ICST Transactions on Scalable Information Systems Research Article ICST.ORG

# Privacy Preserving Large-Scale Rating Data Publishing

Xiaoxun Sun[1], Lili Sun[2,*]

[1]Australian Council for Educational Research, Australia
[2]Department of Mathematics & Computing, University of Southern Queensland, Australia

## Abstract

Large scale rating data usually contains both ratings of sensitive and non-sensitive issues, and the ratings of sensitive issues belong to personal privacy. Even when survey participants do not reveal any of their ratings, their survey records are potentially identifiable by using information from other public sources.

In order to protect the privacy in the large-scale rating data, it is important to propose new privacy principles which consider the properties of the rating data. Moreover, given the privacy principle, how to efficiently determine whether the rating data satisfied the required privacy principle is crucial as well. Furthermore, if the privacy principle is not satisfied, an efficient method is needed to securely publish the large-scale rating data. In this paper, all these problem will be addressed.

**Keywords:** Privacy preserving, anonymity

Received on 30 December 2011; accepted on 4 January 2012; published on 04 February 2013



doi:10.4108/trans.sis.2013.01-03.e3

## 1. Introduction

The problem of privacy-preserving data publishing has received a lot of attention in recent years. Privacy preservation on relational data has been studied extensively. A major category of privacy attacks on relational data is to re-identify individuals by joining a published table containing sensitive information with some external tables. Most of existing work can be formulated in the following context: several organizations, such as hospitals, publish detailed data (called microdata) about individuals (e.g. medical records) for research or statistical purposes [22, 23, 28, 32].

Privacy risks of publishing microdata are well-known. Famous attacks include de-anonymisation of the Massachusetts hospital discharge database by joining it with a public voter database [32] and privacy breaches caused by AOL search data [16]. Even if identifiers such as names and social security numbers have been removed, the adversary can use linking [32], homogeneity and background attacks [23] to re-identify individual data records or sensitive information of individuals. To overcome the re-identification attacks, $k$-anonymity was proposed [25–27, 32]. Specifically, a data set is said to be $k$-anonymous ($k \geq 1$) if, on the quasi-identifier (QID) attributes (that is, the maximal set of join attributes to re-identify individual records), each record is identical with at least $k-1$ other records. The larger the value of $k$, the better the privacy is protected. Several algorithms are proposed to enforce this principle [1, 7, 12, 18–21]. Machanavajjhala et al. [23] showed that a $k$-anonymous table may lack of diversity in the sensitive attributes.

To overcome this weakness, they propose the $l$-diversity [23]. However, even $l$-diversity is insufficient to prevent attribute disclosure due to the skewness and the similarity attack. To amend this problem, $t$-closeness [22] was proposed to solve the attribute disclosure vulnerabilities inherent to previous models.

Recently, a new privacy concern has emerged in privacy preservation research: how to protect individuals' privacy in large survey rating data. Though several models and many algorithms have been proposed to preserve privacy in relational data (e.g., $k$-anonymity [32], $l$-diversity [23], $t$-closeness [22], etc.),

*Corresponding author. Email: xiaoxun.sun@gmail.com

| | non-sensitive | | | sensitive |
|---|---|---|---|---|
| ID | issue 1 | issue 2 | issue 3 | issue 4 |
| $t_1$ | 6 | 1 | *null* | 6 |
| $t_2$ | 1 | 6 | *null* | 1 |
| $t_3$ | 2 | 5 | *null* | 1 |
| $t_4$ | 1 | *null* | 5 | 1 |
| $t_5$ | 2 | *null* | 6 | 5 |

(a)

| | non-sensitive issues | | |
|---|---|---|---|
| name | issue 1 | issue 2 | issue 3 |
| Alice | excellent | so bad | - |
| Bob | awful | top | - |
| Jack | bad | - | good |

(b)

**Table 1.** (a) A published survey rating data set containing ratings of survey participants on both sensitive and non-sensitive issues. (b) Public comments on some non-sensitive issues of some participants of the survey. By matching the ratings on non-sensitive issues with public available preferences, $t_1$ is linked to Alice, and her sensitive rating is revealed.

most of the existing studies are incapable of handling rating data, since the survey rating data normally does not have a fixed set of personal identifiable attributes as relational data, and it is characterized by high dimensionality and sparseness. The survey rating data shares the similar format with transactional data. The privacy preserving research of transactional data has recently been acknowledged as an important problem in the data mining literature [14, 37].

## 2. Motivation

On October 2, 2006, Netflix, the world's largest online DVD rental service, announced the $1-million Netflix Prize to improve their movie recommendation service [15]. To aid contestants, Netflix publicly released a data set containing 100,480,507 movie ratings, created by 480,189 Netflix subscribers between December 1999 and December 2005. Narayanan and Shmatikov shown in their recent work [24] that an attacker only needs a little information to identify the anonymized movie rating transaction of the individual. They re-identified Netflix movie ratings using the Internet Movie Database (IMDb) as a source of auxiliary information and successfully identified the Netflix records of known users, uncovering their political preferences and other potentially sensitive information.

We consider the privacy risk in publishing anonymous survey rating data. For example, in a life style survey, ratings to some issues are non-sensitive, such as the likeness of book "Harry Potter", movie "Star Wars" and food "Sushi". Ratings to some issues are sensitive, such as the income level and sexuality frequency. Assume that each survey participant is cautious about his/her privacy and does not reveal his/her ratings. However, it is easy to find his/her preferences on non-sensitive issues from publicly available information sources, such as personal weblog or social networks. An attacker can use these preferences to re-identify an individual in the anonymous published survey rating data and consequently find sensitive ratings of a victim.

Based on the public preferences, person's ratings on sensitive issues may be revealed in a supposedly anonymized survey rating data set. An example is given in the Table 1. In a social network, people make comments on various issues, which are not considered sensitive. Some comments can be summarized as in Table 1(b). People rate many issues in a survey. Some issues are non-sensitive while some are sensitive. We assume that people are aware of their privacy and do not reveal their ratings, either non-sensitive or sensitive ones. However, individuals in the anonymoized survey rating data are potentially identifiable based on their public comments from other sources. For example, Alice is at risk of being identified, since the attacker knows Alice's preference on issue 1 is 'excellent', by cross-checking Table 1(a) and (b), s/he will deduce that $t_1$ in Table 1(a) is linked to Alice, the sensitive rating on issue 4 of Alice will be disclosed. This example motivates us the following research questions:

*(Satisfaction Problem): Given a large scale rating data set $T$ with the privacy requirements, how to efficiently determine whether $T$ satisfies the given privacy requirements?*

