Utility-Based Privacy Preserving Data Publishing

A Thesis

Submitted in partial fulfillment of the

requirements for the Degree of

Doctor of Philosophy

by

Korra Sathya Babu

Under the Supervision of

Professor Sanjay Kumar Jena

Department of Computer Science & Engineering

National Institute of Technology Rourkela

Rourkela – 769 008

SEPTEMBER 2013

This thesis is dedicated to my parents.

Department of Computer Science & Engineering
National Institute of Technology Rourkela
Rourkela-769 008, India. www.nitrkl.ac.in

Dr. Sanjay Kumar Jena
Professor

March 04, 2013

Certificate

This is to certify that this thesis entitled "**Utility-Based Privacy Preserving Data Publishing**", submitted by **Korra Sathya Babu**, a research scholar in the Department of Computer Science & Engineering, National Institute of Technology, for partial fulfillment for the award of the degree of **Doctor of Philosophy**, is a record of an original research work carried out by him under my supervision and guidance. The thesis has fulfilled all requirements as per the regulations of the institute and in my opinion has reached the standard needed for submission. The results embodied in this thesis have not been submitted to any other university or institute for the award of any degree or diploma.

Sanjay Kumar Jena

Acknowledgment

This dissertation would not have been possible without the constant encouragement I have received from my supervisor *Prof. Sanjay Kumar Jena*. He has constantly encouraged me to remain focused on achieving the goal. His observations and comments helped me to establish the overall direction of the research and to move forward with investigation in depth.

I am very much indebted to the Doctoral Scrutiny committee members *Prof. Prafulla Chandra Panda*, *Prof. Ashok Kumar Turuk* and *Prof. Durga Prasad Mohapatra* for their time to provide more insightful opinions into my research. I am also thankful to *Prof. Bansidhar Majhi*, *Prof. Santanu Kumar Rath*, Prof. *Bidyut Kumar Patra* and all the Professors of the department for their in time support, advice and encouragement.

I am really thankful to all my fellow research colleagues. My sincere thanks to everyone who has provided me new ideas and useful criticism, I am truly indebted.

I do acknowledge the academic resources that I have received from NIT Rourkela. I also thank the administrative and technical staff members of the Computer Science & Engineering Department for their intime support.

My special acknowledgement to all my teachers from school to now for their inspiring words, which brought me to this stage.

I feel proud to acknowledge my parents for their throughout support and motivation in my career for whom I am today and always. I would like to dedicate this thesis to my wife and my Father *Ramanna (Late)* and Mother *Kasulamma (Late)* for their unconditional love, patience and cooperation.

Korra Sathya Babu

Contents

List of Tables

List of Figures

Abstract

Advances in data collection techniques and need for automation triggered in proliferation of a huge amount of data. This exponential increase in the collection of personal information has for some time represented a serious threat to privacy. With the advancement of technologies for data storage, data mining, machine learning, social networking and cloud computing, the problem is further fueled. Privacy is a fundamental right of every human being and needs to be preserved. As a counterbalance to the socio-technical transformations, most nations have both general policies on preserving privacy and specific legislation to control access to and use of data. Privacy preserving data publishing is the ability to control the dissemination and use of one's personal information.

Mere publishing (or sharing) of original data in raw form results in identity disclosure with linkage attacks. To overcome linkage attacks, the techniques of statistical disclosure control are employed. One such approach is k-anonymity that reduce data across a set of *key variables* to a set of classes. In a k-anonymized dataset each record is indistinguishable from at least *k-1* others, meaning that an attacker cannot link the data records to population units with certainty thus reducing the probability of disclosure. Algorithms that have been proposed to enforce k-anonymity are Samarati's algorithm and Sweeney's Datafly algorithm. Both of these algorithms adhere to full domain generalization with global recording. These methods have a tradeoff between utility, computing time and information loss. A good privacy preserving technique should ensure a balance of utility and privacy, giving good performance and level of uncertainty. In this thesis, we propose an improved greedy heuristic that maintains a balance between utility, privacy, computing time and information loss.

Given a dataset and k, constructing the dataset to k-anonymous dataset can be done by the above-mentioned schemes. One of the challenges is to find the best value of k, when the dataset is provided. In this thesis, a scheme has been proposed to achieve the best value of k for a given dataset.

