Chapter 16
Avoiding Attribute Disclosure with the (Extended) *p*-Sensitive *k*-Anonymity Model

Traian Marius Truta and Alina Campan

Abstract Existing privacy regulations together with large amounts of available data created a huge interest in data privacy research. A main research direction is built around the *k*-anonymity property. Several shortcomings of the *k*-anonymity model were addressed by new privacy models such as *p*-sensitive *k*-anonymity, *l*-diversity, (α, k)-anonymity, *t*-closeness. In this chapter we describe two algorithms (*GreedyPKClustering* and *EnhancedPKClustering*) for generating (extended) *p*-sensitive *k*-anonymous microdata. In our experiments, we compare the quality of generated microdata obtained with the mentioned algorithms and with another existing anonymization algorithm (*Incognito*). Also, we present two new branches of *p*-sensitive *k*-anonymity, the constrained *p*-sensitive *k*-anonymity model and the *p*-sensitive *k*-anonymity model for social networks.

16.1 Introduction

The increased availability of individual data combined with today's significant computational power and the tools available to analyze this data, have created major privacy concerns not only for researchers but also for the public [16] and legislators [3]. Privacy has become an important aspect of regulatory compliance, and the ability to automate the privacy enforcement procedures would lead to reduced cost for enterprises. Policies must be developed and modeled to describe how data has to be stored, accessed, manipulated, processed, managed, transferred, and eventually deleted in any organization that stores confidential data. Still, many of these aspects of data management have not been rigorously analyzed from a privacy perspective [15].

Traian Marius Truta · Alina Campan
Department of Computer Science, Northern Kentucky University, Highland Heights, KY 41099, USA, e-mail: trutat1@nku.edu;campana1@nku.edu;ford1@nku.edu

R. Stahlbock et al. (eds.), *Data Mining,* Annals of Information Systems 8,
DOI 10.1007/978-1-4419-1280-0_16,

Data privacy researchers have presented several techniques that aim to avoid the disclosure of confidential information by processing sensitive data before public release ([1, 23], etc.). Among them, the *k*-anonymity model was recently introduced [18, 19]. This model requires that in the *released* (also referred as *masked*) *microdata* (data sets where each tuple belongs to an individual entity, e.g., a person, a company) every tuple will be undistinguishable from at least *k*–1 other tuples with respect to a subset of attributes called *key* or *quasi-identifier* attributes.

Although the model's properties and the techniques used to enforce it on data have been extensively studied ([2, 5, 10, 18, 20], etc.), recent results have shown that *k*-anonymity fails to protect the privacy of individuals in all situations ([13, 21, 24], etc.). New enhanced privacy models have been proposed in the literature to deal with *k*-anonymity's limitations with respect to *sensitive attributes disclosure* [9]. These models include *p*-sensitive *k*-anonymity [22] with its expansion called extended *p*-sensitive *k*-anonymity [6], *l*-diversity [13], (α , *k*)-anonymity [24], *t*-closeness [12], *m*-confidentiality [25], personalized anonymity [26], etc.

In this chapter we describe two algorithms, called *GreedyPKClustering* [7] and *EnhancedPKClustering* [22], that anonymize a microdata set such that its released version will satisfy *p*-sensitive *k*-anonymity. We tailored both algorithms to also generate extended *p*-sensitive *k*-anonymous microdata. We compare the results obtained by our algorithms with the results produced by the *Incognito* algorithm [10], which was adapted to generate *p*-sensitive *k*-anonymous microdata.

Additionally, new branches developed out of the *p*-sensitivity *k*-anonymity model are presented. The first of these two new extensions, called the constrained *p*-sensitive *k*-anonymity model, allows quasi-identifiers generalization boundaries to be specified and *p*-sensitive *k*-anonymity is achieved within the imposed boundaries. This model has the advantage of protecting against identity and attribute disclosure, while controlling the microdata modifications within allowed boundaries. The other new *p*-sensitive *k*-anonymity extension targets the social networks field. A social network can be anonymized to comply with *p*-sensitive *k*-anonymity model, and this model will provide protection against disclosure of confidential information in social network data.

The chapter is structured as follows. Section 16.2 presents the *p*-sensitive *k*-anonymity model, the extended *p*-sensitive *k*-anonymity model, and the anonymization algorithms. Section 16.3 contains an extensive set of experiments. The new branches of *p*-sensitive *k*-anonymity model are defined in Section 16.4. This chapter ends with conclusions and future work directions (Section 16.5).

16.2 Privacy Models and Algorithms

16.2.1 The p-Sensitive k-Anonymity Model and Its Extension

P-sensitive *k*-anonymity is a natural extension of *k*-anonymity that avoids several shortcomings of this model [21]. Next, we present these two models.

Let IM be the initial data set (called initial microdata). IM is described by a set of attributes that are classified into the following three categories:

- $I_1, I_2, \ldots, I_m$ are identifier attributes such as *Name* and *SSN* that can be used to identify a record.
- $K_1, K_2, \ldots, K_q$ are key or quasi-identifier attributes such as *ZipCode* and *Sex* that may be known by an intruder.
- $S_1, S_2, \ldots, S_r$ are confidential or sensitive attributes such as *Diagnosis* and *Income* that are assumed to be unknown to an intruder.

In the released data set (called *masked microdata* and labeled MM) only the quasi-identifier and confidential attributes are preserved; identifier attributes are removed as a prime measure for ensuring data privacy. In order to rigorously and succinctly express the k-anonymity property, we use the following concept.

Definition 16.1 (*QI-Cluster*). Given a microdata, a ***QI-cluster*** consists of all the tuples with identical combination of quasi-identifier attribute values in that microdata.

We define k-anonymity based on the minimum size of all QI-clusters.

Definition 16.2 (*k-Anonymity Property*). The ***k-anonymity property*** for a MM is satisfied if every QI-cluster from MM contains k or more tuples.

Unfortunately, k-anonymity does not provide the amount of confidentiality required for every individual [12, 18, 21]. k-anonymity protects against identity disclosure [8] but fails to protect against attribute disclosure [8] when all tuples of a QI-cluster share the same value for one sensitive attribute [18].

The p-sensitive k-anonymity model considers several sensitive attributes that must be protected against attribute disclosure. It has the advantage of simplicity and allows the data owner to customize the desired protection level by setting various values for p and k.

Definition 16.3 (*p-Sensitive k-Anonymity Property*). A MM satisfies the ***p-sensitive k-anonymity property*** if it satisfies k-anonymity and the number of distinct values for each confidential attribute is at least p within every QI-cluster from MM.

To illustrate this property, we consider the masked microdata from Table 16.1 where *Age* and *ZipCode* are quasi-identifier attributes, and *Diagnosis* and *Income* are confidential attributes.

This masked microdata satisfies the 3-anonymity property with respect to *Age* and *ZipCode*. The first QI-cluster (the first three tuples in Table 16.1) has two different incomes (*60,000* and *40,000*) and only one diagnosis (*AIDS*): therefore, the highest value of p for which p-sensitive 3-anonymity holds is 1. As a result, an intruder who searches information about a young person in his twenties that lives in zip code area 41,099 will discover that the target entity suffers from *AIDS*, even if he does not know which tuple in the first QI-cluster corresponds to that person. This attribute disclosure problem can be avoided if one of the tuples from the first

Table 16.1 Masked microdata example for p-sensitive k-anonymity property

Age	ZipCode	Diagnosis	Income
20	41099	AIDS	60,000
20	41099	AIDS	60,000
20	41099	AIDS	40,000
30	41099	Diabetes	50,000
30	41099	Diabetes	40,000
30	41099	Tuberculosis	50,000
30	41099	Tuberculosis	40,000

QI-cluster would have a value other than *AIDS* for the *Diagnosis* attribute. In this case, both *QI*-clusters would have two different illnesses and two different incomes, and, as a result, the highest value of p would be 2.