Although the satisfaction problem is easy and straightforward to be determined in the relational databases, it is nontrivial in the large scale rating data set. The research of the privacy protection initiated in the relational databases, in which several state-of-art privacy paradigms [22, 23, 32] are proposed and many greedy or heuristic algorithms [12, 19, 20, 28] are developed to enforce the privacy principles. In the relational database, taking $k$-anonymity as an example [26, 32], it requires each record be identical with at least $k-1$ others with respect to a set of quasi-identifier attributes. Given an integer $k$ and a relational data set $T$, it is easy to determine if $T$ satisfies $k$-anonymity requirement since the equality has the transitive property, whenever a transaction $a$ is identical with $b$, and $b$ is in turn indistinguishable with $c$, then $a$ is the same as $c$. With this property, each transaction in $T$ only needs to be check once and

the time complexity is at most $O(n^2d)$, where $n$ is the number of transactions in $T$ and $d$ is the size of the quasi-identifier attributes. So the satisfaction problem is trivial in relational data sets. While, the situation is different for the large rating data. First of all, the survey rating data normally does not have a fixed set of personal identifiable attributes as relational data. In addition, the survey rating data is characterized by high dimensionality and sparseness. The lack of a clear set of personal identifiable attributes together with its high dimensionality and sparseness make the determination of satisfaction problem challenging. Second, the defined dissimilarity distance between two transactions ($\epsilon$-proximate) does not possess the transitive property. When a transaction $a$ is $\epsilon$-proximate with $b$, and $b$ is $\epsilon$-proximate with $c$, then usually $a$ is not $\epsilon$-proximate with $c$. Each transaction in $T$ has to be checked for as many as $n$ times in the extreme case, which makes it highly inefficient to determine the satisfaction problem. It calls for smarter technique to efficiently determine the satisfaction problem before anonymizaing the survey rating data. To our best knowledge, this research is the first touch of the satisfaction of privacy requirements in the survey rating data.

*How to preserve individual's privacy in the large scale rating data set?*

Though several models and algorithms have been proposed to preserve privacy in relational data, most of the existing studies can deal with relational data only [22, 23, 31**?** ]. Divide-and-conquer methods are applied to anonymize relational data sets due to the fact that tuples in a relational data set are separable during anonymisation. In other words, anonymizing a group of tuples does not affect other tuples in the data set. However, anonymizing a survey rating data set is much more difficult since changing one record may cause a domino effect on the neighborhoods of other records, as well as affecting the properties of the whole data set. Hence, previous methods can not be applied to deal with survey rating data and it is much more challenging to devise anonymisation methods for large scale rating data than for relational data.

## 3. Related work

Privacy preserving data publishing has received considerable attention in recent years. especially in the context of relational data [1, 7, 12, 18–20, 23, 25, 36]. All these works assume a given set of attributes QID on which an individual is identified, and anonymize data records on the QID. Their main difference consist in the selected privacy model and in various approaches employed to anonymize the data. The author of [1] presents a study on the relationship between the dimensionality of QID and information loss, and concludes that, as the dimensionality of QID increases, information loss increases quickly. Transactional databases present exactly the worst case scenario for existing anonymisation approaches because of high dimension of QID. To our best knowledge, all existing solutions in the context of $k$-anonymity [26, 27], $l$-diversity [23] and $t$-closeness [22] assume a relational table, which typically has a low dimensional QID.

There are few previous work considering the privacy of large rating data. In collaboration with MovieLens recommendation service, [11] correlated public mentions of movies in the MovieLens discussion forum with the users' movie rating histories in the internal MovieLens data set. Recent study reveals a new type of attack on anonymized data for transactional data [24]. Movie rating data supposedly to be anonymized is re-identified by linking non-anonymized data from other source. No solution exists for high dimensional large scale rating databases.

Privacy-preservation of transactional data has been acknowledged as an important problem in the data mining literature. There us a family of literature [5, 6] addressing the privacy threats caused by publishing data mining results such as frequent item sets and association rules. Existing works on topic [4, 34] focus on publishing patterns, The patterns are mined from the original data, and the resulting set of rules is sanitized to present privacy breaches. In contrast, our work addresses the privacy threats caused by publishing data for data mining. As discussed above, we do not assume that the data publisher can perform data mining tasks, and we assume that the data must be made available to the recipient. The two scenarios have different assumptions on the capability of the data publisher and the information requirement of the data recipient. The recent work on topic [14, 37] focus on high dimensional transaction data, while our focus is preventing linking individuals to their ratings.

This paper is also related to the work on anonymizing social networks [8], and the large scale rating data can be considered as a special case of the complex social network. A social network is a graph in which a node represents a social entity (e.g., a person) and an edge represents a relationship between the social entities. Although the data is very different from transaction data, the model of attacks is similar to ours: An attacker constructs a small subgraph connected to a target individual and then matches the subgraph to the whole social network, attempting to re-identify the target individual's node, and therefore, other unknown connection to the node. [8] demonstrates the severity of privacy threats in nowadays social networks, but does not provide a solution to prevent such attacks.

## 4. Privacy models

The auxiliary information of an attacker includes: (i) the knowledge that a victim is in the survey rating data; (ii) preferences of the victims on some non-sensitive issues. The attacker wants to find ratings on sensitive issues of the victim.

In practice, knowledge of Types (i) and (ii) can be gleaned from an external database [24]. For example, in the context of Table 1(b), an external database may be the IMDb. By examining the anonymous data set in Table 1(a), the adversary can identify a small number of candidate groups that contain the record of the victim. It will be the unfortunate scenario where there is only one record in the candidate group. For example, since $t_1$ is unique in Table 1(a), Alice is at risk of being identified. If the candidate group contains not only the victim but other records, an adversary may use this group to infer the sensitive value of the victim individual. For example, although it is difficult to identify whether $t_2$ or $t_3$ in Table 1(a) belongs to Bob, since both records have the same sensitive value, Bob's private information is identified.

Intuitively,in order to avoid such attack, a two-step protection model is needed. The first step is to protect individual's identity, which is to make sure that in the released data set, every transaction should be "similar" to at least to $(k-1)$ other records based on the non-sensitive ratings so that no survey participants are identifiable. For example, $t_1$ in Table 1(a) is unique, and based on the preference of Alice in Table 1(b), her sensitive issues can be re-identified in the supposed anonymized data set. Jack's sensitive issues, on the other hand, is much safer. Since $t_4$ and $t_5$ in Table 1(a) form a similar group based on their non-sensitive rating.

The second step is to prevent the sensitive rating from being inferred in an anonymized data set. The idea is to require that the sensitive ratings in a similar group should be diverse. For example, although $t_2$ and $t_3$ in Table 1(a) form a similar group based on their non-sensitive rating, their sensitive ratings are identical. Therefore, an attacker can immediately infer Bob's preference on the sensitive issue without identifying which transaction belongs to Bob. In contrast, Jack's preference on the sensitive issue is much safer than both Alice and Bob.

In our previous work, two privacy models have been proposed. The first one is $(k,\epsilon)$-anonymity model, which targets at protecting individual's identity and the second model is $(k,\epsilon,l)$-anonymity model, which not only protects individual's identity, but also the personal sensitive information. In the next, section, these two models will be discussed.