The k-anonymity scheme suffers from homogeneity attack. As a result, the l-diverse scheme was developed. It states that the diversity of domain values of the dataset in an equivalence class should be l. The l-diversity scheme suffers from background knowledge attack. To address this problem, t-closeness scheme was proposed. The t-closeness principle states that the distribution of records in an equivalence class and the distribution of records in the table should not exceed more than t. The drawback with this scheme is that, the distance metric deployed in constructing a table, satisfying t-closeness, does not follow the distance characteristics. In this thesis, we have deployed an alternative distance metric namely, Hellinger metric, for constructing a t-closeness table. The t-closeness scheme with this alternative distance metric performed better with respect to the discernability metric and computing time.

The k-anonymity, l-diversity and t-closeness schemes can be used to anonymize the dataset before publishing (releasing or sharing). This is generally in a static environment. There are also data that need to be published in a dynamic environment. One such example is a social network. Anonymizing social networks poses great challenges. Solutions suggested till date do not consider utility of the data while anonymizing. In this thesis, we propose a novel scheme to anonymize the users depending on their importance and take utility into consideration. Importance of a node was decided by the centrality and prestige measures. Hence, the utility and privacy of the users are balanced.

Chapter 1

Introduction

Advances in data storage, data collection and inference techniques have enabled the creation of huge databases of personal information. The collection of these data was driven by guidelines of government agencies, private corporations, hospitals or for mutual benefits. The guidelines may include various types of surveillance, mandatory disclosures, investigations, right to information, etc. Mutual benefits include disclosure of personal information by people due to coupons, discounts, profits, etc. Regardless of the techniques used for collecting data, dissemination of information from such databases, even if formally anonymized, paves serious threats to individual privacy through statistical disclosure. Proliferation of data mining techniques has fueled the privacy breach problem. Many civil rights groups and privacy groups oppose surveillance as a violation of people's right to privacy. Some of the groups include Electronic Privacy Information Center (EPIC), Electronic Frontier Foundation (EFF), American Civil Liberties Union (ACLU), etc. This resulted in setting up many laws pertaining to individual nation or organization. Examples are Personal Health Information Protection Act [PHIPA (2004)], Health Insurance Portability and Accountability Act [US Department of Health (1996)] and Data Mining Moratorium Act [DM Moratorium Act (2003)].

Privacy reflects diverse things to different people like seclusion (desire to be left alone), property (desire to be paid for one's data) or autonomy (ability to act freely).

Robert Ellis Smith, editor of the privacy journal states that privacy is "the desire by

each of us for physical space where we can be free of interruption, intrusion, embarrassment or accountability and the attempt to control the time and manner of disclosures of personal information about ourselves" [Robert (2004)]. As per *Louis Brandeis, United States supreme court justice*, "Privacy was the most cherished of freedoms in a democracy, and it's the individual right to be left alone" [Louis (1890)]. Privacy is a fundamental human right recognized in all major international agreements regarding human rights such as Article 12 of universal declaration of human rights [Human rights law (1948)]. *The preamble to the Australian Privacy Charter* declares that "A free and democratic society requires respect for the autonomy of individuals, and limits on the power of both state and private organizations to intrude on that autonomy. Privacy is a key value which underpins human dignity and other key values such as freedom of association and freedom of speech. Privacy is a basic human right and basic expectation of every person" [Australia (1994)]. *The Calcutt committee in UK* pointed out that "nowhere have we found a wholly satisfactory, statutory definition of Privacy" [Calcutt (1990)].

Privacy can also be classified from different aspects. It can be bodily privacy, communication privacy, territorial privacy or information privacy [Simon Davies (1996)]. *Bodily privacy* is concerned about protection of people's physical selves against invasive procedures such as genetic tests, drug testing and cavity searches. *Communication Privacy* covers the security and privacy of mail, telephones, e-mail and other forms of communication. *Territorial privacy* is concerned about the setting of limits on intrusion into the domestic and other environments such as the workplace or public space. This includes searches, video surveillance and identity checks. *Information privacy* involves the establishment of rules governing the collection and handling of personal data such as credit information, and medical and government records. Personal information is the information about an identifiable individual that is recorded in any form. It is also known as *data preserving*.

