M3AR: A Privacy Preserving Algorithm that Maintains Association Rules

VAN QUOC PHUONG HUYNH, TRAN KHANH DANG

ABSTRACT

Privacy preservation has become an important issue in the age of information to prevent the exposition and abuse of personal information. This problem has attracted much research and the k-anonymity protection model 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, firstly, we propose a new approach that preserves privacy and also maintains data utility towards a specific data mining technique. Concretely, we use a k-anonymity model to preserve privacy while discovering and maintaining association rules through a novel algorithm, called 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 novel Member Migration technique that inherits advantages, avoids disadvantages of the existing k-anonymity-based techniques, and that is more appropriate for the requirements of maintaining association rules. Next, we generalize the M3AR algorithm to introduce another new algorithm, named eM2. This algorithm also employs the Member Migration technique to ensure k-anonymity model, avoid information loss as much as possible, but it does not concentrate on any specific data mining techniques. This shows that our newly proposed Member Migration technique is not only appropriate for maintaining association rules but also suitable for a variety of data mining techniques while protecting user privacy. Experimental results with real-world datasets establish the practical value and confirm the theoretical analyses of our novel approach to the problem of privacy-preserving data mining.

Key words: Privacy preservation, k-anonymity, data mining, Member Migration technique, M3AR

1. INTRODUCTION

The vigorous development 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.
1.1. K-Anonymity Model and Techniques

In order to preserve privacy, identification attributes such as \textit{id}, \textit{name}, etc. must be removed. However, this does not ensure privacy since combinations of remaining attributes such as \textit{gender}, \textit{birthday}, \textit{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 \textit{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 the advantage over the attribute level of losing less information as it does 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 origin 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 as shown in [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]. Figure 1 illustrates a k-anonymity example for Student data with attributes Dept, Course, Birth, Sex, PCode regarded as quasi-identifier attributes. Table (a) is the original data. Tables (b) and (c) are 3-anonymity and 2-anonymity versions of table (a), where anonymization is achieved using the generalization at the cell level and attribute level, respectively.
1.2. Existing 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], NCP [7], CAVG [11]. These 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. So with these traditional approaches, modified data may not bring high data quality to any mining technique. Therefore, in this paper, we will focus on following subjects:

First, we propose a new approach to preserve data privacy while concentrating on maintaining data quality towards a specific data mining technique. Output data of algorithms following this new approach will have better data quality (towards data mining techniques that algorithms direct to) than that of algorithms following the traditional approaches. According to this new approach, with the same origin dataset \( D \), if receiver wants to mine data by a mining technique \( DM \), he/she will receive a ‘tailored’ dataset \( D' \) modified by a PP algorithm, which focuses on maintaining data quality towards \( DM \). Herein, 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.

Second, to maintain association rules, we will introduce a totally new technique called Member Migration (MM). Basically, the technique will choose each two groups of tuples and move some tuples from one group to the other to achieve k-anonymity model. As discussed in section 1.1, there are various techniques to obtain a k-anonymity model. However, we do not use

![Figure 1 k-anonymity model on Student data](image)
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 rule 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 origin data. The attribute Suppression is not suitable either since if the values of attribute A that needs rejecting appear in many association rules, many rules will be lost. Moreover, if a receiver does not accept the rejection of attribute A out of the dataset that he received, this is also a good reason not to select this technique.

Third, two novel algorithms, M3AR and eM^2, will be proposed. Both of them employ the MM technique in order to let original data archive k-anonymity model. However, our two algorithms completely aim at different targets. While M3AR tries to maintain Association Rules set of datasets as much as possible, eM^2 tends to the object as that of the traditional approaches, which make the difference between dataset D and its modified version D' be as little as possible.

The rest of this paper is organized as follows. Section 2 discusses the Member Migration technique. Sections 3, 4 introduce M3AR and eM^2 algorithms respectively. Section 5 shows our experimental results. Finally, section 6 concludes the paper.

2. THE MEMBER MIGRATION TECHNIQUE

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

**Definition 2:** A group is k-unsafe if its number of tuples is fewer than k; otherwise, it is k-safe where k is a given anonymity degree.

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 tuples in group B considering on a given quasi-identifier attribute set. This technique is illustrated in Figure 2. 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 is easily to see that this technique inherits advantages of Generalization techniques: less information loss as the cell level Generalization, consistent attribute values as the attribute level Generalization. Besides, it has its own advantages as follows: no difference between numerical and category attribute; no need to build hierarchies for attribute values based on
generality; and finally, when a receiver gets a dataset $D'$ modified by this technique, he/she will sense that $D'$ has never been modified.

<table>
<thead>
<tr>
<th>ID</th>
<th>Att1</th>
<th>Att2</th>
<th>Att3</th>
<th>No</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>α</td>
<td>x</td>
<td>α</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>α</td>
<td>y</td>
<td>β</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>β</td>
<td>x</td>
<td>γ</td>
<td>7</td>
</tr>
<tr>
<td>4</td>
<td>α</td>
<td>x</td>
<td>β</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>β</td>
<td>y</td>
<td>β</td>
<td>2</td>
</tr>
<tr>
<td>6</td>
<td>c</td>
<td>x</td>
<td>α</td>
<td>5</td>
</tr>
</tbody>
</table>

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

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).

$$
F_R(m) = \begin{cases} 
0, & \text{when } m \geq k \\
2k - m, & \text{when } 0 < m < k 
\end{cases}
$$

(1)

Note: A group is only existent and meaningful when it has at least one tuple, it means that $m$ is always greater than zero.

Consider data $D$, after grouped, has set of groups $G = \{g_1, g_2, ..., g_n\}$, $|g_i|$ is the number of tuples in the $i^{th}$ group. Then the total risk $Risk_D$ of data $D$ is

$$Risk_D = \sum_{i=1}^{n} F_R(|g_i|)$$

(2)

Observation 1. Assume $Risk_{D'}$ is risk of data $D'$. $Risk_{D'} = 0$ if and only if $D'$ has achieved a $k$-anonymity model.

Proof is simple, using the disproof method.

Definition 3. Let $mgrt(g_i \rightarrow g_j):T$ be a Member Migration operation from $g_i$ to $g_j$ ($i \neq j, T \subseteq g_i$) if values of all tuples in $T$ are changed to values of tuples in $g_j$ considering a given quasi-identifier attribute set.