### 4.1. $(k,\epsilon)$-anonymity

Let $T_A = \{o_{A_1}, o_{A_2}, \cdots, o_{A_p}, s_{A_1}, s_{A_2}, \cdots, s_{A_q}\}$ be the ratings for a survey participant $A$ and $T_B = \{o_{B_1}, o_{B_2}, \cdots, o_{B_p}, s_{B_1}, s_{B_2}, \cdots, s_{B_q}\}$ be the ratings for a participant $B$. We define the dissimilarity between two non-sensitive ratings as follows.

$$Dis(o_{A_i}, o_{B_i}) = \begin{cases} |o_{A_i} - o_{B_i}| & \text{if } o_{A_i}, o_{B_i} \in \{1:r\} \\ 0 & \text{if } o_{A_i} = o_{B_i} = null \\ r & \text{otherwise} \end{cases} \quad (1)$$

**Definition 1** ($\epsilon$-proximate). Given a survey rating data set $T$ with a small positive number $\epsilon$, two transactions $T_A$, $T_B \in T$, where $T_A = \{o_{A_1}, o_{A_2}, \cdots, o_{A_p}, s_{A_1}, s_{A_2}, \cdots, s_{A_q}\}$ and $T_B = \{o_{B_1}, o_{B_2}, \cdots, o_{B_p}, s_{B_1}, s_{B_2}, \cdots, s_{B_q}\}$. We say $T_A$ and $T_B$ are $\epsilon$-proximate, if $\forall\ 1 \le i \le p$, $Dis(o_{A_i}, o_{B_i}) \le \epsilon$. We say $T$ is $\epsilon$-proximate, if every two transactions in $T$ are $\epsilon$-proximate.

If two transactions are $\epsilon$-proximate, the dissimilarity between their non-sensitive ratings is bounded by $\epsilon$. In our running example, suppose $\epsilon = 1$, ratings 5 and 6 may have no difference in interpretation, so $t_4$ and $t_5$ in Table 1(a) are 1-proximate based on their non-sensitive rating. If a group of transactions are in $\epsilon$-proximate, then the dissimilarity between each pair of their non-sensitive ratings is bounded by $\epsilon$. For example, if $T = \{t_1, t_2, t_3\}$, then it is easy to verify that $T$ is 5-proximate.

**Definition 2** ($(k,\epsilon)$-anonymity). A survey rating data set $T$ is said to be $(k,\epsilon)$-anonymous if every transaction is $\epsilon$-proximate with at least $(k-1)$ other transactions. The transaction $t \in T$ with all the other transactions that $\epsilon$-proximate with $t$ form a $(k,\epsilon)$-anonymous group.

For instance, there are two (2,5)-anonymous groups in Table 1(a). The first one is formed by $\{t_1, t_2, t_3\}$ and the second one is formed by $\{t_4, t_5\}$. The idea behind this privacy principle is to make each transaction contains non-sensitive attributes are similar with other transactions in order to avoid linking to personal identity. $(k,\epsilon)$-anonymity well preserves identity privacy. It guarantees that no individual is identifiable with the probability greater than the probability of $1/k$. Both parameters $k$ and $\epsilon$ are intuitive and operable in real-world applications. The parameter $\epsilon$ captures the protection range of each identity, whereas the parameter $k$ is to lower an adversary's chance of beating that protection. The larger the $k$ and $\epsilon$ are, the better protection it will provide.

Although $(k,\epsilon)$-anonymity privacy principle can protect people's identity, it fails to protect individuals' private information. Let us consider one $(k,\epsilon)$-anonymous group. If the transactions of the group have the same rating on a number of sensitive issues, an attacker can know the preference on the sensitive issues of each individual without knowing which transaction

belongs to whom. For example, in Table 1(a), $t_2$ and $t_3$ are in a $(2,1)$-anonymous group, but they have the same rating on the sensitive issue, and thus Bob's private information is breaching.

### 4.2. $(k, \epsilon, l)$-anonymity

This example illustrates the limitation of the $(k, \epsilon)$-anonymity model. To mitigate the limitation, we require more diversity of sensitive ratings in the anonymous groups. In the following, we define the distance between two sensitive ratings, which leads to the metric for measuring the diversity of sensitive ratings in the anonymous groups.

First, we define dissimilarity between two sensitive rating scores as follows.

$$Dis(s_{A_i}, s_{B_i}) = \begin{cases} |s_{A_i} - s_{B_i}| & \text{if } s_{A_i}, s_{B_i} \in \{1:r\} \\ r & \text{if } s_{A_i} = s_{B_i} = null \\ r & \text{otherwise} \end{cases} \quad (2)$$

Note that there is only one difference between dissimilarities of sensitive ratings $Dis(s_{A_i}, s_{B_j})$ and dissimilarities of non-sensitive ratings $Dis(o_{A_i}, o_{B_j})$, that is, in the definition of $Dis(o_{o_i}, o_{o_j})$, $null - null = 0$, and for the definition of $Dis(s_{A_i}, s_{B_j})$, $null - null = r$. This is because for sensitive issues, two *null* ratings mean that an attacker will not get information from two survey participants, and hence are good for the diversity of the group.

Next, we introduce the metric to measure the diversity of sensitive ratings. For a sensitive issue $s$, let the vector of ratings of the group be $[s_1, s_2, \cdots, s_g]$, where $s_i \in \{1:r, null\}$. The means of the ratings is defined as follows:

$$\bar{s} = \frac{1}{Q} \sum_{i=1}^{g} s_i$$

where $Q$ is the number of non-*null* values, and $s_i \pm null = s_i$. The standard deviation of the rating is then defined as:

$$SD(s) = \sqrt{\frac{1}{g} \sum_{i=1}^{g} (s_i - \bar{s})^2} \quad (3)$$

For instance in Table 1(a), for the sensitive issue 4, the means of the ratings is $(6+1+1+1+5)/5 = 2.8$ and the standard deviation of the rating is 2.23 according to Equation (3).

**Definition 3** ($(k, \epsilon, l)$-anonymity). A survey rating data set is said to be $(k, \epsilon, l)$-anonymous if and only if the standard deviation of ratings for each sensitive issue is at least $l$ in each $(k, \epsilon)$-anonymous group.

Still consider Table 1(a) as an example. $t_4$ and $t_5$ is 1-proximate with the standard deviation of 2. If we set $k = 2, l = 2$, then this group satisfies (2,1,2)-anonymity requirement. The $(k, \epsilon, l)$-anonymity requirement allows sufficient diversity of sensitive issues in $T$, therefore it could prevent the inference from the $(k, \epsilon)$-anonymous groups to a sensitive issue with a high probability. The following theorem gives the upper bound of the parameter $l$ in the $(k, \epsilon, l)$-anonymity model. The proof of the following theorem can be found in [30].

**Theorem 1.** Let $S$ be the set of ratings of the sensitive issue of $T$. Suppose $S\_min$ and $S\_max$ be the minimum and maximum ratings in $S$, then the maximum standard deviation of $S$ is $\frac{(S\_max - S\_min)}{2}$.

## 5. Validating privacy requirements

In this section, we formulate the satisfaction problem and develop a slicing technique to determine the following *Satisfaction Problem*.

**Problem 1** (Satisfaction Problem). Given a survey rating data set $T$ and privacy requirements $k, \epsilon, l$, the satisfaction problem of $(k, \epsilon, l)$-anonymity is to decide whether $T$ satisfies the $k, \epsilon, l$ privacy requirements.

The satisfaction problem is to determine whether the user's given privacy requirement is satisfied by the given data set. It is a very important step before anonymizing the survey rating data. If the data set has already met the requirements, it is not necessary to make any modifications before publishing. As follows, we propose a novel slice technique to solve the satisfaction problem.