In this thesis, we consider only preserving of *information privacy*, which protects sensitive information from being brought to the attention of others. Privacy preserving is the ability to *control the dissemination* and use of one's personal information. Privacy

can refer to an individual where nobody should know about any entity after performing data mining or an organization to protect knowledge about a collection of entities. Various approaches followed for individual privacy preserving are data obfuscation, value swapping, perturbation, etc. Each organization adopts a framework for disclosing individual entity values to the public.

A data source comprises of public and private attributes. We need to reveal public attributes maximally without revealing the private attributes . Release of data either pseudonymous or anonymous may still be breached of privacy with inference [Mohammad and Somayajulu (2008), Verma and Toshniwal (2009)].

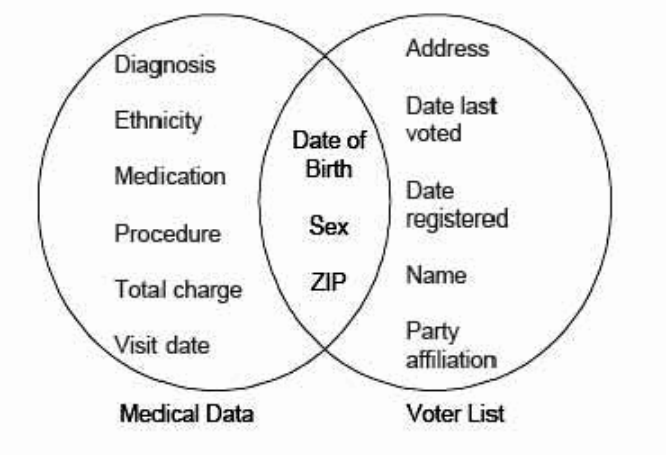

Figure 1.1: Linking to re-identify record owner.

An example given by Sweeney is reproduced in Figure 1 [Sweeney (2002b)]. An individual's name in a public voter list was linked with his record in a published medical database through the combination of zip code, date of birth and sex. Each of these attributes does not uniquely identify a record owner, but their combination, called the quasi-identifier (QI) [Dalenius (1986)], often singles out a unique or a small number of record owners. It is reported that 87% of the U.S. population had resembling characteristics that likely made them uniquely identified with the help of the above stated quasi-identifiers [Sweeney (2002b)].

Rest of the chapter is organized as follows: Section 1.1 mentions common attack models, anonymization operations and privacy models. Section 1.2 shows some important

application domains where privacy preserving techniques are frequently used. In Section 1.3, we discuss major issues and motivation of the thesis. Contributions of the thesis are highlighted in Section 1.4. Finally, organization of this thesis is discussed in Section 1.5.

1.1 Attack Models and Anonymization Operations

Dissemination of data is subjected to various kinds of attacks. This section discusses about the common attack models and general solutions. In all these types of attacks, it is assumed that the attacker knew the quasi-identifiers. Quasi-identifiers are those attributes of a table that can potentially help in inferring additional knowledge about a tuple in the table.

1.1.1 Attack Models and Countermeasures

- Record Linkage

  Record linkage identifies matching record pairs in two separate tables with the help of few quasi-identifier values [Rob and Stephen (2010)]. Record linkage attack has been widely used in crime detection, marketing, antiterrorism, etc. The negative part of record linkage is that it can lead to identity matching of private information of an individual. To overcome this attacks, k-anonymity was proposed [Sweeney (2002a)]. It states that a table comprises of equivalence classes and, each equivalence class should have at least k values.

- Attribute Linkage

  A table can be converted to a k-anonymous table before publishing it in a public domain. A situation may occur such that in a sigle equivalence class, all the domain values may be same. This leads to the homogeneity of the domain values in the equivalence class. In this case, the attacker may not precisely identify the record of the target victim, but could infer the sensitive values from the published table, based on the set of sensitive values associated to the

group that the victim belongs to. Examples of this type of attacks are similarity attack, homogeneity attack, skewness attack, background knowledge attack, etc. The attacker need not use the quasi-identifiers to perform this attack. To overcome this attacks, the l-diversity principle and t-closeness principle was proposed. The l-diversity principle states that there should be at least l-diverse domain values in an equivalence class [Machanavajjhala *et al.* (2007), Krishna and Valli (2011)]. The t-closeness principle states that the difference between the distributions of any equivalence class in the table and the distribution of the whole table should not be more than a threshold t [Ninghui *et al.* (2007)].