Let $mgrt(g_i,g_j)$ be possible migrant directions between two groups $g_i$ and $g_j$. Then, there exists two separate migrant directions, those are from $g_i$ to $g_j$ with symbol $mgrt(g_i \rightarrow g_j)$ and from $g_j$ to $g_i$ with symbol $mgrt(g_j \rightarrow g_i)$. So $mgrt(g_i,g_j) = mgrt(g_i \rightarrow g_j) \cup mgrt(g_j \rightarrow g_i)$. And with two groups $g_i$, $g_j$ ($i \neq j$), it is easy to see that there are at least two MM operations that can be performed between them.

Definition 4: A Member Migration operation $mgrt(g_i \rightarrow g_j):T$ is “valuable” when the risk of data is decreased after performing that Member Migration operation.

When performing a MM operation $mgrt(g_i \rightarrow g_j):T$, there are only changes on risks 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 $mgrt(g_i \rightarrow g_j):T$ 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 when two groups $g_i$ and $g_j$ are both k-safe is obviously true. Moreover, this case can never happen. Consider the cases when there is at least one k-unsafe group, using the disproof method. Assume 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 $l$ is the number of migrant tuples. Without loss of generality, assume the migrant direction is $g_i \rightarrow g_j$. Since the number of k-unsafe groups after migrating is two, the risk after migrating is $Risk_{after} = 4k-(|g_i|-l)-(|g_j|+l) = 4k-|g_i|-|g_j| = Risk_{before}$. 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 two k-unsafe groups after the MM operation, it is obvious that the migrant direction is $g_i \rightarrow g_j$ with the number of tuples is $l$ and satisfies two conditions: $0<|g_i|+l<k$ and $0<|g_j|-l<k$. So we have $0<|g_i|+|g_j|<2k$ (*). Besides $Risk_{after} = 4k-|g_i|-|g_j| < Risk_{before} = 2k-|g_i|$ that means $|g_j|>2k$ (**). From (*) and (**), we have a contradiction.

From two cases above, the theorem is proven.

Let the MM technique be open, we only define its bases. The technique do not define how to choose two groups for executing a MM operation, or how to determine the migrant direction, how many tuples are migrated, or which tuples belonging to a group will be chosen for migrating. The technique leaves all these questions to algorithms using it in order to have flexible algorithms and a big number of variants.

3. M3AR ALGORITHM

3.1. Association Rule and Budget Metric

Given a dataset $D$ includes a set of tuples with an attribute set $I = \{i_1, i_2,...,i_m\}$. An association rule generated from $D$ has the form $A \rightarrow B (A \subset I, B \subset I, A \cap B = \Phi)$. Let $C = A \cup B$ be an itemset. $Support(A \rightarrow B)$ is the percentage of tuples that contain both $A$ and $B$ in $D$. $Confidence(A \rightarrow B) = P(C)/P(A)$ is the percentage of tuples that contain both $A$ and $B$ in a tuple set containing $A$. $min_{sup}$ ($s_m$) is a minimum support and $min_{conf}$ ($c_m$) is a minimum confidence [15]. They are two thresholds supported as input. An association rule $A \rightarrow B$ is valuable when $Support(A \rightarrow B) = s \geq s_m$ and $Confidence(A \rightarrow B) = c \geq c_m$ are strong rules.

$$\Delta s = s - s_m \quad (3)$$

For 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 $Confidence(A \rightarrow B)$. Consider two cases as follows

Case 1: Reduce a rule confidence as changing values of attributes in $A$. Let $\alpha_i$ be the number of tuples supporting this rule being changed, then the confidence of rule is $c'$. To keep rule exist, then $c' \geq c_m$. 

140
\[
c' = \frac{\text{Support}(A \rightarrow B) - \alpha_1}{\text{Support}(A) - \alpha_1} = \frac{s - \alpha_1}{s / c - \alpha_1} \geq c_m \Rightarrow \alpha_1 \leq \frac{s(c - c_m)}{c(1 - c_m)} \quad (4)
\]

From (3) and (4), to keep rule \(A \rightarrow B\) exist, the number of tuples supporting this rule being changed on attributes in \(A\) cannot exceed a value indicated by (5).

\[
\text{MIN}\left(s - s_m, \frac{s(c - c_m)}{c(1 - c_m)}\right) \quad (5)
\]

**Case 2:** Reduce a rule confidence as changing values of attributes in \(B\). Let \(\alpha_2\) be the number of tuples supporting this rule being changed, then the rule confidence is \(c'\). To keep rule exist, \(c' \geq c_m\).

\[
c' = \frac{\text{Support}(A \rightarrow B) - \alpha_2}{\text{Support}(A)} = \frac{s - \alpha_2}{s / c} \geq c_m \Rightarrow \alpha_2 \leq \frac{s - c_m}{c} \quad (6)
\]

From (3) and (6), to keep rule \(A \rightarrow B\) exist, the number of tuples supporting this rule being changed on attributes in \(B\) cannot exceed a value indicated by (7).

\[
\text{MIN}\left(s - s_m, \frac{s - c_m}{c}\right) \quad (7)
\]

Each rule \(r = A \rightarrow B\) has budget metric \(\text{budget}_r\), determined from (5) if the corresponding attribute set of \(B\) does not contain a quasi-identify attribute. Otherwise, \(\text{budget}_r\) is determined from (7). When \(\text{budget}_r\) is less than zero (0), the risk of losing \(r\) is high.