P-sensitive k-anonymity cannot be enforced on any given IM, for any p and k. Two necessary conditions to generate a masked microdata with p-sensitive k-anonymity property are presented in [22].

This privacy model has a shortcoming related to the "closeness" of the sensitive attribute values within a *QI*-cluster. To present this situation, we consider the value generalization hierarchy for a sensitive attribute as defined by Sweeney [19]. We use such a hierarchy for the sensitive attribute *Illness* in the following example. We consider that the information that a person has *cancer* (not a leaf value in this case) needs to be protected, regardless of the cancer type she has (*colon cancer*, *prostate cancer*, *breast cancer* are leaf nodes in this generalization hierarchy). If the p-sensitive k-anonymity property is enforced for the released microdata, it is possible that for one *QI*-cluster all of the *Illness* attribute values are to be descendants of the *cancer* node, therefore leading to disclosure. To avoid such situations, the extended p-sensitive k-anonymity model was introduced [6].

We use the notation $\mathcal{H}_S$ to represent the value generalization hierarchy for the sensitive attribute S. We assume that the data owner has the following requirements in order to release a masked microdata:

- All ground (leaf) values in $\mathcal{H}_S$ must be protected against disclosure.
- Some non-ground values in $\mathcal{H}_S$ must be protected against disclosure.
- All the descendants of a protected non-ground value in $\mathcal{H}_S$ must also be protected.

The following definitions allow us to rigorously define the extended p-sensitive k-anonymity property.

Definition 16.4 (*Strong Value*). A protected value in the value generalization hierarchy $\mathcal{H}_S$ of a confidential attribute S is called ***strong*** if none of its ascendants (including the root) is protected.

Definition 16.5 (*Protected Subtree*). We define a ***protected subtree*** of a hierarchy $\mathcal{H}_S$ as a subtree in $\mathcal{H}_S$ that has as root a strong protected value.

Definition 16.6 (*Extended p-Sensitive k-Anonymity Property*). The masked microdata $\mathcal{MM}$ satisfies ***extended p-sensitive k-anonymity property*** if it satisfies k-anonymity and, for each QI-cluster from $\mathcal{MM}$, the values of each confidential attribute S within that group belong to at least p different protected subtrees in $\mathcal{H}_S$.

At a closer look, extended p-sensitive k-anonymity is equivalent to p-sensitive k-anonymity where the confidential attribute values are generalized to their first protected ancestor starting from the hierarchy root (their strong ancestor). Consequently, in order to enforce extended p-sensitive k-anonymity to a data set, the following two-step procedure can be applied:

- Each value of a confidential attribute is generalized (temporarily) to its strong ancestor.
- Any algorithm which can be used for p-sensitive k-anonymization is applied to the modified data set. In the resulted masked microdata the original values of the confidential attributes are restored.

The microdata obtained following these steps satisfy the extended p-sensitive k-anonymity property. Due to this procedure, the algorithms from the next section refer only to p-sensitive k-anonymity. In the experiments related to the extended model, we applied the above-mentioned procedure.

16.2.2 Algorithms for the p-Sensitive k-Anonymity Model

Besides achieving the properties required by the target privacy model (p-sensitive k-anonymity or its extension), anonymization algorithms must also consider minimizing one or more cost measure. We know that optimal k-anonymization is a NP-hard problem [2]. By simple reduction to k-anonymity, it can be easily shown that p-sensitive k-anonymization is also a NP-hard problem. The algorithms we will describe next are good approximations of the optimal solution.

The microdata p-sensitive k-anonymization problem can be formulated as follows.

Definition 16.7 (*p-Sensitive k-Anonymization Problem*). Given a microdata $\mathcal{IM}$, the p-sensitive k-anonymization problem for $\mathcal{MM}$ is to find a partition $\mathcal{S} = \{cl_1, cl_2, \ldots, cl_v\}$ of $\mathcal{IM}$, where $cl_j \in \mathcal{IM}$, $j = 1..v$, are called clusters and: $\bigcup_{j=1}^{v} cl_j = \mathcal{IM}$; $cl_i \cap cl_j = \emptyset$, $i, j = 1..v, i \neq j$; $|cl_j| \geq k$ and cl_j is p-sensitive, $j = 1..v$; and a cost measure is optimized.

Once a solution $\mathcal{S}$ to the above problem is found for a microdata $\mathcal{IM}$, a masked microdata $\mathcal{MM}$ that is p-sensitive k-anonymous is formed by generalizing the quasi-identifier attributes of all tuples inside each cluster of $\mathcal{S}$ to the same values. The generalization method consists in replacing the actual value of an attribute with a less specific, more general value that is faithful to the original [19].

We call *generalization information* for a cluster the minimal covering tuple for that cluster, and we define it as follows.

Definition 16.8 (*Generalization Information*). Let $cl = \{r_1, r_2, \ldots, r_q\} \in \mathcal{S}$ be a cluster, $KN = \{N_1, N_2, \ldots, N_s\}$ be the set of numerical quasi-identifier attributes and $KC = \{C_1, C_2, \ldots, C_t\}$ be the set of categorical quasi-identifier attributes. The ***generalization information of cl***, w.r.t. quasi-identifier attribute set $\mathcal{K} = KN \cup KC$ is the "tuple" $gen(cl)$, having the scheme $\mathcal{K}$, where

- For each categorical attribute $C_j \in \mathcal{K}$, $gen(cl)[C_j]$ = the lowest common ancestor in $\mathcal{H}_{C_j}$ of $\{r_1[C_j], \ldots, r_q[C_j]\}$, where $\mathcal{H}_C$ denotes the hierarchies (domain and value) associated to the categorical quasi-identifier attribute C.
- For each numerical attribute $N_j \in \mathcal{K}$, $gen(cl)[N_j]$ = the interval $[min\{r_1[N_j], \ldots, r_q[N_j]\}, max\{r_1[N_j], \ldots, r_q[N_j]\}]$.

For a cluster cl, its generalization information $gen(cl)$ is the tuple having as value for each quasi-identifier attribute the most specific common generalized value for all the attribute values from cl. In $\mathcal{MM}$, each tuple (its quasi-identifier part) from the cluster cl will be replaced by $gen(cl)$, and thus forming a *QI*-cluster.

There are several possible cost measures that can be used as optimization criterion for the p-sensitive k-anonymization problem ([4, 5], etc.). A simple cost measure is based on the size of each cluster from $\mathcal{S}$. This measure, called *discernibility metric* (*DM*) [4], assigns to each record x from $\mathcal{IM}$ a penalty that is determined by the size of the cluster containing x:

$$DM(\mathcal{S}) = \sum_{j=1}^{v} |cl_j|^2 \tag{16.1}$$

LeFevre introduced the alternative measure called *normalized average cluster size metric* (*AVG*) [11]:

$$AVG(\mathcal{S}) = \frac{n}{v \cdot k} \tag{16.2}$$

where n is the size of the $\mathcal{IM}$, v is the number of clusters, and k is as in k-anonymity. It is easy to notice that the *AVG* cost measure is inversely proportional to the number of clusters, and minimizing *AVG* is equivalent to maximizing the total number of clusters.

Another cost measure described in the literature is the *information loss* (*IL*) caused by generalizing each cluster to a common tuple [5].