### 5.1. Search by slicing

The slicing technique is proposed to efficiently search for the neighbor within distance $\epsilon$ in high dimension. As we shall see, the complexity of the proposed algorithm grows very slowly with dimension for small $\epsilon$. We illustrate the proposed slicing technique using a simple example in 3-D space, as shown in Figure 1. Given $t = (t_1, t_2, t_3) \in T$, our goal is to slice out a set of transactions $T$ $(t \in T)$ that are $\epsilon$-proximate. Our approach is first to find the $\epsilon$-proximate of $t$, which is the set of transactions that lie inside a cube $C_t$ of side $2\epsilon$ centered at $t$. Since $\epsilon$ is typically small, the number of points inside the cube is also small. The $\epsilon$-proximate of $C_t'$ can then be found by an exhaustive comparison within the $\epsilon$-proximate of $t$. If there are no transactions inside the cube $C_t$, we know that the $\epsilon$-proximate of $t$ is empty, so as the $\epsilon$-proximate of the set $C_t'$.

The transactions within the cube can be found as follows. First we find the transactions that are sandwiched between a pair of parallel planes $X_1$, $X_2$ (See Figure 1) and add them to a *candidate set*. The planes are perpendicular to the first axis of coordinate

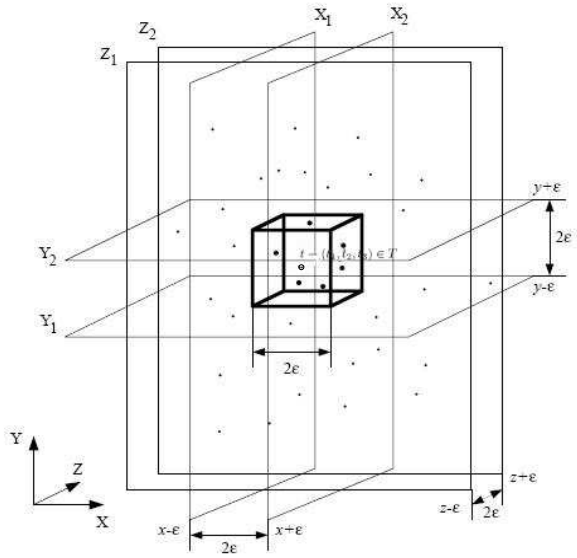



**Figure 1.** The slicing technique finds a set of transactions $C_t$ inside a cube of size $2\epsilon$ within the $\epsilon$-proximate of $t$. The $\epsilon$-proximate of the set $C_t$ can then be found by an exhaustive search in the cube.

frame and are located on either side of the transaction $t$ at a distance of $\epsilon$. Next, we trim the candidate set by disregarding transactions that are not also sandwiched between the parallel pair of $Y_1$ and $Y_2$, that are perpendicular to $X_1$ and $X_2$, again located on either side of $t$ at a distance of $\epsilon$. This procedure is repeated for $Z_1$ and $Z_2$ at the end of which, the candidate set contains only transactions within the cube of size $2\epsilon$ centered at $t$.

Since the number of transactions in the final $\epsilon$-proximate is typically small, the cost of the exhaustive comparison is negligible. The major computational cost in the slicing process occurs therefore in constructing and trimming the candidate set.

## 6. Anonymous survey rating data

In this section, we describe our modification strategies through the graphical representation of the $(k, \epsilon)$-anonymity model. Given a survey rating data set $T = \{t_1, t_2, \cdots, t_n\}$, its graphical representation is the graph $G = (V, E)$, where $V$ is a set of nodes, and each node in $V$ corresponds to a record $t_i$ $(i = 1, 2, \cdots, n)$ in $T$, and $E$ is the set of edges, where two nodes are connected by an edge if and only if the distance between two records is bounded by $\epsilon$ with respect to the non-sensitive ratings.

Two nodes $t_i$ and $t_j$ are called connected if $G$ contains a path from $t_i$ to $t_j$ $(1 \le i, j \le n)$. The graph $G$ is called connected if every pair of distinct nodes in the graph can be connected through some paths. A connected component is a maximal connected subgraph of $G$. Each node belongs to exactly one connected component, as

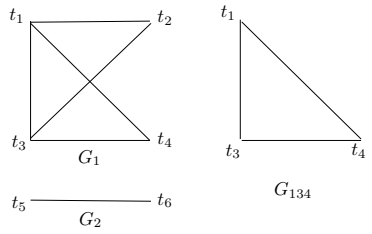



**Figure 2.** Graphical representation example

does each edge. The degree of the node $t_i$ is the number of edges incident to $t_i$ $(1 \le i \le n)$.

We say $G$ is a clique if every pair of distinct nodes is connected by an edge. The $k$-clique is a clique with at least $k$ nodes. The maximal $k$-clique is the a $k$-clique that is not a subset of any other $k$-clique. We say the connected component $G = (V, E)$ is $k$-decomposable if $G$ can be decomposed into several $k$-cliques $G_i = (V_i, E_i)$ $(i = 1, 2, \cdots, m)$, and satisfies $V_i \bigcap V_j = \emptyset$ for $(i \ne j)$, $\bigcup_{i=1}^{m} V_i = V$, and $\bigcup_{i=1}^{m} E_i \subseteq E$. The graph is $k$-decomposable if all its connected components are $k$-decomposable.

**Theorem 2.** Given the survey rating data set $T$ with its graphical representation $G$, if $G$ is $k$-decomposable, then $T$ is $(k, \epsilon)$-anonymous.

The proof of Theorem 2 can be found in [29]. For instance, the survey rating data shown in Table 2 is $(2, 2)$-anonymous since its graphical representation (Figure 2(a)) is 2-decomposable. With Theorem 2, to make the rating data satisfy privacy requirement, it only needs to make its graphical representation $k$-decomposable.

### 6.1. Distortion Metrics

In this section, we define a measure to capture the information loss.

**Definition 4** (Tuple distortion). Let $t = (t_1, t_2, \cdots, t_m)$ be a tuple and $t' = (t'_1, t'_2, \cdots, t'_m)$ be an anonymized tuple of $t$. Then, the distortion of this anonymisation is defined as:

$$\text{Distortion}(t, t') = \sum_{i=1}^{m} |t_i - t'_i|$$

For example, if the tuple $t = (5, 6, 0)$ is generalized to $t' = (5, 5, 0)$, then the distortion of this anonymisation is $|5 - 5| + |6 - 5| + |0 - 0| = 1$.

**Definition 5** (Total distortion). Let $T' = (t'_1, t'_2, \cdots, t'_n)$ be the anonymized data set from $T = (t_1, t_2, \cdots, t_n)$. Then, the total distortion of this anonymisation is defined as:

$$\text{Distortion}(T, T') = \sum_{i=1}^{n} \text{Distortion}(t_i, t'_i)$$

| | non-sensitive | | | sensitive |
|---|---|---|---|---|
| ID | issue 1 | issue 2 | issue 3 | issue 4 |
| $t_1$ | 3 | 6 | *null* | 6 |
| $t_2$ | 2 | 5 | *null* | 1 |
| $t_3$ | 4 | 7 | *null* | 4 |
| $t_4$ | 5 | 6 | *null* | 1 |
| $t_5$ | 1 | *null* | 5 | 1 |
| $t_6$ | 2 | *null* | 6 | 5 |

**Table 2.** Sample survey rating data (I)

| | non-sensitive | | | sensitive |
|---|---|---|---|---|
| ID | issue 1 | issue 2 | issue 3 | issue 4 |
| $t_1$ | 3 | 6 | *null* | 6 |
| $t_2$ | 2 | 5 | *null* | 1 |
| $t_3$ | 4 | 7 | *null* | 4 |
| $t_4$ | 5 | 6 | *null* | 1 |
| $t_5$ | 1 | *null* | 5 | 1 |
| $t_6$ | 2 | *null* | 6 | 5 |
| $t_7$ | 6 | *null* | 6 | 3 |
| $t_8$ | 5 | *null* | 5 | 2 |

**Table 3.** Sample survey rating data (II)

For example, let $T = (t_1, t_2, t_3, t_4)$, where $t_1 = (5, 6, 0)$, $t_2 = (2, 5, 0)$, $t_3 = (4, 7, 0)$ and $t_4 = (5, 6, 0)$. Let $T' = (t'_1, t'_2, t'_3, t'_4)$ be anonymization of $T$, where $t'_1 = (5, 5, 0)$, $t'_2 = (3, 5, 0)$, $t'_3 = (3, 7, 0)$ and $t'_4 = (5, 7, 0)$. Then, the distortion between the two data sets is $1 + 1 + 1 + 1 = 4$.