### 3.2. Impact of the MM Technique on Association Rules

Let \(D\) be a set of tuples, \(R\) be a set of association rules mined from \(D\), \(QI\) be a set of quasi-identify attributes of \(D\), \(Qi_r\) be a set of quasi-identify attributes corresponding to items of association rule \(r, r \in R\). \(R_{\text{care}} = \{r | r \in R, Qi_r \cap QI \neq \emptyset\}\). Therefore, there are only rules in \(R_{\text{care}}\) impacted during the process of changing data \(D\) to data \(D'\). \(R_i = \{r | r \in R_{\text{care}}, t \in D, \text{sup}(t, r)\}\). Here, \(\text{sup}(t, r)\) means that tuple \(t\) supporting association rule \(r\). In other word, \(t\) contains an itemset of \(r\). \(R_i = \{r | r \in R_{\text{care}}, t \in D, -\text{sup}(t, r)\}\), \(R_{\text{care}} = R \cup \overline{R}\). \(G = \{g | g_i \cap g_j = \Phi, \forall i \neq j\}\) is a set of groups grouped according to the similarity of values of tuples considering on attribute set \(QI\). Consider the MM operation \(\text{mgrt}(g_i, g_j): T, \forall t \in T\) we have to change its values in some attributes in \(QI\). This attribute set is \(Qi_j, Qi_j \subseteq QI\). We have some association rule sets as follows

\[
\begin{align*}
R_{1, g_i \rightarrow g_j} &= \{r | r \in R, t \in T, Qi_j \cap Qi_r \neq \Phi\} \\
R_{1, g_i \rightarrow g_j} &= \{r | r \in R, t \in T, Qi_j \cap Qi_r = \Phi\} \\
\overline{R}_{1, g_i \rightarrow g_j} &= \{r | r \in \overline{R}, t \in T, Qi_j \cap Qi_r \neq \Phi\} \\
\overline{R}_{1, g_i \rightarrow g_j} &= \{r | r \in \overline{R}, t \in T, Qi_j \cap Qi_r = \Phi\}
\end{align*}
\quad (8)
\]

141
∀t ∈ T, t migrating from gi to gj does not influence rules in two rule sets \( \overline{R}_{i,g_i \rightarrow g_j} \) and \( \overline{R}_{i,g_i \rightarrow g_j} \). However, it has an impact on rules in two rule sets \( R_{i,g_i \rightarrow g_j} \) and \( \overline{R}_{i,g_i \rightarrow g_j} \). budget_r of all rules \( r \in R_{i,g_i \rightarrow g_j} \) will be reduced one unit for each \( t \in T \). However, \( \forall r = A \rightarrow B \) in \( \overline{R}_{i,g_i \rightarrow g_j} \), it is not certain that its budget_r will be reduced. There exist those cases as follows:

**Case 1:** just increase the support of \( A \). Obviously, this case will reduce the confidence of \( r \), leading to risk of losing rule. Let \( \alpha \) be the times that the same rule \( r \) falls into this case. To keep rule \( r \) exist, the following inequality must be satisfied:

\[
\frac{s}{s + \alpha} \geq c_m \Leftrightarrow \alpha \leq \frac{s(c - c_m)}{c \cdot c_m}
\]

(9)

**Case 2:** just increase the support of \( B \). This case does not influence \( r \).

**Case 3:** increase the support of an itemset containing both \( A \) and \( B \). This case will concurrently increase the support and confidence of rule \( r \), since

\[
\frac{\text{sup}(A \rightarrow B)}{\text{sup}(A)} \leq \frac{\text{sup}(A \rightarrow B) + 1}{\text{sup}(A) + 1} \Leftrightarrow \frac{\text{sup}(A \rightarrow B)}{\text{sup}(A)} = \text{confidence}(r) \leq 1. \text{ This condition is true.}
\]

**Case 4:** all remaining cases. Those cases do not influence \( r \) as well.

Let \( p \) (0 ≤ \( p \) ≤ 1) be the probability that \( t \) impacts on rules \( r \in \overline{R}_{i,g_i \rightarrow g_j} \), falling into case 1. Then, \( t \) reduces \( p \) in budget_r for each \( r \in \overline{R}_{i,g_i \rightarrow g_j} \). Let \( C_t \) be the cost of migrating each \( t \in T \), then \( C_t = |R_{i,g_i \rightarrow g_j}| + p|\overline{R}_{i,g_i \rightarrow g_j}| \). So, the total cost of the operation \( \text{mgrt}(g_i \rightarrow g_j) : T \) is

\[
\sum_{t \in T} C_t = \sum_{t \in T} |R_{i,g_i \rightarrow g_j}| + p|\overline{R}_{i,g_i \rightarrow g_j}| = \sum_{t \in T} |\overline{R}_{i,g_i \rightarrow g_j}| + p \sum_{t \in T} |\overline{R}_{i,g_i \rightarrow g_j}|
\]

(10)

Since a k-anonymity model only considers attributes \( QI \), determining exactly the value of \( p \) is difficult. However, there has a solution to determine exactly \( C_t \) for each \( t \in T \), but it requires us to consider all attributes of \( t \). This will increase algorithm complexity. We see that, for each \( r \in \overline{R}_{i,g_i \rightarrow g_j} \), case 1 weakens \( r \) while other cases do not impact or strengthen \( r \). So, to simplify, we can ignore the impact on \( \overline{R}_{i,g_i \rightarrow g_j} \) of each \( t \in T \). So, the cost for each \( t \in T \) is \( C_t = |R_{i,g_i \rightarrow g_j}| \) and the total cost of the MM operation \( \text{mgrt}(g_i \rightarrow g_j) : T \) is

\[
\sum_{t \in T} C_t = \sum_{t \in T} |R_{i,g_i \rightarrow g_j}|
\]

(11)

### 3.3. 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 \( \text{min}_\text{sup} \) and \( \text{min}_\text{conf} \).
\[ NRP = \frac{|R' - R|}{|R|}, \quad LRP = \frac{|R - R|}{|R|}, \quad DRP = \frac{|R - R| + |R' - R|}{|R|} \]  

(12)

### 3.4. Policy

This section presents some policies that are basic to operation mechanism of the algorithm M3AR. Given 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 origin(\( g \)) be all remaining original tuples of \( g \) and 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, when 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. \( mgrt(g_i \rightarrow g_j) : T \) must satisfy the constraint \( \forall t \in T \rightarrow \left( \forall r \in R, g_i \rightarrow g_j \rightarrow Budget_r > 0 \right) \).

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 (\( mgrtN \)) is determined as follows:

   **Case 1:** \( g_j \) 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 \( g_i \leftarrow g_j \) then the number of migrant members is \( \text{Min}(|g_j|, k - |g_i|) \).