- Table Linkage

  Sometimes the presence or absence of the victim's record in the published table can be inferred by the attacker with certain level of confidence. These types of attacks are called table linkage attacks. Assume a table, T_1, is published. There exists another table T_2, such that, $T_1 \subseteq T_2$. Then if a tuple in T_1 can be identified based on T_2 with high level of confidence then we call this attack as table linkage attack. To overcome this type of attack, the δ-Presence model was introduced [Nergiz and Christopher (2010)]. δ-Presence states that given two tables, T_1 and T_2, the probability of inferring the victims record should be within δ range, *i.e.*, $\delta = (\delta_{min}, \delta_{max})$ based on the two tables. So, $\delta_{min} \leq P(victim's\ record \in released\ table) \leq \delta_{max}$.

1.1.2 Anonymization Operations

Anonymization is the process of modifying the relation in such a way that minimal sensitive data may be inferred. General techniques used to achieve it are: generalization and suppression, anatomization and permutation, and perturbation.

- Generalization and Supression

  A generalization scheme replaces some values with a parent value in the taxonomy of an attribute. The reverse operation of generalization is called

specialization. A suppression replaces some values with a special value, indicating that the replaced values are not disclosed. The reverse operation of suppression is called disclosure.

Given an attribute A, generalization for an attribute is a function on A. That is, each $f : A \rightarrow B$ is a generalization where B is on higher level of generalization on the taxonomy tree. So,

$$A_0 \xrightarrow{f_0} A_1 \xrightarrow{f_1} A_2...A_{n-1} \xrightarrow{f_n} A_n$$

is a generalization sequence or a functional generalization sequence.

Given an attribute A of a private table T, Domain Generalization Hierarchy (DGH) for A is a set of functions f_h where, $h = 0, ..., n-1$ such that:

$$A_0 \xrightarrow{f_0} A_1 \xrightarrow{f_1} A_2...A_{n-1} \xrightarrow{f_{n-1}} A_n$$

So, DGH of an attribute A is over: $\bigcup_{h=0}^{n-1} A_h$

There are basically five types of generalizations [Benjamin *et al.* (2010)] namely, full generalization, subtree generalization, sibling generalization, cell generalization and multidimensional generalization.

- Anatomization and Permutation

  Anatomization and permutation dissociate the correlation between quasi-identifiers and sensitive attributes by grouping and shuffling sensitive values in a quasi-identifier group. Unlike generalization and suppression, anatomization does not modify the quasi-identifier or the sensitive attribute, but dissociates the relationship between the two. The idea of the permutation is to dissociate the relationship between a quasi-identifier and a numerical sensitive attribute by partitioning a set of data records into groups and shuffling their sensitive values within each group.

- Perturbation

Perturbation distorts the data by adding noise, aggregating values, swapping values or generating synthetic data based on some statistical properties of the original data. The general idea is to replace the original data values with some synthetic data values so that the statistical information computed from the perturbed data does not differ significantly from the statistical information computed from the original data.

1.2 Applications of Privacy Preserving Technique

Public data should not be released without preserving the privacy of each record. It should ensure that no individual is identified from the anonymized data. This technique has been utilized in numerous application domains. Some of them are listed below.

- **Clinical Records**

  Huge number of clinical records are gathered through registration of patients in clinics. These data are shared with the scientific and government firms for research and mobilization. The clinical records can be anonymized before publishing the records with other firms.

- **Social Networking**

  People these days have become savvy to the internet and social networking. People are unaware of the privacy issues. The published data in the social network can be anonymized and do not infer to any single individual

- **Government and Financial Companies**

  The government and financial companies are bound to publish certain data to the public. The data to be published may be anonymized so that no one uncovers any identity and lead to self degradation.

- **Business**

  Huge data is generated by piling up data from everyday shopping of customers.

The retailers may share with other retailers for mutual benefit. The anonymization algorithms can be applied on the data to ensure the privacy of each individual.

- **Rule Enforcement**

  Privacy is a fundamental right of every human being. Every nation ensures privacy preserving guidelines for the data stewards and data collectors. Many of the rules can be enforced by incorporating anonymization algorithms on the dataset that needs to be released.

- **Surveillance**

  Raw samples collected from surveillance and features of biometric traits of each individual is stored in databases. It may end up in the hands of a bad person and he takes advantage of the available feature. Privacy preserving algorithms can be tailored and deployed on the databases storing raw data of each individual.