While k-anonymity is satisfied for each individual cluster when its size is k or more, the p-sensitive property is not so obvious to achieve. For this, two diversity measures that quantify, with respect to sensitive attributes, the *diversity between a tuple and a cluster* and the *homogeneity of a cluster* were introduced [22].

The *GreedyPKClustering* algorithm is briefly described below. A complete presentation including a pseudocode-like algorithm can be found in [7].

The *QI*-clusters are formed one at a time. For forming one *QI*-cluster, a tuple in $\mathcal{IM}$ not yet allocated to any cluster is selected as a seed for the new cluster. Then the algorithm gathers tuples to this currently processed cluster until it satisfies both requirements of the p-sensitive k-anonymity model. At each step, the current cluster grows with one tuple. This tuple is selected, of course, from the tuples not

yet allocated to any cluster. If the p-sensitive part is not yet satisfied for the current cluster, then the chosen tuple is the one most probable to enrich the diversity of the current cluster with regard to the confidential attribute values. This selection is made by the diversity measure between a tuple and a cluster. If the p-sensitive part is already satisfied for every confidential attribute, then the least different or diverse tuple (w.r.t. the confidential attributes) of the current cluster is chosen. This selection is justified by the need to spare other different confidential values, not present in the current cluster, in order to be able to form as many as possible new p-sensitive clusters. When a tie happens, i.e., multiple candidate tuples exist conforming to the previous selection criteria, then the tuple that minimizes the cluster's IL growth will be preferred.

It is possible that the last constructed cluster will contain less than k tuples or it will not satisfy the p-sensitivity requirement. In that case, this cluster needs to be dispersed between the previously constructed groups. Each of its tuples will be added to the cluster whose IL will minimally increase by that tuple addition. At the end, a solution for p-sensitive k-anonymity problem is found.

The *EnhancedPKClustering* algorithm is an alternative solution for the p-sensitive k-anonymization problem. It considers AVG (or the partition cardinality) that has to be maximized as the cost measure. Its complete presentation can be found in [22].

This algorithm starts by enforcing the p-sensitive part using the properties proved for the p-sensitive k-anonymity model [22]. The tuples from $\mathcal{IM}$ are distributed to form p-sensitive clusters with respect to the sensitive attributes. After p-sensitivity is achieved, the clusters are further processed to satisfy k-anonymity requirement as well. A more detailed description of how the algorithm proceeds follows.

In the beginning, the algorithm determines the p-sensitive equivalence classes [22], orders the attributes based on the harder to make sensitive relation [22], and computes the value *iValue* that divides the p-sensitive equivalence classes into two categories: one with less frequent values for the hardest to anonymize attribute and one with more frequent values. Now, the QI-clusters are created using the following steps:

- First, the tuples in the least frequent category of p-sensitive equivalence classes are divided into *maxClusters* clusters (maximum possible number of clusters can be computed in advance based on frequency distributions of sensitive attributes [21]) such that each cluster will have *iValue* tuples with unique values within each cluster for the harder to make sensitive attribute [22].
- Second, the remaining p-sensitive equivalence classes are used to fill the clusters such that each of them will have exactly p tuples with p distinct values for S_1.
- Third, the tuples not yet assigned to any cluster are used to add diversity for all remaining sensitive attributes until all clusters are p-sensitive. If no tuples are available, some of the less diverse (more homogenous) clusters are removed and their tuples are reused for the remaining clusters. At the end of this step all clusters are p-sensitive.

- Fourth, the tuples not yet assigned to any cluster are used to increase the size of each cluster to k. If no tuples are available, some of the less populated clusters are removed and their tuples are reused for the remaining clusters. At the end of this step all clusters are p-sensitive k-anonymous.

Along all the steps, when a choice is to be made, one or more optimization criteria are used (diversity between a tuple and a cluster, and increase in information loss).

While both of these algorithms achieve the p-sensitive k-anonymous data sets, their approach is different. *GreedyPKClustering* is an extension of the *greedy_k_member_clustering* [6], a clustering algorithm used for k-anonymity, and *Enhanced-PKClustering* is a novel algorithm that takes advantage of the p-sensitive k-anonymity model properties and does not have an equivalent for k-anonymity only.

16.3 Experimental Results

16.3.1 Experiments for p-Sensitive k-Anonymity

In this section we compare the performance of *EnhancedPKClustering*, *Greedy-PKClustering*, and an adapted version of *Incognito* [10].

The first two algorithms are explained in the previous section, and *Incognito* is the first efficient algorithm that generates a k-anonymous data set. This algorithm finds a full-domain generalization that is k-anonymous by creating a multi-domain generalization lattice for the domains of the quasi-identifiers attributes. Starting with the least general domain at the root of the lattice, the algorithm performs a breadth-first search, checking whether each generalization encountered satisfies k-anonymity. This algorithm can be used to find a single (weighted) minimal generalization, or it can be used to find the set of all k-anonymous minimal domain generalizations [10]. We easily adapted this algorithm by testing for p-sensitive k-anonymity (instead of k-anonymity) at every node in the generalization lattice.

All three algorithms have been implemented in Java, and tests were executed on a dual CPU machine running Windows 2003 Server with 3.00 GHz and 1 GB of RAM.

A set of experiments has been conducted for an IM consisting of 10,000 tuples randomly selected from the *Adult* data set [17]. In all the experiments, we considered *age*, *work-class*, *marital-status*, *race*, *sex*, and *native-country* as the set of quasi-identifier attributes; and *education-num*, *education*, and *occupation* as the set of confidential attributes. Among the quasi-identifier attributes, *age* was numerical, and the other five attributes were categorical. The value generalization hierarchies of the quasi-identifier categorical attributes were as follows: for *work-class*, *race*, and *sex* two-level hierarchies (i.e., ground level and root level); for *marital-status* a three-level hierarchy; and for *native-country* a four-level hierarchy. The value hierarchy for the *native-country* quasi-identifier attribute, the most complex among the hierarchies for all our quasi-identifiers, is depicted in Fig. 16.1.

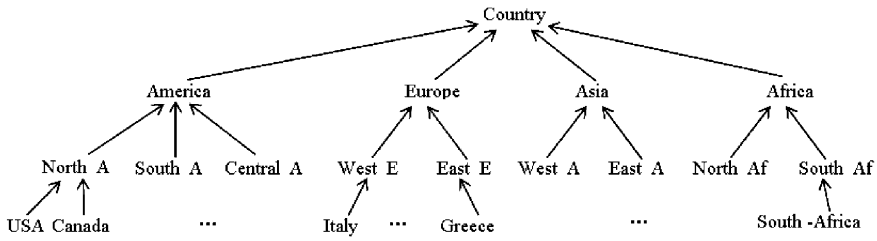


Fig. 16.1 The value hierarchy for the quasi-identifier categorical attribute *Country*

P-sensitive *k*-anonymity was enforced with respect to all six quasi-identifier attributes and all three confidential attributes. Figures 16.2 and 16.3 show comparatively the *AVG* and *DM* values of the three algorithms, *EnhancedPKClustering*, *GreedyPKClustering*, and *Incognito*, produced for $p = 3$, respectively, $p = 10$, and different k values. As expected, the results for the first two algorithms clearly outperform *Incognito* results in all cases. We also notice that *EnhancedPKClustering*

