs to be rejected appear in many association rules, then many rules will be loss. The attribute Suppression, therefore, is not suitable either. Moreover, if a receiver does not accept the rejection of attribute A out of the dataset received, it is a good reason not to select this technique.

Hence, to maintain association rules, we will introduce a novel technique, named Member Migration (MM). Basically, the technique first groups tuples in the original dataset D into separate groups based on the similarity of values on a given quasi-identify attribute set and then performs a MM operation between each group pair where there is at least one group having the number of tuples less than k. If a tuple \( t \) in group \( A \) migrates to group \( B \), values of \( t \) have to change to ones of those in group \( B \) with respect to the quasi-identify attribute set. In addition, we propose M3AR algorithm that uses the MM technique to concretize the new approach.

The rest of this paper is organized as follows. Section II discusses MM technique. Section III introduces typical metrics. Section IV presents M3AR algorithm. Section V shows experimental results. Finally, section VI concludes the paper.

II. THE MEMBER MIGRATION TECHNIQUE

Firstly, our technique is to group tuples in the original dataset \( D \) into separate groups based on the identity of values on a quasi-identifier attribute set and then, performs a MM operation between every two-group where there is at least one group having the number of tuples less than \( k \). A group can perform a MM operation with one or more other groups. If a tuple \( t \) in group \( A \) migrates to group \( B \), values of \( t \) have to change to ones of those in group \( B \) with respect to a quasi-identifier attribute set. This technique is illustrated in Fig. 1. Table (a) is the result after grouping tuples in a sample dataset into six separate groups based on the identity of values on a quasi-identifier attribute set \{Att1,Att2,Att3\}. Table (b) obtains a 5-anonymity model with four groups after applying three MM operations between groups. One member (tuple) in group 5 migrated to group 4, values changed from (b,y,β) to (b,x,β). One member in group 5 migrated to group 2, values changed from (b,y,β) to (a,y,β). Two members in group 1 migrated to group 2, values changed from (a,x,α) to (a,y,β).

The MM technique only replaces necessary cell values by other ones in the current value domain, so it inherits advantages of Generalization techniques: less information loss as the cell level Generalization, consistent attribute values as the attribute level Generalization. Besides, it possesses other advantages as follows: no difference between numerical and category attributes; no need to build hierarchies for attribute values based on generality; and finally, as a receiver get a modified dataset \( D' \), he/she will sense that \( D' \) has never been modified.

Figure 1. The migration member technique to obtain a k-anonymity model

Definition 1: A group is a subset of tuples (the number of tuples is greater than zero) of a table that has same values with respect to a given quasi-identifier attribute set.

Definition 2: A group is k-unsafe if has fewer than \( k \) tuples; otherwise, it is k-safe where \( k \) is a given anonymity degree.

Risk: Assume that the desired anonymity degree is \( k \). A group that has the number of tuples \( m \) (\( m > 0 \)) will be estimated risk through the function (cf. Appendix A).

\[
F_{\alpha}(m) = \begin{cases} 
0 & \text{when } m \geq k \\
2k - m & \text{when } 0 < m < k 
\end{cases}
\]

Consider a dataset \( D \), after grouped, resulting in a set of groups \( G = \{g_1, g_2, ..., g_d\} \). Let \( |g| \) denote the number of tuples in the \( i \)th group. Then the total risk \( Risk_D \) of the dataset \( D \) is:

\[
Risk = \sum_i F_{\alpha}(|g_i|) \tag{2}
\]

Observation 1: Assume Risk is a risk of a dataset \( D' \). Risk(\( D' \)) = 0 if and only if \( D' \) has achieved a k-anonymity model.

Proof is simple, using the disproof method.