1.3 Major Issues, Motivation and Problem Definition

1.3.1 Issues and Motivation

Projected below are few issues faced while publishing data to the public.

Issues with tradeoff between utility and privacy

Algorithms that ensure privacy should notably ensure the utility of the published data [Verykios *et al*. (2004), Ming and Jian (2008), Pin Lv (2008), Kok and Myung (2012), Junqiang Liu (2011)]. High privacy ensured data may be minimal in utility and vice versa. A good privacy preserving technique should ensure a balance of utility and privacy, giving good performance and level of uncertainty.

Issues with finding the best value of k for k-anonymity

Given a dataset and k, constructing the dataset to k-anonymous dataset can be done. One of the challenges is to find the best value of k for a given dataset.

Issues with finding the quasi-identifiers

A released dataset can be used to infer sensitive data with the help of the quasi-identifiers of the released dataset. Quasi-identifiers are those set of attributes that contribute in inference. Quasi-identifiers depends on the dataset and it is a great challenge to identify them.

Issues with online privacy

Conventional techniques existing in the literature cannot be adopted directly to provide privacy in online environment [Raymond *et al.* (1984)]. They need to be tailored. Moreover, its dynamic nature poses huge challenges.

1.3.2 Problem Definition

Given a dataset $\mathcal{D}$, the goal is to find $\mathcal{D}'$ using some anonymization function f_A: $\mathcal{D} \rightarrow \mathcal{D}'$, subjected to the following conditions:

- Maximum similarity between $\mathcal{D}$ and $\mathcal{D}'$ *i.e*, minimum $|\mathcal{D} - \mathcal{D}'|$,
- Minimum information loss and
- Abide by the quasi-identifies with their domain generalization hierarchies.

1.4 Contributions of the Thesis

In this thesis, we investigate the various privacy preserving data publishing techniques and study on the privacy and utility issues of the existing mechanisms. We proposed various algorithms to improve the utility of the data while still preserving the privacy. Contributions of this thesis are summarized as follows:

- Before the release of any data product, there is a need to consider a balance between utility and privacy. An improved heuristic is proposed for achieving k-anonymity. The proposed algorithm has a better balance between privacy, computing time,

information loss and utility. The proposed algorithm was compared with the established Datafly and Samarati's algorithms. It is observed that the new greedy algorithm performed better than the established ones for balancing utility and privacy.

- In order to achieve utility and privacy best value of k is required. Various experiments were conducted on the Adult dataset with different ranges. It was observed that the value of k differed from dataset to dataset. A procedure has been suggested for determining the best value of k for a dataset.

- The k-anonymity scheme suffers from homogeneity attack. As a result, the l-diverse scheme was developed. The l-diversity scheme suffers from background knowledge attack. To address this problem, t-closeness scheme was proposed. The drawback with this scheme is that, the distance metric deployed in constructing a table, satisfying t-closeness, does not satisfy the distance characteristics. In this thesis, we have deployed an alternative distance metric (Hellinger) that satisfy the distance characteristics for constructing the t-closeness table. In addition to this, an improved t-closeness was proposed to preserve the identity disclosure.

- The k-anonymity, l-diversity and t-closeness schemes can be used to anonymize the dataset before publishing (releasing or sharing) in a static environment. There are situations where data need to be published in a dynamic environment. One such example is the social network where huge information can be derived using data mining [Bellaachia and Anasse (2011), Rajput *et al.* (2011), Yung and Lin (2012)]. Of the late research states that anonymizing social networks poses great challenges. Current solution adopted, is to leave the freedom of setting the privacy levels to users. This poses another problem like every user setting the privacy level to the maximum. This reduces the utility of the data. In this thesis, we propose a novel scheme to anonymize the users depending on their importance [Huangmao *et al.* (2012)]. Therefore the utility and privacy of each user are balanced before they are published in the public domains.

1.5 Organization of the Thesis

The thesis is organized as follows:

Chapter 2 presents a survey of existing methods and background of the present work. Contributions of the thesis are discussed in subsequent chapters, Chapter 3 to Chapter 6. Conclusions and future research directions are discussed in Chapter 7.

Chapter 3 presents the strengths and weaknesses of the existing algorithms for k-anonymity. The observations of the results motivated to move in the direction of proposing an improved greedy approach for balancing utility, privacy, information loss and computing time.