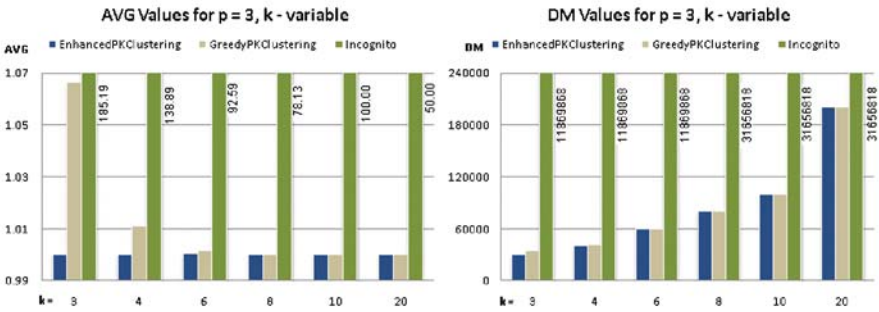


Fig. 16.2 *AVG* and *DM* for *EnhancedPKClustering*, *GreedyPKClustering*, and *Incognito*, p=3 and k variable

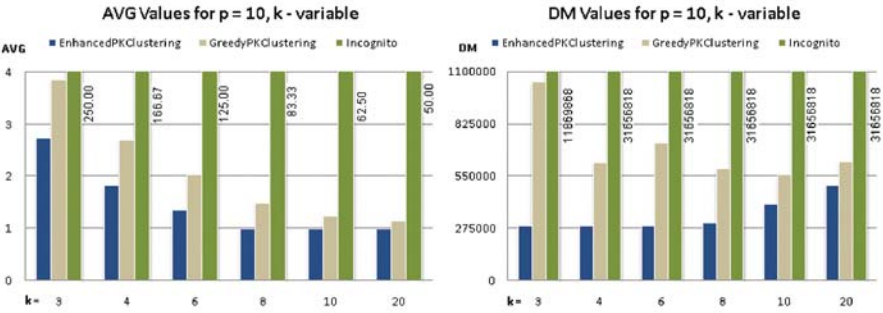


Fig. 16.3 *AVG* and *DM* for *EnhancedPKClustering*, *GreedyPKClustering*, and *Incognito*, p=10 and k variable

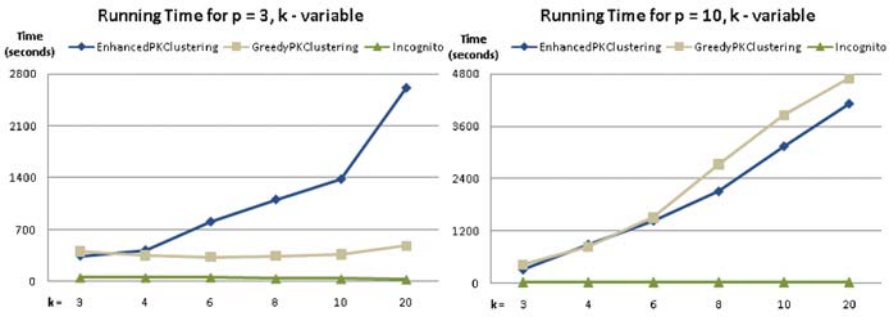


Fig. 16.4 Running time for *EnhancedPKClustering*, *GreedyPKClustering*, and *Incognito* algorithms

is able to improve the performances of the *GreedyPKClustering* algorithm in cases where solving the p-sensitivity part takes prevalence over creating clusters of size k.

Figure 16.4 shows the time required to generate the masked microdata by all three algorithms, for $p = 3$, respectively, $p = 10$, and different k values. Since *Incognito* uses global recording and our domain generalization hierarchies for this data set have a low height, its running time is very fast. The *GreedyPKClustering* is faster than the new algorithm for small values of p, but when it is more difficult to create p-sensitivity within each cluster the *EnhancedPKClustering* has a slight advantage.

Based on these results, it is worth noting that a combination of *GreedyPKClustering* (for low values of p, in our case 3) and *EnhancedPKClustering* (for high values of p, in our experiment 10) would be desirable in order to improve both running time and the selected cost measure (AVG or DM).

16.3.2 Experiments for Extended p-Sensitive k-Anonymity

The *EnhancedPKClustering* and *GreedyPKClustering* algorithms can easily be adapted to generate extended p-sensitive k-anonymous microdata. In order to do so, the algorithms are applied to a modified IM in which the sensitive attributes are replaced with their strong ancestors. In the resulting MM the sensitive attributes are restored to their original values.

In this section we compare the performance of *EnhancedPKClustering* and *GreedyPKClustering* algorithms for the extended p-sensitive k-anonymity model. A set of experiments were conducted for the same IM as in the previous section. We also reused the generalization hierarchies of all six quasi-identifier categorical attributes. All three confidential attributes were considered categorical, and their value hierarchies and strong values are depicted in Fig. 16.5 – strong values are bolded and delimited by * characters. We make an observation with regard to the *education* sensitive attribute's hierarchy. This hierarchy is not balanced, but this has

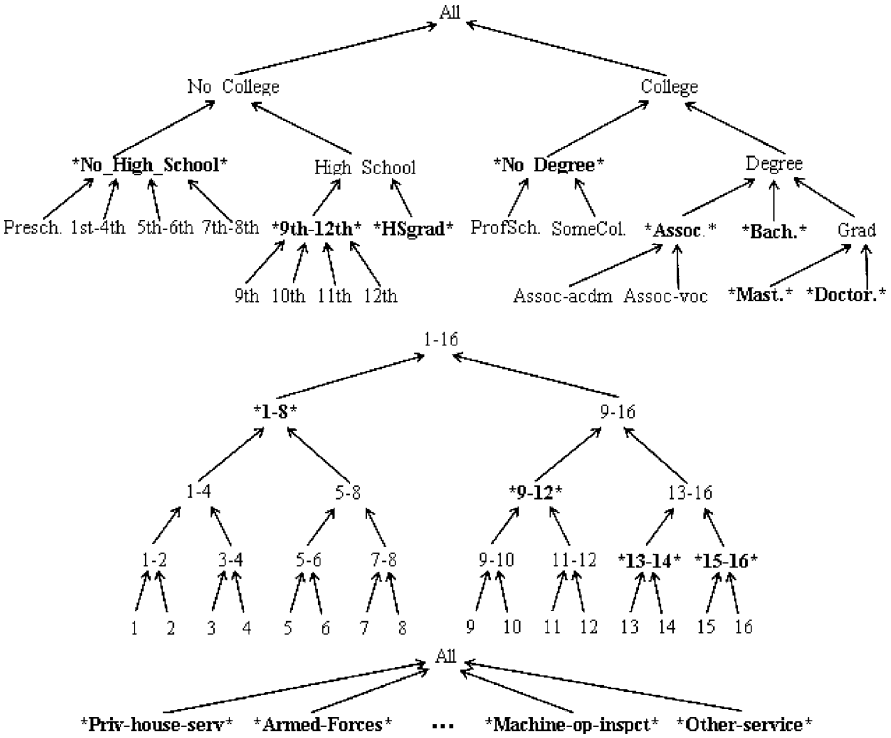


Fig. 16.5 The value hierarchies and strong values for the sensitive categorical attributes *education*, *education-num*, and *occupation*

no influence on the algorithm's performance or results' quality, as long as no cost measures are computed w.r.t. generalization performed according to this hierarchy; its only role is to give guidance about the sensitivity of the values of the confidential attribute *education*.