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

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

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

### 3.5. 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 \( R_{\text{care}} \) set from input association rule set \( R \) and calculate budget for each rule in \( R_{\text{care}} \) set. Time 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 SelG is null then randomly select a group in \( UG \) to assign to SelG and find the most “useful” group \( g \) in the rest groups that performs a MM operation with SelG (4th policy in section 3.4). If \( g \) can not be found then move SelG into UM. Otherwise, perform the MM operation between SelG and \( g \), update budget for influenced association rules. Note that there is at most one k-unsafe group in set \{SelG, \( g \)\} after the MM operation (theorem 1), the k-unsafe group will be assigned to SelG so that it is processed in the next loop. Otherwise if there exists no k-unsafe group in \{SelG, \( g \)\} then \( SelG = \text{null} \) so that a new k-unsafe group in \( UG \) is selected randomly and processed in next loop. When the while loop ends, groups in UM will be dispersed by Disperse function. Time complexity of this stage is mainly in while loop because processing groups in UM is much fewer than in while
loop, normally $|UM| = 0$. Easily conclude that time complexity of this stage is $O(|UG| \cdot |G|) \approx O(|G|^2)$. 

**M3AR Algorithm**

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

**Output:** $D'$ achieved $k$-anonymity model.

**Var:** $G, SG, UG$ set of groups; $R_{core}$; $SelG=null$; $UM=\emptyset$ set of groups that cannot migrate member with other groups.

**Begin**

**Initialization:**
1. $G =$ set of groups obtained from $D$;
2. Divide $G$ into $k$-safe groups set $SG$ and $k$-unsafe groups set $UG$;
3. Construct $R_{core}$ and calculate budget for all these rules;

**Process:**
1. while$(((|UG|>0)) \text{ or } |SelG|=null)$
2. if$(SelG==null)$
3. $SelG=$randomly select a group in $UG; UG=|UG\backslash SelG|$
4. $G$ find in $UG$ and $SG$ a group $g$ that can migrate member with $SelG$ so that it is the most "useful";
5. if$(g==null)$
6. $UM=|UM\cup SelG|; SelG=null$
7. else$
8. if$(SelG\rightarrow g)$
9. Migrate $t \in T$ to $g; \forall t \in T \rightarrow (\exists r \in R_{core} \rightarrow \text{Budget}_x = \text{Budget}_x - 1)$

10. else$
11. Migrate $t \in T$ to $SelG; \forall t \in T \rightarrow (\exists r \in R_{core} \rightarrow \text{Budget}_x = \text{Budget}_x - 1)$

12. if$(g \in UG)$
13. $SelG =$ the $k$-unsafe group in set $(SelG, g)$, the case of having no $k$-unsafe group then $SelG = null$. And guarantee that if there exists the $k$-safe group in set $(SelG, g)$, it must be moved into $SG$

"end while"/
14. if$(|UM|>0)$ for each $g$ in $UM \backslash (UM \cap SG)$: $\text{Disperse}(g)\}"

**Figure 3.** M3AR algorithm

Line 6 of *Disperse* function with the rule to select the most "useful" group does not use 4th policy, it uses the following rule: Select a more useful group according to descending priority order of two factors: fewer in number of rule $r$ with $budget_r < 0$ (if $t$ migrates to $g$) and less cost.
This section will present essential bases of eM^2 and the eM^2 algorithm itself in order to show that MM technique is completely suitable to the general approach.

4.1. Metrics

In this subsection, we present three metrics Distortion, IL and Uncertainty used respectively in three algorithms, which are typical for the general approach: KACA [5], OKA [6] and Bottom-Up [7]. Because modified data of MM technique is different to that of Generalization techniques, the formulas of this metrics are adapted to MM technique that will be proposed in the subsection. Besides this three metrics, we also use the CAVG metric employed in [5,7,10,11].

Distortion metric

All allowable values of an attribute form a hierarchical value tree. Each value is represented as a node in the tree, and a node has a number of child nodes corresponding to its more specific values. Let \( t_1 \) and \( t_2 \) be two tuples. \( t_{1,2} \) is the closest common generalization of \( t_1 \) and \( t_2 \) for all \( i \). The value of the closest common generalization \( t_{1,2} \) is calculated as follows

\[
\begin{align*}
v_{i,2} = \begin{cases} v_i \text{ if } v_i = v_{i}^1, v_{i}^2 \text{ and } v_{i,2}^1 \text{ are the values of } \\
\text{the } i\text{-th attribute in } t_1, t_1 \text{ and } t_{1,2} \end{cases} 
\end{align*}
\]

Let \( h \) be the height of a domain hierarchy, and let levels 1, 2, ..., \( h-1 \), \( h \) be the domain levels from the most general to most specific, respectively. Let the weight between domain level \( i \)
and \( i - 1 \) be predefined, denoted by \( w_{i,i-1} \), where \( 2 \leq i \leq h \). When a cell is generalized from level \( p \) to level \( q \), where \( p > q \), the weighted hierarchical distance of this generalization is defined as

\[
WHD(p, q) = \frac{\sum_{i=q+1}^{p} w_{i,i-1}}{\sum_{i=2}^{h} w_{i,i-1}}
\]

where \( w_{i,i-1} = 1/(i-1)^\beta \)

with \( 2 \leq i \leq h, \beta \) is a real number \( \geq 1 \)

Let \( t = \{v_1, v_2, ..., v_m\} \) be a tuple and \( t' = \{v'_1, v'_2, ..., v'_m\} \) be a generalized tuple of \( t \) where \( m \) is the number of attributes in the quasi-identifier. Let \( \text{level}(v_j) \) be the domain level of \( v_j \) in an attribute hierarchy. The Distortion (Dstr) of this generalization is defined as

\[
Dstr(t, t') = \sum_{j=1}^{m} WHD(\text{level}(v_j), \text{level}(v'_j))
\]

Let \( D' \) be generalized from table \( D \), \( t_i \) be the \( i \)-th tuple in \( D \) and \( t'_i \) be the \( i \)-th tuple in \( D' \). The Distortion of this generalization is defined as

\[
Dstr(D, D') = \sum_{i=1}^{|D|} Dstr(t_i, t'_i)
\]