For ease, we first illustrate our approach in the scale of single attribute, and then we extend it to multiple attributes.

Let $t = (t_1, t_2, \cdots, t_n)$ be the ratings of some issue from $n$ survey participants with the privacy requirement $\epsilon$. We assume that some ratings in $t$ are not bounded by $\epsilon$, and our aim is to modify $t$ to make every pair of ratings is bounded by $\epsilon$ while minimizing the distortion. The idea of the approach is as follows. Order all ratings for the issue $t$, and find the minimum rating $Min$ and maximum rating $Max$. Find all intervals of the size $\epsilon$ between $Min$ and $Max$. Change the ratings that does not fit in this interval such that the distortion is minimized. In the case of some tuples with the same minimum distortion, randomly pick up one of them as the anonymization. The process is described in **Algorithm 1**.

**Algorithm 1**: $single\_anonymizer(t, \epsilon)$

**Input**: an ascended tuple $t = (t_1, \cdots, t_n)$, and $\epsilon$
**Output**: $t' = (t'_1, \cdots, t'_n)$ with minimum distortion
/* Computing distortions for all intervals */
**for** $i \leftarrow 1$ **to** $\frac{t_n - t_1}{\epsilon}$
  **do for** $j \leftarrow 1$ **to** $n$
    **do if** $t_j \in (t_i, t_i + \epsilon)$
      **then** $t'_j \leftarrow t_j$
      **else if** $t_j < t_i$
        $t'_j \leftarrow t_i$
      **else** $t'_j \leftarrow t_i + \epsilon$
  $D(i) \leftarrow Distortion(t', t)$;
/* Finding minimum distortion */
$k \leftarrow 1; D_{min} \leftarrow D(k)$;
**for** $i \leftarrow 2$ **to** $\frac{t_n - t_1}{\epsilon}$
  **do if** $D(i) < D_{min}$
    **then** $D_{min} \leftarrow D(i)$;
      $k \leftarrow i$;
/* Retrieving $t'$ with minimum distortion */
**for** $i \leftarrow 1$ **to** $n$
  **do if** $t_i \in (t_k, t_k + \epsilon)$
    **then** $t'_i \leftarrow t_i$
    **else if** $t_i < t_k$
      $t'_i \leftarrow t_k$
    **else** $t'_i \leftarrow t_k + \epsilon$
**return** $t'$

For example, if $t = (3, 4, 5, 6, 7, 7, 8, 8)$ and $\epsilon = 2$. The $Min$ is 3 and $Max$ is 8. Build all the intervals with the size of 2, which are (3,5), (4,6), (5,7) and (6,8). Following Algorithm 1, the anonymization of $t$ is shown in Table 4, in which the vector in bold is the anonymisation we choose.

| Intervals | Anonymization | Distortion |
|---|---|---|
| (3, 5) | (3, 4, 5, 5, 5, 5, 5, 5) | 11 |
| (4, 6) | (4, 4, 5, 6, 6, 6, 6, 6) | 7 |
| (5, 7) | **(5, 5, 5, 6, 7, 7, 7, 7)** | **5** |
| (6, 8) | (6, 6, 6, 6, 7, 7, 8, 8) | 6 |

**Table 4.** Example of the anonymization algorithm

Let us take Table 1(a) as an example with $k = 2, \epsilon = 1$. There are two groups $HG_1 = \{t_1, t_2, t_3, t_4\}$ and $HG_2 = \{t_5, t_6\}$. $HG_2$ has already satisfied the privacy requirement, but $HG_1$ does not. The anonymization of $HG_1$ is shown in Table 5, in which the vector in bold is the anonymisation we choose.

| | Intervals | Anonymization | Distortion |
|---|---|---|---|
| Issue 1 | (2,3) | (3,3,3,2) | 4 |
| | (3,4) | **(4,3,4,3)** | 3 |
| | (4,5) | (5,4,4,4) | 4 |
| | (5,6) | (6,5,5,5) | 6 |
| | Intervals | Anonymization | Distortion |
| Issue 2 | (1,2) | (1,2,2,2) | 10 |
| | (2,3) | (2,3,3,3) | 8 |
| | (3,4) | (3,4,4,4) | 6 |
| | (4,5) | **(4,5,5,5)** | 4 |
| | (5,6) | (5,6,5,5) | 4 |

**Table 5.** Anonymizing $HG_1$ of Table 1(a)

## 6.2. Complexity analysis

Recall that our objective is to anonymize data consisting of a set of transactions $T = \{t_1, t_2, \cdots, t_n\}$, $|T| = n$. Each

transaction $t_i \in T$ contains $m$ issues. The computation cost consists of three parts, which are sorting, finding intervals and computing distortion. The complexity of the sorting is $O(mn\log n)$. During the next phrase of the algorithm, for each attribute, we find the $Min$ and $Max$ and all the possible intervals with size $\epsilon$, which incur the amount of $O(2(n-1))$ overhead, and the cost for comparisons to search the one with least distortion is $O(n)$. So, the total complexity of all attributes in this phrase is $O(mn)$. The last phrase to compare original and anonymous data sets to estimate the distortion has the cost of $O(mn)$. The computational complexity of this alternative approach is $O(mn\log n + mn)$.

## 7. Experimental study

In this section, we experimentally evaluate the efficiency of the proposed slicing algorithm and the proposed anonymization algorithm. Our objectives are two-fold. First, we verify that our slice algorithm is fast and scalable for the satisfaction problem. Second, we show that the slicing technique is not only time efficient, but also space efficient compared with the heuristic pairwise algorithm.

### 7.1. Data sets

Our experimentation deploys two real-world databases. MovieLens[1] and Netflix data sets[2]. MovieLens data set was made available by the GroupLens Research Project at the University of Minnesota. The data set contains 100,000 ratings (5-star scale), 943 users and 1682 movies. Each user has rated at lease 20 movies. Netflix data set was released by Netflix for competition. The movie rating files contain over 100,480,507 ratings from 480,189 randomly-chosen, anonymous Netflix customers over 17 thousand movie titles. The data were collected between October, 1998 and December, 2005 and reflect the distribution of all ratings received during this period. The ratings are on a scale from 1 to 5 (integral) stars. In both data sets, a user is considered as an object while a movie is regarded as an attribute and many entries are empty since a user only rated a small number of movies. Except for rating movies, users' ratings some simple demographic information (e.g., age range) are also included. In our experiments, we treat the users' ratings on movies as non-sensitive issues and ratings on others as sensitive ones.