Another set of experiments used synthetic data sets, where the quasi-identifier and the sensitive attribute values were generated to follow some predefined distributions. For our experiments, we generated four microdata sets using normal and uniform distributions. All four data sets have identical schema *(QI_N; QI_C1; QI_C2; QI_C3; S_C1; S_C2)* where the first attribute *(QI_N)* is a numerical quasi-identifier (*age* like), the next three *(QI_C1; QI_C2; QI_C3)* are categorical quasi-identifiers and the last two *(S_C1* and *S_C2)* are categorical sensitive attributes. The distributions followed by each attribute for the four data sets are illustrated in Table 16.2.

Figure 16.6 depicts the common value generalization hierarchy for the categorical quasi-identifiers of the synthetic data sets. Figure 16.7 shows the value generalization hierarchies and the strong values for the sensitive attributes of the synthetic data sets.

For the numerical attribute we use *age*-like values 0, 1, ..., 99. To generate a uniform distribution for this range we use the mean 99/2 and standard deviation 99/6.

Table 16.2 Data distribution in the synthetic data sets

	All *QI* attributes	All sensitive attributes
Dataset_UU	Uniform	Uniform
Dataset_UN	Uniform	Normal
Dataset_NU	Normal	Uniform
Dataset_NN	Normal	Normal

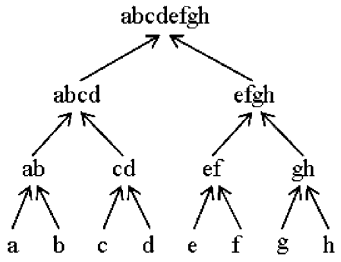
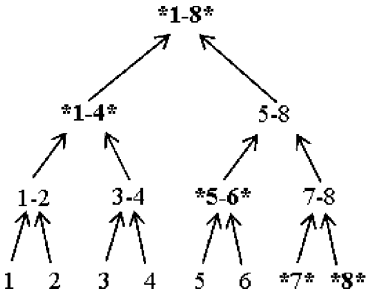


Fig. 16.6 The value generalization hierarchy of the categorical attributes of the synthetic data sets



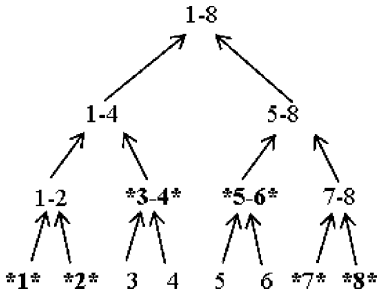


Fig. 16.7 The value generalization hierarchies and strong values for the sensitive attributes of the synthetic data sets: *S_C*1 and *S_C*2

For each categorical attribute we use eight values that are grouped in a hierarchy as shown in Fig. 16.6. To generate a uniform-like distribution for the categorical attributes we use the range 0–8 with mean 8/2 and standard deviation 8/6 and the mapping shown in Table 16.3 (val is the value computed by the generator).

Table 16.3 Mapping between 0 and 8 range and discrete values

$val < 1$	$1 \leq val < 2$	$2 \leq val < 3$	...	$6 \leq val < 7$	$val \leq 8$
a	b	c	...	g	h

Next, for each of the five experimental data sets used, we present the *AVG*, *DM*, and some of the execution time cost measure values for each of the two algorithms, *EnhancedPKClustering* and *GreedyPKClustering* for $p = 3$ and different k values (Figs. 16.8, 16.9, 16.10, 16.11, 16.12, and 16.13).

The *AVG* and *DM* results are very similar. We notice that when the *p*-sensitive part is difficult to achieve, the *EnhancedPKClustering* algorithm performs better. These results are similar with the ones obtained for *p*-sensitive *k*-anonymity property.

The following observations are true for both *p*-sensitive *k*-anonymity and its extension. The *GreedyPKClustering* is faster than the *EnhancedPKClustering* algorithm for large values of k, but when it is more difficult to create *p*-sensitivity within each cluster the *EnhancedPKClustering* has a slight advantage. We also notice that the running time of *GreedyPKClustering* algorithm is influenced by the sensitive attributes' distribution.

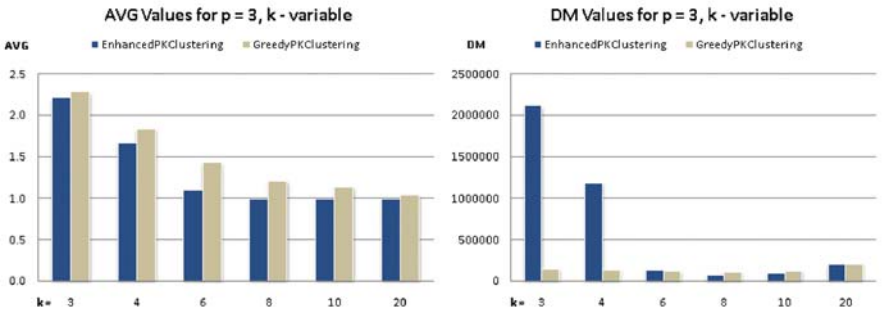


Fig. 16.8 *AVG*, *DM* for *EnhancedPKClustering* and *GreedyPKClustering*, Adult Dataset

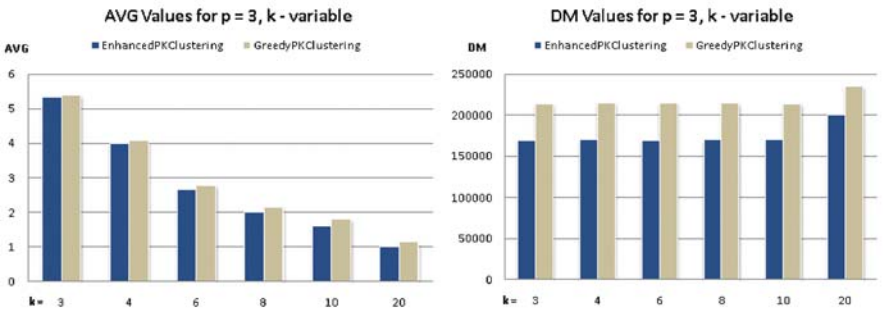


Fig. 16.9 *AVG*, *DM* for *EnhancedPKClustering* and *GreedyPKClustering*, Dataset_NN

16.4 New Enhanced Models Based on *p*-Sensitive *k*-Anonymity

16.4.1 Constrained p-Sensitive k-Anonymity

In general, the existing anonymization algorithms use different quasi-identifier's generalization/tuple suppression strategies in order to obtain a masked microdata that is *k*-anonymous (or satisfies an extension of *k*-anonymity) and conserves as much information intrinsic to the initial microdata as possible. To our knowledge, none of these models limits the amount of generalization that is permitted to be performed for specific quasi-identifier attributes. The ability to limit the amount of allowed generalization could be valuable, and, in fact, indispensable for real-life data sets and applications. For example, for some specific data analysis tasks, available masked microdata with the address information generalized beyond the US state level could be useless. Our approach consists of specifying quasi-identifiers generalization boundaries and achieving *p*-sensitive *k*-anonymity within the imposed boundaries. Using this approach we recently introduced a similar model for *k*-anonymity only, entitled constrained *k*-anonymity. In this subsection we present

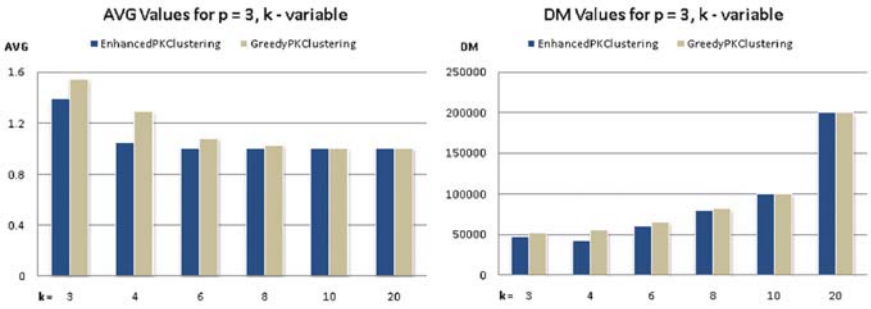


Fig. 16.10 *AVG*, *DM* for *EnhancedPKClustering* and *GreedyPKClustering*, Dataset_NU