Definition 3: Let \( g_i \xrightarrow{T} g_j \) be a MM operation from \( g_i \) to \( g_j \) (\( i \neq j \)), if values of all tuples in \( T \) are altered to values of tuples in \( g_j \) considering a quasi-identify attribute set. Then, there exists two separate migrant directions between two groups \( g_i \) and \( g_j \): from \( g_i \) to \( g_j \) and from \( g_j \) to \( g_i \). If \( g_i \xrightarrow{T} g_j \) is “valuable” when the risk of data is decreased after performing that Member Migration operation.

When performing a MM operation \( g_i \xrightarrow{T} g_j \), there are only changes on risk of two groups \( g_i \) and \( g_j \). Therefore, risk reduction of all data is equal to the sum of risk reductions in two group \( g_i \) and \( g_j \) after performing that MM operation.

Theorem 1: If a Member Migration operation \( g_i \xrightarrow{T} g_j \) is “valuable” then the number of k-unsafe groups in set \( \{g_i, g_j\} \) after this operation can not exceed one.

Proof. The case as both two groups \( g_i \) and \( g_j \) are k-safe is obviously true. Let’s consider the cases when there is at least one k-unsafe group, using the disproof method. Assume that after performing a “valuable” MM operation on \( g_i \) and \( g_j \), we still have two k-unsafe groups. We have two cases as follows:

1) When both \( g_i \) and \( g_j \) are k-unsafe, the total risk of two groups before migrating is Risk before\( = 4k - |g_i| - |g_j| \). Assume \( 1 \) is the number of migrant tuples. Without loss of generality, assume the migrant direction is \( g_i \xrightarrow{T} g_j \). Since the number of k-unsafe groups after migrating is two, the risk after migrating is Risk after\( = 4k - (|g_i| - 1) - (|g_j| + 1) = 4k - |g_i| - |g_j| = Risk_{\text{before}} \). However, this is a “valuable” MM operation, so we have a contradiction.
2) One k-safe group and one k-unsafe one. Assume \( g_i \) is a k-unsafe group and \( g_j \) is k-safe. Risk\(_{before}=2k-|g_i| \). Because we have 2 k-unsafe groups after the MM operation, it is obvious that the migrant direction is \( g_i \rightarrow g_j \) with 1 tuples and satisfies two conditions: \( 0<|g_i|+1<k \) and \( 0<|g_j|-1<k \). Hence, \( 0<|g_i|+|g_j|-2k \). Besides, Risk\(_{before}=4k-|g_i|-|g_j|-Risk\(_{before}=2k-|g_i| \), that means \( |g_j|=2k \). From (*) and (**), we have a contradiction.

III. PRELIMINARIES

A. Association Rule and Budget Metric

Given a dataset \( D \), a set of tuples with an attribute set \( l=[A_1, A_2, \ldots, A_n] \). An association rule from \( D: A \rightarrow B \) \((A \cap B=\emptyset) \). Let \( C=A \cup B \) be an itemset. \(\text{Support}(A \rightarrow B)=P(C) \) is the percentage of tuples containing both \( A \) and \( B \) in \( D \). \(\text{Confidence}(A \rightarrow B)=P(B|A)=P(C)/P(A) \) is the percentage of tuples that contain both \( A \) and \( B \) in a tuple set containing \( A \), \(\min sup_s \) and \(\min conf_c \) are minimum support and confidence \([15]\), individually, which are two input thresholds. A rule \( A \rightarrow B \) is valuable when \(\text{Support}(A \rightarrow B)=s \geq s_m \) and \(\text{Confidence}(A \rightarrow B)=c \geq c_m \) are strong rules.

\[ \Delta s=s-s_m \quad (3) \]

For a rule \( A \rightarrow B \) to exist, the number of tuples supporting this rule being changed (on itemset of rule) can not exceed \( \Delta s \). However, we need to consider \(\text{Confidence}(A \rightarrow B) \):

**Case 1:** Reduce a rule confidence as changing values of \( A \) from (3) and (6), to keep rule \( A \rightarrow B \)

\[ c^* = \frac{\text{Support}(A \rightarrow B) - a_1}{\text{Support}(A)} = \frac{s - a_1}{s/c - a_1} \geq c_n \quad (4) \]