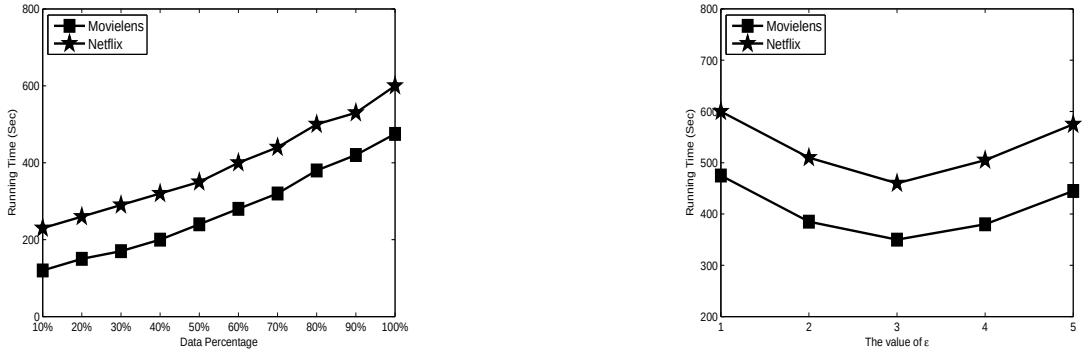
### 7.2. Efficiency

Data used for Figure 3(a) is generated by re-sampling the Movielens and Netflix data sets while varying the percentage of data from 10% to 100%. For both data sets, we evaluate the running time for the $(k, \epsilon, l)$-anonymity model with default setting $k = 20, \epsilon = 1, l = 2$. For both testing data sets, the execution time for $(k, \epsilon, l)$-anonymity is increasing with the increased data percentage. This is because as the percentage of data increases, the computation cost increases too. The result is expected since the overhead is increased with the more dimensions.

Next, we evaluate how the parameters affect the cost of computing. Data set used for this sets of experiments are the whole sets of MovieLens and Netflix data and we evaluate by varying the value of $\epsilon$, $k$ and $l$. With $k = 20, l = 2$, Figure 3(b) shows the computational cost as a function of $\epsilon$, in determining $(k, \epsilon, l)$-anonymity requirement of both data sets. Interestingly, in both data sets, as $\epsilon$ increases, the cost initially becomes lower but then increases monotonically. This phenomenon is due to a pair of contradicting factors that push up and down the running time, respectively. At the initial stage, when $\epsilon$ is small, more computation efforts are put into finding $\epsilon$-proximate of the transaction, but less used in exhaustive search for proper $\epsilon$-proximate neighborhood, and this explains the initial decent of overall cost. On the other hand, as $\epsilon$ grows, there are fewer possible $\epsilon$-proximate neighborhoods, thus reducing the searching time for this part, but the number of transactions in the $\epsilon$-proximate neighborhood is increased, which results in huge exhaustive search for proper $\epsilon$-proximate neighborhood and this causes the eventual cost increase. Setting $\epsilon = 2$, Figure 4(a) displays the results of running time by varying $k$ from 10 to 60 for both data sets. The cost drops as $k$ grows. This is expected, because fewer search efforts for proper $\epsilon$-proximate neighborhoods needed for a greater $k$, allowing our algorithm to terminate earlier. We also run the experiment by varying the parameter $l$ and the results are shown in Figure 4(b). Since the rating of both data sets are between 1 and 5, then according to Theorem 1, 2 is already the largest possible $l$. When $l = 0$, there is no diversity requirement among the sensitive issues, and the $(k, \epsilon, l)$-anonymity model is reduced to $(k, \epsilon)$-anonymity model. As we can see, the running time increases with $l$, because more computation is needed in order to enforce stronger privacy control.

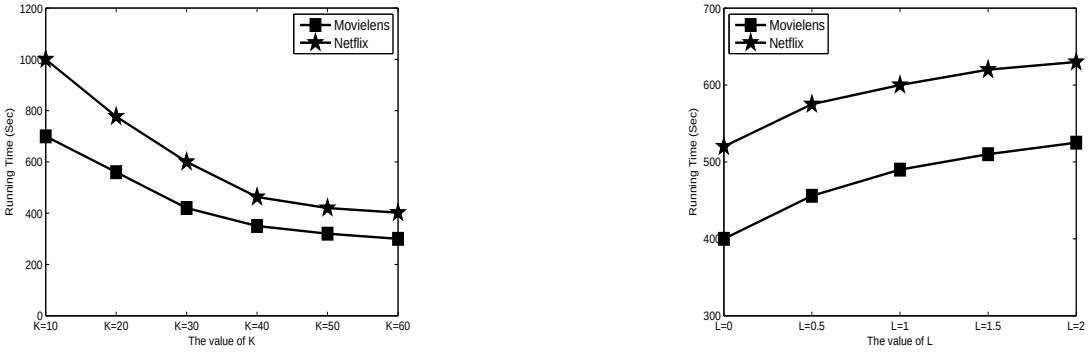
In addition to show the scalability and efficiency of the slicing algorithm itself, we also experimented the comparison between the slicing algorithm (Slicing) and the heuristic pairwise algorithm (Pairwise), which works by computing all the pairwise distance to construct the dissimilarity matrix and identify the violation of the privacy requirements. We implemented both algorithms and studied the impact of the execution time on the data percentage, the value of $\epsilon$, the value of $K$ and the value of $L$.

[1] http://www.grouplens.org/taxonomy/term/14.
[2] http://www.netflixprize.com/.



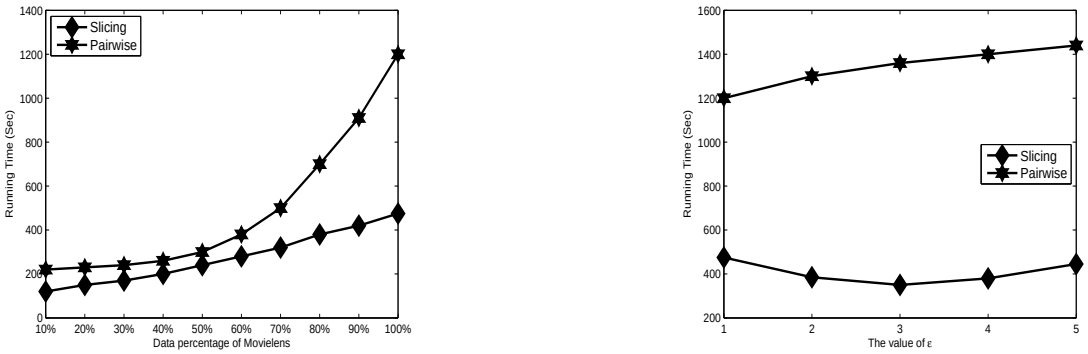

(a) (b)

**Figure 3.** Running time comparison on Movielens and Netflix data sets vs. (a) Data percentage varies (b) $\epsilon$ varies




(a) (b)