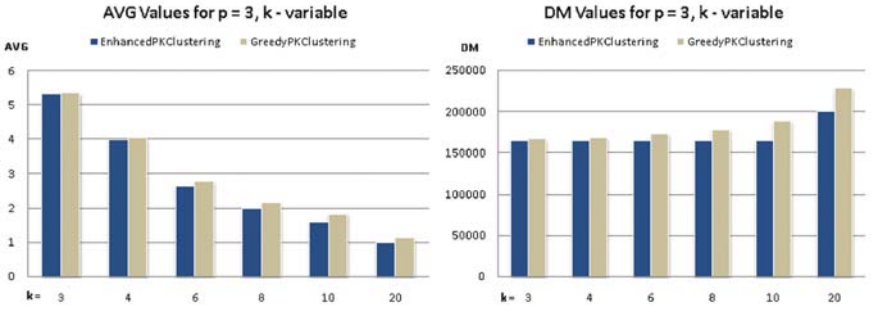


Fig. 16.11 *AVG*, *DM* for *EnhancedPKClustering* and *GreedyPKClustering*, Dataset_UN

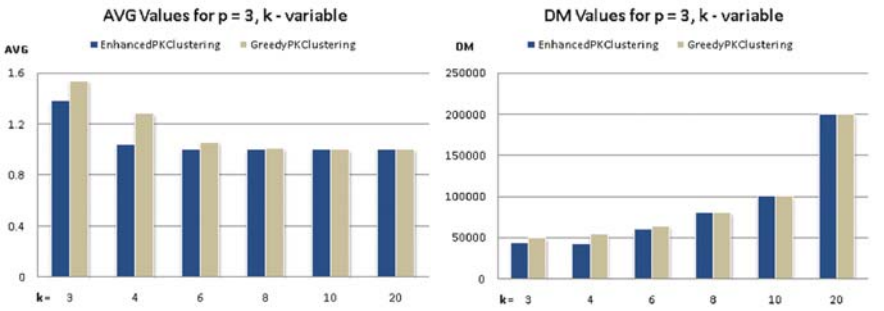


Fig. 16.12 *AVG*, *DM* for *EnhancedPKClustering* and *GreedyPKClustering*, Dataset_UU

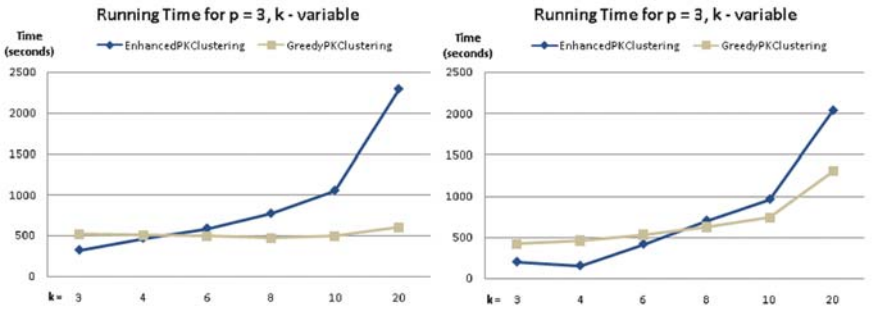


Fig. 16.13 Running time for *EnhancedPKClustering* and *GreedyPKClustering*, Adult, and Dataset_NU

the constrained *p*-sensitive *k*-anonymity privacy model. A complete presentation of constrained *k*-anonymity can be found in [14].

In order to specify a generalization boundary, we introduced the concept of a maximal allowed generalization value that is associated with each quasi-identifier attribute value. This concept is used to express how far the owner of the data thinks that the quasi-identifier's values could be generalized, such that the resulted masked microdata would still be useful. Limiting the amount of generalization for quasi-identifier attribute values is a necessity for various uses of the data. The data owner is often aware of the way various researchers are using the data and, as a consequence, he/she is able to identify maximal allowed generalization values. For instance, when the released microdata is used to compute various statistical measures related to the US states, the data owner will select the states as maximal allowed generalization values.

Definition 16.9 (*Maximal Allowed Generalization Value*). Let Q be a quasi-identifier attribute (categorical or numerical), and $\mathcal{H}_Q$ its predefined value generalization hierarchy. For every leaf value $v \in \mathcal{H}_Q$, the ***maximal allowed generalization value*** of v, $MAGVal(v)$, is the value (leaf or not-leaf) in $\mathcal{H}_Q$ situated on the path from v to the root, such that

- for any released microdata, the value v is permitted to be generalized only up to $MAGVal(v)$ and
- when several *MAGVal*s exist on the path between v and the hierarchy root, then the $MAGVal(v)$ is the first *MAGVal* that is reached when following the path from v to the root node.

Figure 16.14 contains an example of defining *MAGVal*s for a subset of values for the *Location* attribute.

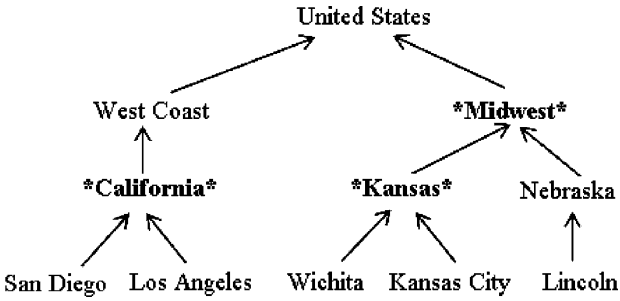


Fig. 16.14 Examples of maximal allowed generalization values

The *MAGVal*s for the leaf values "San Diego" and "Lincoln" are "California," and, respectively, "Midwest" (the maximal allowed generalization values are bolded and marked by * characters that delimit them). This means that the quasi-identifier *Location*'s value "San Diego" may be generalized to itself or "California," but not to "West Coast" or the "United States." Also, "Lincoln" may be generalized to itself, "Nebraska," or "Midwest," but it may not be generalized to the "United States."

Usually, the data owner has generalization restrictions for most of the quasi-identifiers. If for a particular quasi-identifier attribute Q there are not any restrictions with respect to its generalization, then the $\mathcal{H}_Q$'s root value will be considered the maximal allowed generalization value for all the leaf values.

Definition 16.10 (*Constraint Violation*). We say that the masked microdata $\mathcal{MM}$ has a constraint violation if one quasi-identifier value, v, in $\mathcal{IM}$, is generalized in one tuple in $\mathcal{MM}$ beyond its specific maximal generalization value, $MAGVal(v)$.

Definition 16.11 (*Constrained p-Sensitive k-Anonymity*). The masked microdata $\mathcal{MM}$ satisfies the constrained p-sensitive k-anonymity property if it satisfies p-sensitive k-anonymity and it does not have any constraint violation.