However, if \( D' \) is modified by MM technique then every tuple \( t'_i \) in \( D' \) will be an identical tuple or a non-generalized tuple of \( t_i \) in \( D \). Therefore, if using (16) then \( Dstr(D, D') = 0 \), this is not right. The right formula is defined as

\[
Dstr(D, D') = \sum_{i=1}^{|D|} (Dstr(t_i, t'_i) + Dstr(t'_i, t^*_i))
\]

where \( t^*_i \) is the closest common generalization of \( t_i, t'_i \)

Let \( g_1 \) be a group containing \( |g_1| \) identical tuples \( t_i \) and \( g_2 \) be a group containing \( |g_2| \) identical tuples \( t_2. t_{1,2} \) is the closest common generalization of \( t_1 \) and \( t_2 \). The distance between two groups in KACA is defined as

\[
Dstr(g_1, g_2) = |g_1| \times Dstr(t_1, t_{1,2}) + |g_2| \times Dstr(t_2, t_{1,2})
\]

However, with the MM technique, (18) is no longer right because there is only some tuples in \( g_1 \) or \( g_2 \) modified with respect to given quasi-identifier attribute set. The right formula for the MM technique is defined as

\[
Dstr(\text{mgrt}(g_1 \rightarrow g_2): T) = |T| \times (Dstr(t_1, t_{1,2}) + Dstr(t_2, t_{1,2}))
\]

\[
Dstr(\text{mgrt}(g_2 \rightarrow g_1): T') = |T'| \times (Dstr(t_1, t_{1,2}) + Dstr(t_2, t_{1,2}))
\]

Uncertainty metric

Let \( (A_1, ..., A_n) \) be quasi-identifier attributes, give a numerical attribute \( A_i \). Suppose a tuple \( t = (..., x_i, ...) \) is generalized to tuple \( t' = (..., [y_i, z_i], ...) \) such that \( y_i \leq x_i \leq z_i \) \((I \leq i \leq n)\). On attribute \( A_i \), the normalized certainty penalty is defined as
Give a categorical attribute $A_i$. Let $v_1, \ldots, v_n$ be a set of leaf nodes in a hierarchy tree of $A_i$. Let $u$ be the node in the hierarchy tree such that $u$ is an ancestor of $v_1, \ldots, v_n$ and $u$ does not have any descendant that is still an ancestor of $v_1, \ldots, v_n$. $u$ is called closest common ancestor of $v_1, \ldots, v_n$, denoted by $\text{ancestor}(v_1, \ldots, v_n)$. The number of leaf nodes that are descendants of $u$ is called the size of $u$, denoted by $\text{size}(u)$.

Suppose a tuple $t$ has value $v$ on a categorical attribute $A_i$. When it is generalized in anonymization, the value will be replaced by $\text{ancestor}(v_1, \ldots, v_n)$, where $v_1, \ldots, v_n$ are the values of tuples on the attribute in the same generalized group. The normalized certainty penalty of $t$ is defined as

$$NCP_{A_i}(t) = \frac{\text{size}(u)}{|A_i|}$$

where $|A_i|$ is the number of distinct values wrt. attribute $A_i$.

Let $D$ be a table, $D$ consists of both numerical and categorical attributes, $D'$ be a generalized table of $D$. The total weighted normalized certainty penalty of $D'$ is

$$NCP(D') = \sum_{t \in D'} \sum_{i=1}^{n} (w_i \cdot NCP_{A_i}(t'))$$

(22)

Depends on whether $A_i$ is a numerical or categorical attribute, $NCP_{A_i}(t')$ will be computed by (20) or (21); $w_i$ is weight of attribute $A_i$. (22) is suitable for generalization technique. But with the MM technique, (22) is no longer right because $NCP(D')$ will be zero. For proper with the MM technique, $NCP_{A_i}(t')$ in (22) is adapted as

$$NCP(D') = \sum_{t \in D'} \sum_{i=1}^{n} (w_i \cdot NCP_{A_i}(t', t))$$

$$NCP_{A_i}(t', t) = \frac{|A_i| - t'.A_i}{\text{size}(\text{ancestor}(t.A_i, t'.A_i))}$$

(23)

where $t'$ in $D'$ corresponds with $t$ in $D$.

$D'$ is a version of $D$ modified by MM technique

(*) numeric attribute, (**) category attribute

Let $g_1$ be a group containing $|g_1|$ identical tuples $t_1$ and $g_2$ be a group containing $|g_2|$ identical tuples $t_2$. The total normalized certainty penalty of a MM operation is defined as

$$NCP(\text{mgrt}(g_1 \rightarrow g_2); T) = |T| \sum_{i=1}^{n} (w_i \cdot NCP_{A_i}(t_1, t_2))$$

$$NCP(\text{mgrt}(g_1 \leftarrow g_2); T') = |T'| \sum_{i=1}^{n} (w_i \cdot NCP_{A_i}(t_1, t_2))$$

(24)

where $A_i$ is $i^{th}$ attribute.

IL metric
Let $D$ denote a set of records, which is described by $m$ numerical quasi-identifier attributes $N_1, \ldots, N_m$ and $q$ categorical quasi-identifier attributes $C_1, \ldots, C_q$. Let $P = \{P_1, \ldots, P_k\}$ be a partitioning of $D$, namely, $\bigcup_{i=1}^{k} P_i = D$, $P_i \cap P_j = \emptyset$ $(i \neq j)$. Each categorical attribute $C_i$ is associated with a taxonomy tree $T_{C_i}$ that is used to generalize the values of this attribute. With a partition $P \subset P$, let $\bar{N}_i(P), \tilde{N}_i(P), \bar{N}_i(D)$ respectively denote the max, min, and average values of the tuples in $P$ with respect to the numerical attribute $N_i$. Let $\bar{C}_i(P)$ be the set of values of the records in $P$ with respect to the categorical attribute $C_i$. Let $T_{C_i}(P)$ be the maximal subtree of $T_{C_i}$ rooted at the lowest common ancestor of values of $\bar{C}_i(P)$. Then the diversity of $P$, denoted by $\bar{D}(P)$, is defined as

$$\bar{D}(P) = \sum_{i \in [1,m] \cup [1,q]} \frac{\bar{N}_i(P) - \tilde{N}_i(P)}{\bar{N}_i(D) - \tilde{N}_i(D)} + \sum_{i \in [1,q]} \frac{H(T_{C_i}(P))}{H(T_{C_i})}$$

where $H(T)$ is the height of tree $T$

Let $r', r^*$ be two records, then the distance between $r'$ and $r^*$ is defined as the diversity of the set $\{r', r^*\}$, i.e., $\bar{D}(\{r', r^*\})$. To anonymize the records in $P$ means to generalize these records to the same values with respect to each quasi-identifier attribute. The amount of information loss occurred by such a process, denoted as $L(P)$, is defined as

$$L(P) = |P| \cdot \bar{D}(P)$$

where $|P|$ is the number of records in $P$.