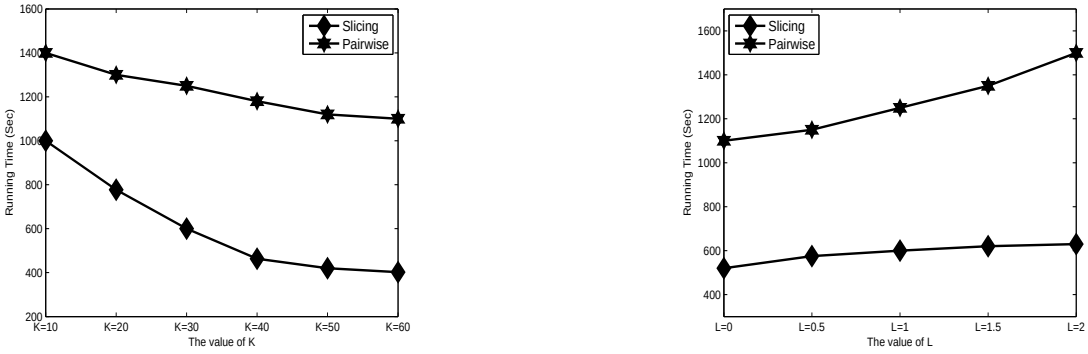
**Figure 4.** Running time comparison on Movielens and Netflix data sets vs. (c) $k$ varies (d) $L$ varies




(a) (b)

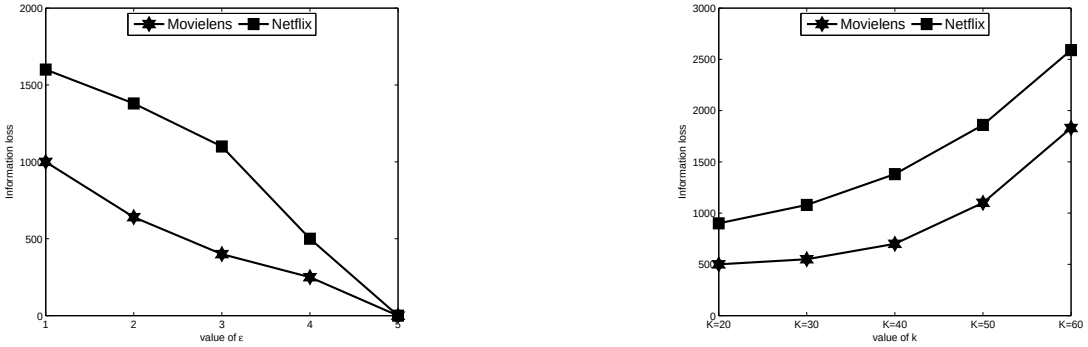
**Figure 5.** Running time comparison of Slicing and Pairwise methods on Movielens data set vs. (a) Data percentage varies (b) $\epsilon$ varies

Figure 5 plots the running time of both slicing and pairwise algorithms on the Movielens data set. Figure 5(a) describe the trend of the algorithms by varying the percentage of the data set. From the graph we can see, the slicing algorithm is far more efficient than the heuristic pairwise algorithm especially when the volume of the data becomes larger. This is because, when the dimension of the data increases, the disadvantage of the heuristic pairwise algorithm, which is to compute all the dissimilarity distance, dominates the most of the execution time. On the other hand, the smarter grouping technique used in the slicing process makes less computation cost for the slicing algorithm. The similar trend is shown in Figure 5(b) by varying the



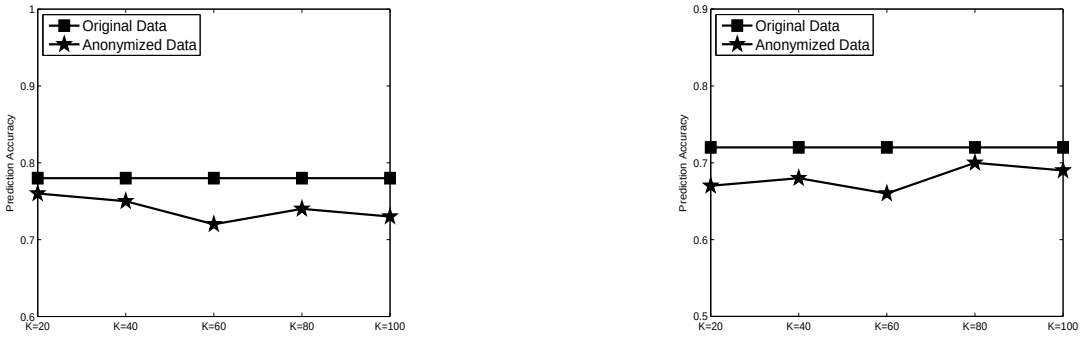

(c) (d)

**Figure 6.** Running time comparison of Slicing and Pairwise methods on Netflix data set vs. (c) $k$ varies (d) $L$ varies




(a) (b)

**Figure 7.** Information loss comparison on Movielens and Netflix databases vs. (a) $k$ varies; (b) $\epsilon$ varies




(a) (b)

**Figure 8.** Prediction Accuracy: (a) Movielens; (b) Netflix

value of $\epsilon$, in which the slicing algorithm is almost 3 times faster than the the heuristic pairwise algorithm. The running time comparisons of both algorithms in Netflix data set by varying the value of $K$ and $L$ are shown in Figure 6(a) and (b). Even on a larger data set, the slicing algorithm outperformed the pairwise algorithm, and the running time of Slicing is quick enough to be used in practical.

## 7.3. Data Utility

Having verifying the efficiency of the slicing technique, we proceed to test its effectiveness. We measure the utility by the distortion metric defined in previous

sections. Generally speaking, the more the distortion is, the less useful the anonymized data would be.

We first study the influence of $\epsilon$ (i.e., the length of a proximate neighborhood) on data utility. Towards this, we set $k$ to 40. Concerning $(40, \epsilon)$-anonymity, Figure 7(a) plots the information loss on both data sets as a function of $\epsilon$. The anonymization algorithm incurs less distortion as $\epsilon$ increases. This is expected, since a smaller $\epsilon$ demands stricter privacy preservation, which reduces data utility. When $\epsilon = 5$, there will be no anonymization required, and therefore the information loss reaches 0. Next, we examine the utility of $(k, 2)$-anonymous solution with different $k$. Figure 7(b) presents the information loss as a function of $k$. The error grows with $k$ because a larger $k$ demands tighter anonymity control requiring much more data modification.

Figures 8(a) and (b) evaluate the classification and prediction accuracy of the greedy anonymization algorithm. Our evaluation methodology is that we first divide data into training and testing sets, and we apply the anonymization algorithm to the training and testing sets to obtain the anonymized training and testing sets, and finally the classification or regression model is trained by the anonymized training set and tested by anonymized testing set. The Weka implementation [35] of simple Naive Bayes classifier was used for the classification and prediction. Using the Movielens data, Figure 8(a) compares the predictive accuracy of classifier trained on Movielens data produced by the greedy anonymization algorithm. In these experiments, we generated 50 independent training and testing sets, each containing 2000 records, and we fixed $\epsilon = 2$. The results are averaged across these 50 trials. For comparison, we also include the accuracies of classifier trained on the (not anonymized) original data. From the graph, we can see that the average prediction accuracy is around 75%, very close to the original accuracy, which preserves better utility for data mining purposes. Similar results are obtained by using the Netflix rating data in Figure 8(b).