We illustrate the above concept with the following example. The initial microdata set $\mathcal{IM}$ in Table 16.4 is characterized by the following attributes: *Name* and *SSN* are identifier attributes (removed from the $\mathcal{MM}$), *Age* and *Location* are the quasi-identifier attributes, and *Diagnosis* is the sensitive attribute. The attribute *Location* values and their *MAGVal*s are described in Fig. 16.14. *Age* does not have any generalization boundary requirements. This microdata set has to be masked such that the

Table 16.4 An initial microdata set IM

Record	Name	SSN	Age	Location	Diagnosis
r_1	Alice	123456789	20	San Diego	AIDS
r_2	Bob	323232323	40	Los Angeles	Asthma
r_3	Charley	232345656	20	Wichita	Asthma
r_4	Dave	333333333	40	Kansas City	Tuberculosis
r_5	Eva	666666666	40	Wichita	Asthma
r_6	John	214365879	20	Kansas City	Asthma

corresponding masked microdata will satisfy constrained *p*-sensitivity *k*-anonymity, where the user wants that the *Location* attribute values not to be generalized in the masked microdata further than the specified maximal allowed generalization values shown in Fig. 16.14.

Tables 16.5 and 16.6 illustrate two possible masked microdata MM_1 and MM_2 for the initial microdata IM. The first one, MM_1, satisfies 2-sensitive 2-anonymity (it is actually 2-sensitive 3-anonymous), but contradicts constrained 2-sensitive 2-anonymity w.r.t. *Location* attribute's maximal allowed generalization. On the other hand, the second microdata set, MM_2, satisfies constrained 2-sensitive 2-anonymity: every *QI*-cluster consists of at least two tuples, there are two distinct values for the sensitive attribute in each cluster, and none of the *Location* initial attribute's values are generalized beyond its *MAGVal*.

Table 16.5 A masked microdata set MM_1 for the initial microdata IM

Record	Age	Location	Diagnosis
r_1	20	United States	AIDS
r_3	20	United States	Asthma
r_6	20	United States	Asthma
r_2	40	United States	Asthma
r_4	40	United States	Tuberculosis
r_5	40	United States	Asthma

Table 16.6 A masked microdata set MM_2 for the initial microdata IM

Record	Age	Location	Diagnosis
r_1	20–40	California	AIDS
r_2	20–40	California	Asthma
r_3	20–40	Kansas	Asthma
r_4	20–40	Kansas	Tuberculosis
r_5	20–40	Kansas	Asthma
r_6	20–40	Kansas	Asthma

16.4.2 p-Sensitive k-Anonymity in Social Networks

The advent of social networks in the last few years has accelerated the research in this field. Online social interaction has become very popular around the globe and most sociologists agree that this trend will not fade away. Privacy in social networks is still in its infancy, and practical approaches are yet to be developed. *K*-anonymity model has been recently extended to social networks [8, 27] by requiring that every node (individual) in the social network to be undistinguishable from other (*k*–1) nodes. While this seems similar with the microdata case, the requirement of indistinguishability includes the similar network (graph) structure.

We consider the social network modeled as a simple undirected graph $\mathcal{G} = (\mathcal{N}, \mathcal{E})$, where $\mathcal{N}$ is the set of nodes and $\mathcal{E} \subseteq \mathcal{N} \times \mathcal{N}$ is the set of edges. Each node represents an individual entity. Each edge represents a relationship between two entities.

The set of nodes, $\mathcal{N}$, is described by a set of attributes that are classified into identifier, quasi-identifier, and confidential categories. If we exclude the relationship between nodes, the social network data resembles a microdata set.

We allow only binary relationships in our model. Moreover, we consider all relationships as being of the same type and, as a result, we represent them via unlabeled undirected edges. We consider also this type of relationships to be of the same nature as all the other "traditional" quasi-identifier attributes. We will refer to this type of relationship as the *quasi-identifier relationship*. In other words, the graph structure may be known to an intruder and used by matching it with known external structural information, therefore serving in privacy attacks that might lead to identity and/or attribute disclosure.

To create a *p*-sensitive *k*-anonymous social network we reuse the generalization technique for quasi-identifier attributes. For the quasi-identifier relationship we use the generalization approach employed in [26] which consists of collapsing clusters together with their component nodes' structure.

Given a partition of nodes for a social network $\mathcal{G}$, we are able to create an anonymized graph by using generalization information and quasi-identifier relationship generalization (for more details about this generalization see [8].

Definition 16.12 (*Masked Social Network*). Given an initial social network, modeled as a graph $\mathcal{G} = (\mathcal{N}, \mathcal{E})$, and a partition $\mathcal{S} = \{cl_1, cl_2, \ldots, cl_v\}$ of the nodes set $\mathcal{N}$, $\bigcup_{j=1}^{v} cl_j = \mathcal{N}$; $cl_i \cap cl_j = \emptyset$, $i, j = 1..v$, $i \neq j$; the corresponding ***masked social network*** $\mathcal{M}\mathcal{G}$ is defined as $\mathcal{M}\mathcal{G} = (\mathcal{M}\mathcal{N}, \mathcal{M}\mathcal{E})$, where

- $\mathcal{M}\mathcal{N} = \{Cl_1, Cl_2, \ldots, Cl_v\}$, Cl_i is a node corresponding to the cluster $cl_j \in \mathcal{S}$ and is described by the "tuple" $gen(cl_j)$ (the generalization information of cl_j, w.r.t. quasi-identifier attribute set) and the intra-cluster generalization pair $(|cl_j|, |E_{cl_j}|)$ ($|cl|$ – the number of nodes in the cluster cl; $|E_{cl}|$ – the number of edges between nodes from cl);

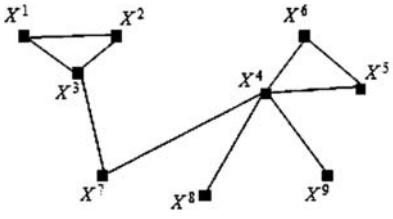
- $\mathcal{ME} \subseteq \mathcal{MN} \times \mathcal{MN}$; $(Cl_i, Cl_j) \in \mathcal{ME}$ iif $Cl_i, Cl_j \in \mathcal{MN}$ and $\exists X \in cl_j, Y \in cl_j$, such that $(X,Y) \in \mathcal{E}$. Each generalized edge $(Cl_i, Cl_j) \in \mathcal{ME}$ is labeled with the inter-cluster generalization value $|\mathcal{E}_{cl_i,cl_j}|$ (the number of edges between nodes from cl_i and cl_j).

By construction, all nodes from a cluster cl collapsed into the generalized (masked) node Cl are undistinguishable from each other.

In order to have p-sensitive k-anonymity property for a masked social network, we need to add two extra conditions to Definition 16.12, first that each cluster from the initial partition is of size at least k and second that each cluster has at least p distinct values for each sensitive attribute. The formal definition of a masked social network that is p-sensitive k-anonymous is presented below.

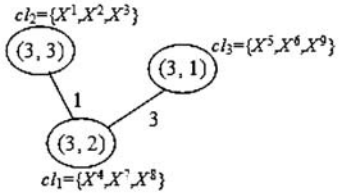
Definition 16.13 (*p-Sensitive k-Anonymous Masked Social Network*). A masked social network $\mathcal{MG} = (\mathcal{MN}, \mathcal{ME})$, where $\mathcal{MN} = \{Cl_1, Cl_2, \ldots, Cl_v\}$, and $Cl_j = [gen(cl_j), (|cl_j|, |E_{cl_j}|)]$, $j = 1, \ldots, v$ is p-sensitive k-anonymous if and only if $|cl_j| \geq k$ for all $j = 1, \ldots, v$ (the k-anonymity requirement) and for each sensitive attribute S and for each $Cl \in \mathcal{MN}$, the number of distinct values for S in Cl is greater than or equal to p (the p-sensitive requirement).