Therefore, the total information loss of $D$ is defined as

$$L(D) = \sum_{i \in [1,k]} |P_i| \cdot \bar{D}(P_i)$$

where $|P_i|$ is the number of records in $P_i$.

Let $D'$ be a $k$-anonymity version of $D$ modified by the MM technique. Assume $D'$ has a set of groups $G' = \{g'_1, \ldots, g'_m\}$. If we apply (27) to $D'$, it means that we apply (26) for each $g'$ in $G'$, this is not right because there is only some tuples in $g'$ modified with respect to quasi-identifier attributes, and all remaining tuples in $g'$ do not have any modification. In order to satisfy the MM technique, $L(D')$ is adapted as

$$L(D') = \sum_{i \in [1,k]} \bar{D}(t, t')$$

where $t'$ corresponds with $t$ in $D$.

Let $g_1$ be a group containing $|g_1|$ identical tuples $t_1$ and $g_2$ be a group containing $|g_2|$ identical tuples $t_2$. The total information loss of a MM operation is defined as

$$L(\text{mgrt}(g_1 \rightarrow g_2) : T) = |T| \cdot \bar{D}(t_1, t_2)$$

$$L(\text{mgrt}(g_1 \leftarrow g_2) : T) = |T| \cdot \bar{D}(t_1, t_2)$$

The eM$^2$ algorithm uses all the three metrics: Distortion, Uncertainty and IL. So, in order to be convenient, we use a common symbol $DIF$ to denote all this three metrics. Give two groups $g_1, g_2$, the number of migrant tuples belonging to each migrant direction is determined by 3rd policy. From (19), (24), or (29), we can see that, the chosen migrant direction is a migrant direction with less number of migrant tuples than the others.
CAVG metric

The metric is defined as the following

$$CAVG = \frac{\text{total records}}{\text{total groups}} / k$$  (30)

The quality of k-anonymity is measured by the average size of groups produced, an objective is to reduce the normalized average group size. It is mathematically sound and not intuitive to reflect changes being made to $D$. However, the metric reflects that the more its value approaches to one the more approximation among sizes of groups are.

4.2. Policy

This subsection presents three policies that are basic to operation mechanism of the eM$^2$ algorithm.

1. Same as 1st policy of M3AR algorithm in section 3.4.
2. Given two groups $g_i, g_j$. Assume we have $\text{mgrt}(g_i \rightarrow g_j): T$ then $T \subseteq \text{origin}(g_i)$. Because all tuples in $\text{origin}(g_i)$ are identical, $T$ can be chosen randomly or from the first $|T|$ tuples in $\text{origin}(g_i)$.
3. Same as 3rd policy of M3AR algorithm in section 3.4.

4.3. The Proposed eM2 Algorithm

The eM$^2$ algorithm is divided into two processing stages. First is the Initialization stage: partition tuples of $D$ into groups, classify those groups into k-safe and k-unsafe groups. Time complexity of this stage is $O(|D|)$ as it linearly depends on the size of 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 a group $g$ in the rest groups so that $\text{DIF}(\text{mgrt}(\text{SelG}, g); T)$ is minimized. If a group $g$ cannot be found then the algorithm exits from the while loop. Otherwise, perform the MM operation between $\text{SelG}$ and $g$. Note that there is at most one k-unsafe group in $\{\text{SelG}, g\}$ after the MM operation (cf. Theorem 1), so if there exists the k-unsafe group, it will be assigned to reference $\text{SelG}$ and processed in the next loop, otherwise $\text{SelG} = \text{null}$ so that a new k-unsafe group in $\text{UG}$ is selected randomly and processed in next loop. When the while loop ends, if $\text{SelG}$ is not null, it will be dispersed by Disperse function. Time complexity of Process stage is mainly in while loop because processing time of Disperse function is so much fewer than that of while loop. Moreover, $\text{SelG}$ is almost null after the loop. It is easy to conclude that the time complexity of this stage is $O(|\text{UG}||G|) \approx O(|G|^2)$. So it is also the time complexity of the algorithm.

Observation 2: After the while loop in the eM$^2$ algorithm finishes, if Disperse function is called then the k-unsafe group processed by this function is the final and only k-unsafe one.

The reason for existing a k-unsafe group, which can not perform a MM operation with any other groups is risen from case 2 in the 3rd policy of eM$^2$ (cf. section 4.2).

Proof. Assume $\text{SelG}$ is a k-unsafe group that is processed by Disperse function. It means that $\text{SelG}$ cannot perform a MM operation with any remaining groups. So we have three conditions as follows:
1. \( SelG \) must have received some tuples from other groups. Because if \( SelG \) has never received tuple(s) from other group(s), it can give its tuple(s) to other groups. This means that \( SelG \) can always perform a MM operation with other groups. Now, \( SelG \) can only receive some tuples (cf. 1st policy of eM\(^2\) in 4.2).

2. Every k-safe group \( g \) in set of k-safe groups (SG) must satisfy \( \text{Min}(|g| - k, |\text{origin}(g)|) = 0 \) so that \( SelG \) can not perform a MM operation \( \text{mgrt}(SelG \leftarrow g):T \) with any k-safe group \( g \) in SG.

3. There does not exist any k-unsafe group apart from \( SelG \). Because if there exists a k-unsafe group \( g \) in set of k-unsafe groups (UG) then \( g \) can give some tuples to \( SelG \). It means that, we still find a group \( g \), which can perform a MM operation \( \text{mgrt}(SelG \leftarrow g):T \) with \( SelG \).