Chapter 4 proposes a method to find the best value of k for a given dataset so that the dataset may be later converted to k-anonymous.

Chapter 5 discusses the t-closeness method of achieving privacy. An alternative distance metric was embedded in the algorithm of t-closeness for better performance.

Chapter 6 proposes a novel method to anonymize dynamic structures like nodes in social networks to ensure privacy of users based on their significance in the network.

Chapter 7 summarizes overall contributions of the thesis and discusses future research directions.

Chapter 2

Background and Related Work

2.1 Introduction

The concern for privacy dates back to 1948 in the United Nations Declaration of Human Rights Article 12 [Human rights law (1948)]. According to Sweeney, the three pillers of privacy are consent (obtain permission to access the information), notice (notify before using one's personal data) and de-identification (altering to remove certain data elements associated with an indivudual). In a deeper sense, the concern is the identification of individuals within anonymized data through the use of linkage attacks. Several models has been investigated over the years to keep the data de-identified from a published anonymized data [Samarati and Sweeney (1998b), Machanavajjhala *et al.* (2007), Ninghui *et al.* (2007), Nergiz and Christopher (2010)].

In the remainder of this chapter, Section 2.2 gives background; Section 2.3 describes the k-anonymity model. We review algorithms that give full domain generalization with record suppression. Section 2.4 describes other models of privacy preservation. Section 2.5 gives a survey on anonymizing social networks followed by conclusion in Section 2.6.

2.2 Background

This section provides the basic definitions and preliminaries.

2.2.1 Relation

Let $B(A_1, ..., A_n)$ be a relation with a finite number of tuples. The finite set of attributes of B are $\{A_1, ..., A_n\}$. Given a relation $B(A_1, ..., A_n)$, $\{A_i, ..., A_j\} \in \{A_1, ..., A_n\}$, and a tuple $t \in B$, we use $r[A_i, ..., A_j]$ to denote the sequence of the values, $v_i, ..., v_j$, of $A_i, ..., A_j$ in r. $B[A_i, ..., A_j]$ denotes the projection, maintaining duplicate tuples, of attributes $A_i, ..., A_j$ in B.

Given a population U, a person-specific table $T(A_1, ..., A_n)$, f_c: $U \rightarrow T$ and f_g: $T \rightarrow U'$, where $U \subseteq U'$. A quasi-identifier of T, written Q_T, is a set of attributes $\{A_i, ..., A_j\} \subseteq \{A_1, ..., A_n\}$ where $\exists pi \in U$ such that $f_g(f_c(pi)[Q_T]) = pi$.

2.2.2 Quasi-identifier

A quasi-identifier is a set of attributes in a relation T that can be linked to external information to re-identify individual records. The quasi-identifier when combined with corresponding external data can lead to the correct association of a data record with a population unit (also known as *identification disclosure*). Whether a variable is a quasi-identifier or not depends on a variety of factors, the most important is the availability of external data with a variable that corresponds to the potential quasi-identifer.

2.2.3 Generalization

A set of values of a particular attribute is called a *domain*. Given two domains S0 and S1, the expression S0 $\leq$ S1 means that values of attributes in S1 are more generalized than those in S0. Generalization T_1 is *k-minimal* if it satisfies k-anonymity and there does not exist another generalization satisfying these conditions less general than T_1.

There are several schools of thought about how generalization ontologies should be generated and represented [Samarati and Sweeney (1998a)]. The Figures 2.1, 2.2, 2.3 and 2.4 shows examples of domain generalizatin hierarchies and value generalization hierarchies.

In most of the anonymizing models the anonymizing operation used is generalization

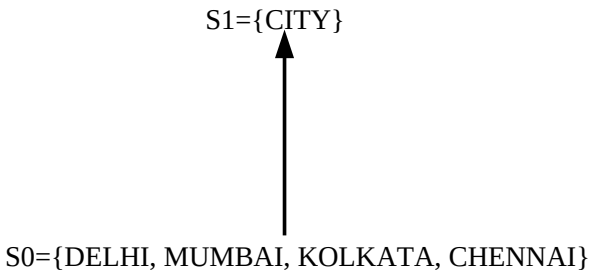

Figure 2.1: Domain generalization hierarchy for residence.