From (3) and (4), to keep rule \( A \rightarrow B \) exist, the number of tuples supporting this rule on attributes in \( A \) can not exceed:

\[ \text{MIN} \left[ s - s_m \cdot \frac{(s - a_1)}{s/c - a_1} \right] \quad (5) \]

**Case 2:** Similarly, as changing values of attributes in \( B \), let \( a_2 \), \( c^* \) be the number of tuples supporting this rule, the rule confidence, individually. To keep rule exist, \( c^* \geq c_m \) must hold.

\[ c^* = \frac{\text{Support}(A \rightarrow B) - a_2}{\text{Support}(A)} = \frac{s - a_2}{s/c - a_2} \geq c_n \quad (6) \]

From (3) and (6), to keep rule \( A \rightarrow B \) exist, the number of tuples supporting this rule on attributes in \( B \) can not exceed:

\[ \text{MIN} \left[ s - s_m \cdot \frac{(s - a_2)}{s/c - a_2} \right] \quad (7) \]

Each rule \( r=\rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \righta...
not impact or strengthen $r$. To simplify, we can ignore the impact on $R_{g_j}$. For each $t \in T$, $C_i = |R_{g_i} \rightarrow g_j|$ and the total cost of an operation $g_i \rightarrow g_j$ is:

$$\sum_{i \in T} C_i = \sum_{i \in T} |R_{g_i} \rightarrow g_j|$$  \hspace{1cm} (11)

C. Data Quality

For the characteristics of PP while preserving association rule set, we propose three metrics appropriate for association rule: NRP, percentage of number of newly generated rules; LRP, percentage of number of loss rules; DRP, percentage of number of different rules. Let $R$, $R'$ correspondingly be rule sets mined from an association rule data mining technique of data $D$ and $D'$ at min_sup and min_conf.

$$NRP = \frac{|R - R'|}{|R|} \hspace{1cm} LRP = \frac{|R - R'|}{|R'|} \hspace{1cm} DRP = \frac{|R - R'| + |R' - R|}{|R' + R|} \hspace{1cm} (12)$$

D. Policy

This section presents some policies that are basic to operation mechanism of the algorithm M3AR. Give a group $g$, original tuples of $g$ is all tuples that $g$ has when it has never executed a MM operation with any other groups. Because a group can receive from or give to other groups some tuples, let $\text{origin}(g)$ be all remaining original tuples of $g$ and $\text{foreign}(g)$ be all tuples that $g$ receives from other groups.

1) A k-unsafe group once has received tuple(s), it can only continue receiving tuple(s); otherwise, as its tuple(s) migrate to another group, it can only continue giving its tuple(s) to other groups. The policy does not apply to k-safe groups.

2) $g_i \rightarrow g_j$ and $\forall t \rightarrow (\forall r \in R_{g_i} \rightarrow g_j \rightarrow \text{Budget} > 0)$

3) Consider two groups $g_i, g_j$. Since there is at least one k-unsafe group, assume $g_i$ is a k-unsafe group. The number of migrant tuples ($\text{mgrtN}$) is determined as follows:

* Case 1: $g_i$ is a k-unsafe group. If $g_i \rightarrow g_j$, then the number of migrant members is $\text{Min}(|g_i| - k, |g_j|)$. If $g_i \rightarrow g_j$ then the number of migrant members is $\text{Min}(|g_i| - k, |g_j|)$

* Case 2: $g_j$ is a k-safe group. If $g_i \rightarrow g_j$, then $\text{mgrtN} = |g_i|$. If $g_i \rightarrow g_j$ then the $\text{mgrtN} = \text{Min}(k - |g_i|, |g_j| - k, \text{origin}(g_j))$, when $\text{Min}(k - |g_i|, \text{origin}(g_j)) = 0$ then $g_i \rightarrow g_j$ is impossible.