## 8. Conclusion and future work

We have studied the problems of protecting sensitive ratings of individuals in a large scale rating data. Such privacy risk has emerged in a recent study on the de-identification of published movie rating data. We proposed a novel $(k, \epsilon, l)$-anonymity privacy principle for protecting privacy in such survey rating data. We theoretically investigated the properties of this model, and studied the satisfaction problem, which is to decide whether a survey rating data set satisfies the privacy requirements given by the user. A greedy anonymization algorithm has been proposed to anonymize large scale rating data. Extensive experiments confirm that our technique produces anonymized data sets that are useful.

This work also initiates the future investigations of approaches on anonymizing the survey rating data. Traditional approaches on anonymizing no matter relational data sets or transactional data set are by generalization or suppression, and the published data set has the same number of data but with some fields being modified to meet the privacy requirements. As shown in the literatures, this kind of anonymization problem is normally NP-hard, and several algorithms are devised along this framework to minimize the certain pre-defined cost metrics. Inspired by the research in this paper, the satisfaction problem can be further used to develop a different method to anonymizing the data set. The idea is straightforward with the result of the satisfaction problem. If the rating data set has already satisfies the privacy requirement, it is not necessary to do any anonymization to publish it. Otherwise, we anonymize the data set by deleting some of the records to make it meet the privacy requirement. The criteria during the deletion can be various (for example, to minimize the number of deleted records) to make it as much as useful in the data mining or other research purposes. We believe that this new anonymization method is flexible in the choice of privacy parameters and efficient in the execution with the practical usage.

## References


[1] C. Aggarwal. On $k$-Anonymity and the curse of dimensionality. VLDB 2005.

[2] R. Agrawal and R. Srikant. Privacy-Preserving Data Mining. SIGMOD 2000.

[3] D. Agrawal and C. C. Aggarwal. On The Design and Qualification of Privacy Preserving Data Mining Algorithm. Proc. Symosium on Principles of Database Systems (PODS), pp247-255, 2001.

[4] M. Atzori, F. Bonchi, F. Giannotti and D. Pedreschi. Anonymity preserving pattern discovery. VLDB J. 17(4): 703-727 (2008)

[5] M. Atzori, F. Bonchi, F. Giannotti, and D. Pedreschi. Blocking anonymity threats raised by frequent itemset mining. ICDM 2005.

[6] M. Atzori, F. Bonchi, F. Giannotti, and D. Pedreschi. $k$-anonymous patterns. PKDD 2005.

[7] R. J. Bayardo and R. Agrawal. Data privacy through optimal $k$-anonymisation. ICDE 2005.

[8] L. Backstrom, C. Dwork and J. Kleinberg. Wherefore Art Thou R3579x?: Anonymized Social Networks, Hidden Patterns, and Structural Steganography. WWW 2007.

[9] R. Evfimievski, R. Srikant, R. Agrawal, and J. Gehrke. Privacy preserving mining of association rules. SIGKDD 2002.

[10] J. K. Friedman, J. L. Bentley, R. A. Finkel. An algorithm for finding best matches in logarithmic expected time, ACM Trans. on Math. Software, 3(1977), pp. 209–226.

[11] D. Frankowski, D. Cosley, S. Sen, L. G. Terveen and J. Riedl. You are what you say: privacy risks of public mentions. SIGIR 2006: 565-572

[12] B. C. Fung, K. Wang, and P. S. Yu. Top-down specialization for information and privacy preservation. ICDE 2005.

[13] M. R. Garey and D. S. Johnson. Computers and Intractability: A Guide to the Theory of $\mathcal{NP}$-Completeness. San Francisco. Freeman, 1979

[14] G. Ghinita, Y. Tao and P. Kalnis. On the Anonymisation of Sparse High-Dimensional Data, In Proceedings of International Conference on Data Engineering (ICDE) April 2008.

[15] K. Hafner. And if you liked the movie, a Netflix contest may reward you handsomely. New York Times, Oct 2 2006.

[16] S. Hansell. AOL removes search data on vast group of web users. New York Times, Aug 8 2006.

[17] R. W. Hamming. Coding and Information Theory, Englewood Cliffs, NJ, Prentice Hall (1980)

[18] V. Iyengar. Transforming data to satisfy privacy constraints. SIGKDD 2002.

[19] K. LeFevre, D. DeWitt, and R. Ramakrishnan. Incognito: efficient full-domain $k$-anonymity. SIGMOD 2005.

[20] K. LeFevre, D. DeWitt, and R. Ramakrishnan. Mondrian multidimensional $k$-anonymity. ICDE 2006.

[21] J. Li, Y. Tao and X. Xiao. Preservation of Proximity Privacy in Publishing Numerical Sensitive Data. ACM Conference on Management of Data (SIGMOD), 2008

[22] N. Li, T. Li and S. Venkatasubramanian. $t$-Closeness: Privacy Beyond $k$-anonymity and $l$-diversity. *ICDE 2007*: 106-115

[23] A. Machanavajjhala, J. Gehrke, D. Kifer, and M. Venkitasubramaniam. $l$-Diversity: Privacy beyond $k$-anonymity. ICDE 2006.

[24] A. Narayanan and V. Shmatikov. Robust De-anonymisation of Large Sparse Datasets. to appear in IEEE Security & Privacy 2008.

[25] P. Samarati and L. Sweeney. Generalizing data to provide anonymity when disclosing Information. PODS 1998.

[26] P. Samarati and L. Sweeney. Protecting privacy when disclosing information: $k$-anonymity and its enforcement through generalization and suppression. *Technical Report SRI-CSL-98-04*, SRI Computer Science Laboratory, 1998.

[27] P. Samarati. Protecting respondents' identities in microdata release. *IEEE Transactions on Knowledge and Data Engineering*, 13(6): pp: 1010-1027. 2001.

[28] X. Sun, H. Wang and J. Li. Injecting purposes and trust into data anonymization. in CIKM 2009.

[29] X. Sun, H. Wang, J. Li and J. Pei. Publishing anonymous survey rating data. Data Min. Knowl. Discov. 23(3): 379-406 (2011)

[30] X. Sun, H. Wang, J. Li and Y. Zhang. Satisfying Privacy Requirements Before Data Anonymization. The Computer Journal. doi:10.1093/comjnl/bxr028. 2011.

[31] L. Sweeney. Weaving technology and policy together to maintain confidentiality. J. of Law, Medicine and Ethics, 25(2ï£¡C3), 1997.

[32] L. Sweeney. $k$-Anonymity: A Model for Protecting Privacy. *International Journal on Uncertainty Fuzziness Knowledge-based Systems*, 10(5), pp 557-570, 2002

[33] T. M. Traian and V. Bindu. Privacy Protection: $p$-sensitive $k$-anonymity Property. International Workshop of Privacy Data Management (PDM2006), In Conjunction with 22th International Conference of Data Engineering (ICDE), Atlanta, 2006

[34] V. S. Verykios, A. K. Elmagarmid, E. Bertino, E. Dasseni and Y. Saygin. Association Rule Hiding. IEEE Transactions on Knowledge and Data Engineering, vol. 16, no. 4, pp. 434-447, April 2004.

[35] I. Witten and E. Frank. Data Mining: Practical machine learning tools and techniques. Morgan Kaufmann, San Francisco, 2nd edition, 2005.

[36] X. Xiao and Y. Tao. Anatomy: simple and effective privacy preservation. VLDB 2006.

[37] Y. Xu, K. Wang, Ada Wai-Chee Fu and Philip S. Yu. Anonymizing Transaction Databases for Publication. KDD 2008.