We illustrate the above concept with the following example. We consider a social network $\mathcal{G}$ as depicted in Fig. 16.15. The quasi-identifier attributes are *ZipCode* and *Gender*, and the sensitive attribute is *Disease*. This social network can be anonymized to comply with 2-sensitive 3-anonymity, and one possible masked social network that corresponds to $\mathcal{G}$ is depicted in Fig. 16.16.



Node	*ZipCode*	*Gender*	*Disease*
X^1	41076	male	Diabetes
X^2	41075	male	Asthma
X^3	41076	male	Asthma
X^4	41099	male	Diabetes
X^5	48201	female	Diabetes
X^6	41075	female	Diabetes
X^7	41099	male	Asthma
X^8	41099	male	Asthma
X^9	41075	female	Asthma

Fig. 16.15 A social network G, its structural information, and its node's attribute values



Node	*ZipCode*	*Gender*	*Disease*
X^1	4107*	male	Diabetes
X^2	4107*	male	Asthma
X^3	4107*	male	Asthma
X^4	41099	male	Diabetes
X^7	41099	male	Asthma
X^8	41099	male	Asthma
X^5	4****	female	Diabetes
X^6	4****	female	Diabetes
X^9	4****	female	Asthma

Fig. 16.16 A masked social network MG, its structural information, and its node's attribute values

16.5 Conclusions and Future Work

Our extensive experiments showed that both *GreedyPKClustering* and *EnhancedPKClustering* produce quality masked microdata that satisfy (extended) p-sensitive k-anonymity and outperform anonymization algorithms based on global recoding.

The new privacy models are a promising avenue for future research; we currently work on developing efficient algorithms for constrained p-sensitive k-anonymity and p-sensitive k-anonymity for social networks models. We expect both *EnhancedPKClustering* and *GreedyPKClustering* to be adjustable for achieving data anonymization in agreement with both these new models.

Another research direction is to adapt the *EnhancedPKClustering* and *GreedyPKClustering* for enforcing similar privacy requirements such as (α, k)-anonymity, l-diversity.

References

1. N.R. Adam and J.C. Wortmann, Security Control Methods for Statistical Databases: A Comparative Study, *ACM Computing Surveys* 21(4) (1989), pp. 515–556.
2. G. Aggarwal, T. Feder, K. Kenthapadi, R. Motwani, R. Panigrahy, D. Thomas, and A. Zhu, Anonymizing Tables, in: *Proceedings of the International Conference on Database Theory*, 2005, pp. 246–258.
3. R. Agrawal, J. Kiernan, R. Srikant, R. and Y. Xu. Hippocratic Databases, in: *Proceedings of the Very Large Data Base Conference*, 2002, pp. 143–154.
4. R.J. Bayardo and R. Agrawal, Data Privacy through Optimal k-Anonymization, in: *Proceedings of the IEEE International Conference on Data Engineering*, 2005, pp. 217–228.
5. J.W. Byun, A. Kamra, E. Bertino and N. Li, Efficient k-Anonymity using Clustering Techniques, in: *Proceedings of Database Systems for Advanced Applications*, 2006, pp. 188–200.
6. A. Campan and T.M. Truta, Extended P-Sensitive K-Anonymity, *Studia Universitatis Babes-Bolyai Informatica* 51(2) (2006), pp. 19–30.
7. A. Campan, T.M. Truta, J. Miller and R.A. Sinca, Clustering Approach for Achieving Data Privacy, in: *Proceedings of the International Data Mining Conference*, 2007, pp. 321–327.
8. A. Campan and T.M. Truta, A Clustering Approach for Data and Structural Anonymity in Social Networks, in: *Proceedings of the Privacy, Security, and Trust in KDD Workshop*, 2008.
9. D. Lambert, Measures of Disclosure Risk and Harm, *Journal of Official Statistics* 9 (1993), pp. 313–331.
10. K. LeFevre, D. DeWitt and R. Ramakrishnan, Incognito: Efficient Full-Domain K-Anonymity, in: *Proceedings of the ACM SIGMOD*, 2005, pp. 49–60.
11. K. LeFevre, D. DeWitt and R. Ramakrishnan, Mondrian Multidimensional K-Anonymity, in: *Proceedings of the IEEE International Conference on Data Engineering*, 2006, 25.
12. N. Li, T. Li and S. Venkatasubramanian, T-Closeness: Privacy Beyond k-Anonymity and l-Diversity, in: *Proceedings of the IEEE International Conference on Data Engineering*, 2007, pp. 106–115.
13. A. Machanavajjhala, J. Gehrke and D. Kifer, L-Diversity: Privacy beyond K-Anonymity, in: *Proceedings of the IEEE International Conference on Data Engineering*, 2006, 24.
14. J. Miller, A. Campan and T.M. Truta, Constrained K-Anonymity: Privacy with Generalization Boundaries, in: *Proceedings of the Practical Preserving Data Mining Workshop*, 2008.
15. M.C. Mont, S. Pearson and R. Thyne, A Systematic Approach to Privacy Enforcement and Policy Compliance Checking in Enterprises, in: *Proceedings of the Trust and Privacy in Digital Business Conference*, 2006, pp. 91–102.

16. MSNBC, Privacy Lost, 2006, *Available online at http://www.msnbc.msn.com/id/15157222*.
17. D.J. Newman, S. Hettich, C.L. Blake and C.J. Merz, UCI Repository of Machine Learning Databases, UC Irvine, 1998, *Available online at www.ics.uci.edu/ mlearn/MLRepository.html*.
18. P. Samarati, Protecting Respondents Identities in Microdata Release, *IEEE Transactions on Knowledge and Data Engineering* 13(6) (2001), pp. 1010–1027.
19. L. Sweeney, k-Anonymity: A Model for Protecting Privacy, *International Journal on Uncertainty, Fuzziness, and Knowledge-based Systems* 10(5) (2002), pp. 557–570.
20. L. Sweeney, Achieving k-Anonymity Privacy Protection Using Generalization and Suppression, *International Journal on Uncertainty, Fuzziness, and Knowledge-based Systems* 10(5) (2002), pp. 571–588.
21. T.M. Truta and V. Bindu, Privacy Protection: P-Sensitive K-Anonymity Property, in: *Proceedings of the ICDE Workshop on Privacy Data Management*, 2006, 94.
22. T.M. Truta, A. Campan and P. Meyer, Generating Microdata with P-Sensitive K-Anonymity Property, in: *Proceedings of the VLDB Workshop on Secure data Management*, 2007, pp. 124–141.
23. L. Willemborg and T. Waal (ed), Elements of Statistical Disclosure Control, Springer Verlag, New York, 2001.
24. R.C.W. Wong, J. Li, A.W.C. Fu and K. Wang, (α, k)-Anonymity: An Enhanced k-Anonymity Model for Privacy-Preserving Data Publishing, in: *Proceedings of the ACM International Conference on Knowledge Discovery and Data Mining*, 2006, pp. 754–759.
25. R.C.W. Wong, J. Li, A.W.C. Fu and J. Pei, Minimality Attack in Privacy-Preserving Data Publishing, in: *Proceedings of the Very Large Data Base Conference*, 2007, pp. 543–554.
26. X. Xiao and Y. Tao, Personalized Privacy Preservation, in: *Proceedings of the ACM SIGMOD*, 2006, pp. 229–240.
27. B. Zhou and J. Pei, Preserving Privacy in Social Networks against Neighborhood Attacks, in: *Proceedings of the IEEE International Conference on Data Engineering*, 2008, pp. 506–515.