4) Rule to select a more “useful” MM operation according to descending priority order of three factors: less cost, more risk reduction, and fewer numbers of migrant members.

5) If the number of members that can migrate between $g_i$ and $g_j$ is greater than $\text{mgrtN}$ then select $\text{mgrtN}$ members that have the least cost to be migrant candidates.

IV. THE PROPOSED M3AR ALGORITHM

Algorithm M3AR is divided into two processing stages. First is the Initialization stage: partition tuples of original data $D$ into groups, classify those groups into k-safe and k-unsafe groups, create $R_{\text{orig}}$, set from input association rule set $R$ and calculate budget for each rule in $R_{\text{orig}}$. Then complexity of this stage is $O(|R| + |D|)$ as it linearly depends on the size of rule set $R$ and data $D$. Second is the Process stage: in each while loop, if $\text{SelG}$ is null then randomly select a group in $\text{UG}$ to assign to $\text{SelG}$ and find the most “useful” group $g$ in the rest groups that performs a MM operation with $\text{SelG}$ (policy 4 in section III.D). If $g$ can not be found then move $\text{SelG}$ into $\text{UM}$. Otherwise, perform the MM operation between $\text{SelG}$ and $g$, update budget for influenced association rules.

M3AR Algorithm

**Input:** $R$ association rule set mined from $D$ at min_sup, min_conf; $k$

**Output:** $D'$ enhanced k-association rule

**Var:** $\text{Gi}$ mined set of groups; $\text{KACA}$; $\text{BU}$

**Begin**

1. **Initialization**:
   - $b = \text{set of groups obtained from } b$
   - Divide $G$ into k-safe groups set $D_0$ and k-unsafe groups set $D_1$
   - Compute $R_{\text{orig}}$, and calculate budget for all these rules

2. **Process**:
   - while $\exists (\text{null} \in \text{Gi})$ and $\exists (\text{null} \in \text{Gi})$
     - if ($\text{SelG} = \text{null}$)
       - Self-randomly select a group in $D_0$; $\text{null} \rightarrow (\text{null} \in \text{Gi})$
     - Find an $D_0$ and $D_0$ a group $g$ that can migrate member with $\text{SelG}$ so that it is the most “useful”
     - if ($g$ = null)
       - $\text{null} \rightarrow (\text{null} \in \text{Gi})$
     - else
       - $\text{Budget}$
       - if ($\text{null} \rightarrow (\text{null} \in \text{Gi})$
       - else
         - $\text{null} \rightarrow (\text{null} \in \text{Gi})$
         - if ($|g| > 0$)
           - $\text{SelG} = \text{k-safe group in set (SelG, g)}$
           - if there is no k-unsafe group then $\text{SelG} = \text{null}$ and guarantee that if there exists the k-safe group in set (SelG, g) it must be moved into $D_0$
           - return $\text{null}$
         - else
           - end while
     - if ($\text{null} \rightarrow (\text{null} \in \text{Gi})$

**End**

**Dispense**

1. For each member $t$ of other groups that has migrated into $\text{SelG}$
   - return to its initial group $g_i$ such that
   - $\forall r \in R_{g_i} \rightarrow g_j \rightarrow \text{Budget} - \text{Budget} - 1$

**Dispense**

1. For each remaining origin member $t$ of $\text{SelG}$
   - migrate to the most “useful” group $g'$
   - $\forall r \in R_{g_i} \rightarrow g_j \rightarrow \text{Budget} - \text{Budget} - 1$

**Dispense**

1. For each member $t$ of other groups that has migrated into $\text{SelG}$
   - return to its initial group $g_i$ such that
   - $\forall r \in R_{g_i} \rightarrow g_j \rightarrow \text{Budget} - \text{Budget} - 1$