Only with condition 3, we can see that \( SelG \) is the final and only k-unsafe group. And line 5 in the eM\(^2\) algorithm says that if group \( g \) is not found then \( SelG \) is the final and only k-unsafe one. Therefore, the while loop will be exited and all tuples in \( SelG \) will be processed by the Disperser function.

```
\begin{figure}[h]
\begin{center}
\begin{minipage}{0.8\textwidth}
\begin{algorithm}
\caption{eM2 algorithm}
\label{alg:eM2}
\textbf{Input:} The original data \( D \); parameter \( k \); 
\textbf{Output:} \( D' \) achieved \( k \)-anonymity model; 
\textbf{Var:} \( G=\emptyset,SG=\emptyset,UG=\emptyset \) are three sets of groups; \( SelG=null \); 
\textbf{Begin}
\begin{itemize}
\item \textbf{Initialization:}
\begin{itemize}
\item 1. \( G \) - set of groups obtained from \( D \); 
\item 2. Divide \( G \) into set of k-safe groups \( SG \) and set of k-unsafe groups \( UG \);
\end{itemize}
\item \textbf{Procedure:}
\begin{itemize}
\item 1. \textbf{while} \((|UG|>0) \text{ or } (|SelG|=null)\) \textbf{do}
\begin{itemize}
\item 2. \textbf{if} \((|SelG|=null)\) \textbf{then}
\begin{itemize}
\item 3. \textbf{SelG} = randomly select a group in \( UG \); \( UG = (UG \setminus SelG) \);
\end{itemize}
\end{itemize}
\item 4. Find in \( UG \), \( SG \) a group \( g \) so that \( \text{Min}(|g| - k, |\text{origin}(g)|) = 0 \) is minimized;
\item 5. \textbf{if} \( g \in UG \) \textbf{then}
\begin{itemize}
\item 6. \textbf{if} \((g \in SG)\) \textbf{then}
\begin{itemize}
\item 7. \( UC = UC \setminus g \);
\item 8. \( \text{mgrt}(SelG \leftarrow g):T \); 
\item 9. \textbf{if} \((|SelG|=0)\) \textbf{then}
\begin{itemize}
\item 10. \( |g| = 0 \) \textbf{then}
\begin{itemize}
\item 11. \( |g| = k \) \textbf{then}
\begin{itemize}
\item 12. \textbf{else} \( SelG = null \); \( SG = SG \cup g \);
\item 13. \textbf{else} \( SG = SG \cup g \);
\item 14. \textbf{else} \( / \ast g \in SG \ast / \)
\begin{itemize}
\item 15. \textbf{if} \((|g|=0)\) \textbf{then}
\begin{itemize}
\item 16. \textbf{if} \((|SelG|=k)\) \textbf{then}
\begin{itemize}
\item 17. \( SC = SC \cup SelG \); \( SelG = null \);
\item 18. \textbf{else} \( SC = SC \cup SelG \); \( SelG = g \);
\item 19. \textbf{else} \( / \ast g \in SC \ast / \)
\begin{itemize}
\item 20. \textbf{if} \((|SelG|=0)\) \textbf{then}
\begin{itemize}
\item 21. \textbf{else} \( / \ast \text{mgrt}(SelG \leftarrow g):T \ast / \)
\begin{itemize}
\item 22. \textbf{if} \((|SelG|=k)\) \textbf{then}
\begin{itemize}
\item 23. \( SC = SC \cup SelG \); \( SelG = null \);
\item 24. \textbf{if} \((|SelG|=null)\) \textbf{Disperser}(SelG, SC);
\item 25. \textbf{End}
\end{itemize}
\end{itemize}
\end{itemize}
\end{itemize}
\end{itemize}
\end{itemize}
\end{itemize}
\end{itemize}
\end{itemize}
\end{itemize}
\end{minipage}
\end{center}
\end{figure}
```

**Figure 5.** eM2 algorithm
This section presents empirical experiments using the real world database Adult [12] to verify the performance of our two algorithms, M3AR and eM², in both processing time and data quality by comparing with the three algorithms: KACA, OKA, Bottom-Up (BU). All algorithms are implemented using VB.Net and executed on a Core (MT) 2 duo CPU 2.0 GHz with 1 GB physical memory, the operating system is MS Windows XP. The Adult database has 6 numerical attributes and 8 categorical attributes. It leaves 30162 records after removing the records with missing values. In our experiments, we retain only nine attributes {age, gender, marital, country, race, edu, h_p_w, income, workclass}. The first six attributes are considered as quasi-identifying attributes. Table I describes the features of these six attributes. Column “Height” is the number of levels that values of an attribute are generalized.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Type</th>
<th># of Values</th>
<th>Height</th>
</tr>
</thead>
<tbody>
<tr>
<td>Age</td>
<td>Numeric</td>
<td>74</td>
<td>4</td>
</tr>
<tr>
<td>Gender</td>
<td>Categorical</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Marital</td>
<td>Categorical</td>
<td>7</td>
<td>3</td>
</tr>
<tr>
<td>Country</td>
<td>Categorical</td>
<td>41</td>
<td>3</td>
</tr>
<tr>
<td>Race</td>
<td>Categorical</td>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td>Edu</td>
<td>Categorical</td>
<td>16</td>
<td>4</td>
</tr>
</tbody>
</table>

5.1. M3AR Experiments

This subsection presents comparisons of M3AR with three algorithms KACA, OKA and BU through our three metrics NRP, LRP and DRP. Association Rule sets are mined on origin data D and modified data D’ of the four algorithms with min_sup = 0.03 and min_conf = 0.5. Here, we use DMX of Analysis Services SQL SERVER 2005 for mining Association Rule. The achieved result is the average of three times executing the four algorithms with each value of k.
Figure 7 shows the result of the four algorithms on LRP metric. M3AR gets very low, not exceed 0.38%, superior to the three remaining algorithms. BU gets higher than KACA. BU gets 16.41% and KACA gets 11.75% at k=30.

Figure 8 shows the result of the four algorithms on NRP metric. BU gets the lowest in four algorithms and gets 6.5% at k=30. M3AR initially gets higher than KACA but when k increases, KACA increases quickly and gets higher than M3AR at k>10. KACA gets 25.68% and M3AR gets 9.53% at k=30.