V. EXPERIMENTS

Our experiments use a real world database Adult [11] to verify the performance of the M3AR algorithm in both process time and data quality by comparing with 3 algorithms: KACA [5], OKA [6], Bottom-Up (BU) [7]. All algorithms are
implemented using VB.Net, Windows XP on a core 2-duo CPU 2.0GHz, 1GB of physical memory. Adult database has 6 numerical and 8 categorical attributes. It leaves 30162 records after removing the records with missing values. In our experiments, we retain only 9 attributes {age, gender, marital, country, race, edu, h_p_w, income, workclass}. The first six attributes are considered as quasi-identifying ones. Mining rule set on original data D and modified data D' with min_sup=0.03 and min_conf=0.5. Beside NRP, LRP and DRP metrics, we also use metric CAVG [11]. The quality of k-anonymity is measured by the average size of groups produced, and an objective is to reduce the normalized average group size. The achieved result is the average of 3 times executing the 4 algorithms with each value of k.

Fig. 4 shows the result of the 4 algorithms on LRP metric. M3AR gets very low, not exceed 0.38%, superior to all remaining algorithms. BU gets higher than KACA, say 16.41% vs. 11.75% at k=30. Fig. 5 shows the result of the 4 algorithms on NRP metric. BU gets the lowest in the four, only 6.5% at k=30. M3AR initially gets higher than KACA but as k increases, KACA increases quickly and higher than M3AR at k=10. KACA gets 25.68% and M3AR gets 9.53% at k=30.

M3AR initially gets higher than KACA but as k increases, KACA increases quickly and higher than M3AR at k=10. KACA gets 25.68% and M3AR gets 9.53% at k=30.

Fig. 6 shows the result of the four algorithms on DRP metric. M3AR gets the lowest with 9.91%, KACA gets 37.44% and BU gets more than 2 times higher than M3AR at k=30. With 3 metrics NRP, LRP and DRP, OAK algorithm gets a very high value, the rule set of original data D is greatly destroyed. Consider all 4 metrics, the stability of M3AR is higher than the 3 remaining algorithms. OAK is the most unstable, its executing time (Fig. 7) is the longest (4010s at k=5), but it is quickly reduced as k increases (482s at k=30). While CAVG metric (Fig. 8) of M3AR, KACA, BU reduces, that of OAK increases as k increases.

VI. CONCLUSIONS

In this paper, our main contribution is threefold: (1) A new approach to PP while maintaining data quality towards some specific data mining technique; (2) The MM technique that overcomes crucial drawbacks of the Generalization and suppression techniques in the k-anonymity protection model; (3) The M3AR algorithm that preserves individual re-identification and maintains association rule sets to concretize our newly proposed approach. Beside the k-anonymity model, there are a variety of its variants proposed to preserve data out of individual re-identification [8,9]. Extending the M3AR algorithm to these models is of our great interests in the future. Also, developing the M3AR to address other problems in the area of privacy preserving data mining [17,18] will also be our future research.

REFERENCES


APPENDIX A: RISK FUNCTION $F_R(m)=2k-m$

Choosing $F_R(m)=C-m$ ($C=2k$, $0<m<k$, $C \in N$) is for the satisfaction of theorem 1. In the proof, case 1 does not depend on C. In case 2, g is k-safe and gi is k-safe $\Rightarrow Risk_{safe}=C-|g_i|$. Because we have 2 k-safe groups after the MM operation, the migrant direction is g$\rightarrow$g with l tuples and satisfies: 2|g|+|g|≤C and l|g|≤C⇒$3e|g_i|+|g_i|=C-2k$. Also, Risk_{safe}=2C-|gi|\{gi|Risk_{safe}=C-|g_i|\Rightarrow|g_i|\leq C-2k\}+1. Thus, |g|\leq C+2. From (***) and (***) we have C+2≤C-2C-2≤C-2k-2. Finally, we choose C=2k and $F_R(m)=2k-m$ ($0<m<k$).