Figure 9 shows the result of the four algorithms on DRP metric, M3AR gets the lowest, it gets 9.91%, KACA gets 37.44% and BU gets 22.01% at k=30. So in this metric, KACA gets approximately four times and BU gets more than two times higher than M3AR at k=30.

With all three metrics NRP, LRP and DRP, OAK algorithm gets a very high value; the rule set of origin data $D$ is greatly destroyed. Consider all four metrics, the stability of M3AR is higher than the three remaining algorithms. OKA is the most unstable; executing time (Figure 10) of OKA is the longest (4010 seconds at k=5), but then it is quickly reduced when k increases (482 seconds at k=30); while CAVG metric (Figure 11) of three algorithms M3AR, KACA, Bottom-Up reduces, that of OKA increases when k increases.
The subsection shows experiments of eM^2 algorithm. Concretely, eM^2 will be compared with KACA, OKA and Bottom-Up respectively three metrics Distortion, IL and Uncertainty. With Distortion metric, all six quasi-identifier attributes are treated as categorical attribute and WHD in (14) uses \( w_{i,j} = 1/(i-1) \), it means that \( \beta = 1 \). With IL and Uncertainty metrics, age attribute is treated as a numerical attribute and five remaining quasi-identifier attributes are treated as categorical attributes. In (22), (23) and (24) we set \( w_j = 1 \) for all attributes. The achieved result is the average of three times executing the algorithms with each k.

Figure 12 shows comparisons between eM^2 and KACA on Distortion, CAVG metrics and execution time. With Distortion, eM^2 gets approximately with KACA. But CAVG and execution time of eM^2 is better than those of KACA.

Figure 13 shows comparisons between eM^2 and Bottom-Up on Uncertainty, CAVG metrics and execution time. With Uncertainty and execution time, eM^2 gets much better than OKA. And with CAVG, eM^2 gets better than Bottom-Up but there are not much differences.

Figure 14 shows comparisons between eM^2 and OKA on IL, CAVG metrics and execution time. With IL, eM^2 gets much better than OKA. Execution time of OKA is long (4010 seconds at k=5), but then it quickly reduces when k increases (482 seconds at k=30). With CAVG, OKA has a difference with three remaining algorithms; CAVG of eM^2, KACA and Bottom-Up reduces when k increases, but that of OKA increases when k increases.
Figure 13. Compare between eM$^2$ and Bottom-Up

Figure 14. Compare between eM$^2$ and OKA

Execution time of eM$^2$ when using Distortion is higher than that of eM$^2$ when using Uncertainty and IL though with the same eM$^2$ algorithm employed. It means that, computation complexity of Distortion metric is higher than that of Uncertainty and IL metrics.

6. CONCLUSIONS

In this paper, our main contribution is threefold:

1. Proposed a new approach to privacy preservation while maintaining data quality for data mining techniques.

2. Proposed the Member Migration technique that is appropriate for maintaining Association Rules set. Besides, it also includes the advantages of the Generalization techniques in the k-anonymity model and has its own unique characteristics for ensuring k-anonymity and mitigating information loss in general-purpose datasets.

3. Proposed (i) M3AR algorithm that preserves individual re-identification and maintains association rule sets to concretize our newly proposed approach; and (ii) eM$^2$ algorithm based on the Member Migration technique that has advantages of data quality and execution time over existing state-of-the-art techniques while obtaining k-anonymity. By proposing adapted formulas of metrics (cf. subsection 4.1) and carrying out intensive experiments, we have shown that the Member Migration technique and eM$^2$ algorithm is completely suitable for traditional approach into privacy preservation.
Beside the k-Anonymity model, there are a variety of its variants proposed to preserve data out of individual re-identification such as: l-Diversity [17], t-Closeness [8], (α, k)-Anonymity [9]. Extending M3AR and eM² to these models is of our great interests in the future. Furthermore, developing or varying the algorithms to deal with other problems in the area of privacy protection in data mining will also be among our future intensive research activities.

ACKNOWLEDGEMENT

The work was contributed by helpful opinions of Dr. Vo Thi Ngoc Chau (IITB, India); Dr. Van-Hoai Tran (CSE/HCMUT), Dr. Nguyen Duc Cuong (IU/VNUHCM), and the reviewers. Also, we would like to thank all people in ASIS Lab that help us accomplish this work.

REFERENCES

5. J.Y. Li, R. C.W. Wong, A.W.C. Fu et al. - Anonymisation by Local Recoding in Data with Attribute Hierarchical Taxonomies, IEEE Transactions on Knowledge and Data Engineering, 20(9), 1187-1194, 2008.
APPENDIX A: RISK FUNCTION $F_R(M) = 2^k - m$

The reason for choosing $F_R(m) = C - m$ ($C = 2k$, $0 < m < k$, $C \in \mathbb{N}$) is for the satisfaction of theorem 1. In the proof of theorem 1, we can see that case 1 does not depend on $C$. So we only consider case 2 ($g_i$ is a $k$-unsafe group and $g_j$ is $k$-safe). We have $Risk_{before} = C - |g_i|$. Because we have two $k$-unsafe groups after the MM operation, it is obvious that the migrant direction is $g_i \leftarrow g_j$ with the number of tuples is $l$ and satisfies two conditions: $2 \leq |g_i| + l \leq k - 1$ and $1 \leq |g_j| - l \leq k - 1$. So we have $3 \leq |g_i| + |g_j| \leq 2k - 2$ (*). Besides $Risk_{after} = 2C - |g_i| - |g_j| < Risk_{before} = C - |g_i|$ that means $|g_i| < C = |g_i| + 1$. So we have $|g_i| + |g_j| \geq C + 2$ (**). From (*) and (**) we have $C + 2 \leq 2k - 2 = C \leq 2k - 4$ (***). If (***') is true, it means that theorem 1 is false. Therefore, for theorem 1 is true, (**') must be false. In other words, $C$ must satisfy $C \geq 2k - 3$. Finally, we choose $C = 2k$ and $F_R(m) = 2k - m$ ($0 < m < k$).

Address: Received Dec 12, 2009

Tran Khanh DANG, Van Quoc Phuong HUYNH
Faculty of Computer Science & Engineering, HCMUT
VNU-HCM, Ho Chi Minh City, Vietnam
E-mail: khanh@cse.hcmut.edu.vn, phuong151@yahoo.com