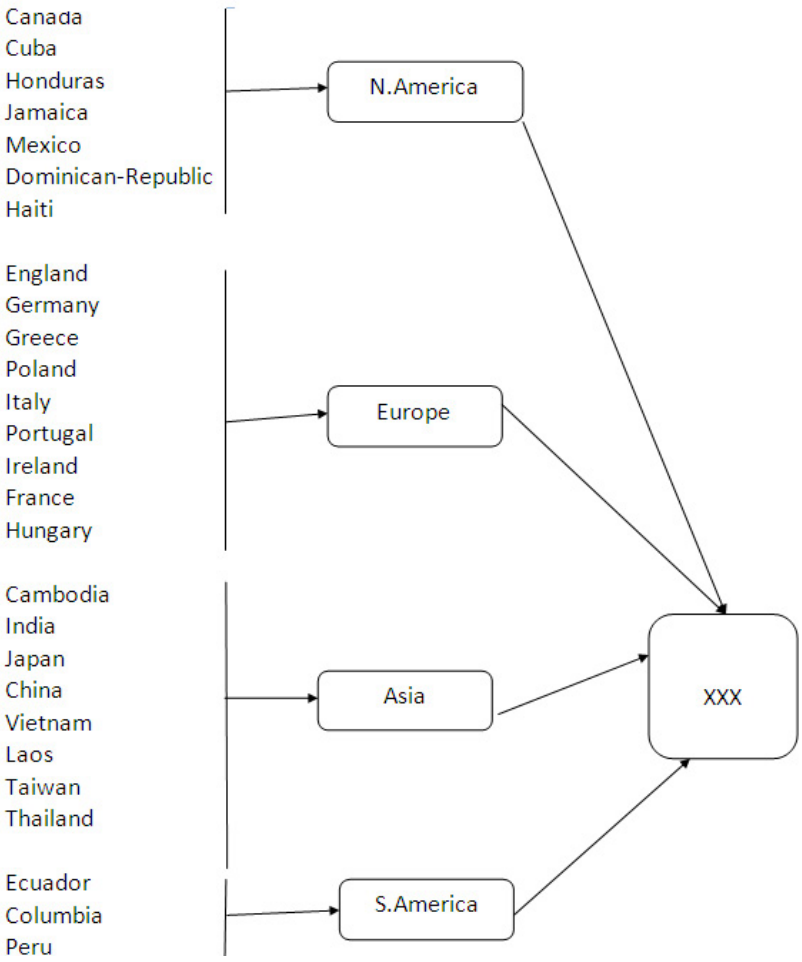

Figure 2.2: Value generalization hierarchy for native country.

and suppression. These operations are performed on the quasi-identifiers of the table. Figure 2.5 shows a generalization hierarchies of three quasi-identifiers namely, age, gender and zipcode. Anonymization can be achieved in different combinations of these generalization hierarchies. This can be viewed like a lattice. Figure 2.6 shows the generalization lattice of the generalization hierarchies of Figure 2.5.

2.3 *k*-anonymity

A relation is comprised of QIs and non-QIs. The QIs need to be anonymized as they can re-identify individual records. Let r be a tuple, then, the value of i^{th} tuple can be represented as $r_i[C]$.

A relation, T satisfies k-anonymity if for every tuple $\mathrm{r}_{i_0} \in \mathrm{T}$, there exist *k-1* other tuples r_{i_1}, r_{i_2}, ... $\mathrm{r}_{i_{(k-1)}} \in \mathrm{T}$, such that $\mathrm{r}_{i_0}[\mathrm{C}] = \mathrm{r}_{i_1}[\mathrm{C}] = ... = \mathrm{r}_{i_{(k-1)}}[\mathrm{C}]$, $\forall$ C $\in QI$.

The first model for anonymizing the data before publishing is the k-anonymity model given by Sweeney [Sweeney (2002b)]. As an example, Table 2.2 is a *2*-anonymous table of Table 2.1.

Table 2.1: Example of original table.

Age	**Gender**	**Zipcode**
21	Male	769008
56	Female	769008
35	Female	769008
24	Male	769008
60	Male	769132
22	Female	743161
24	Male	743372

A k-anonymous dataset can be achieved through local recording or global recording. Local recording refers to different generalization rules within a column to equal the values of the domain. Global recording (also known as full domain generalization (FDG)) apply the same rule throughout the column.

Figure 2.3: Value generalization hierarchy for race.

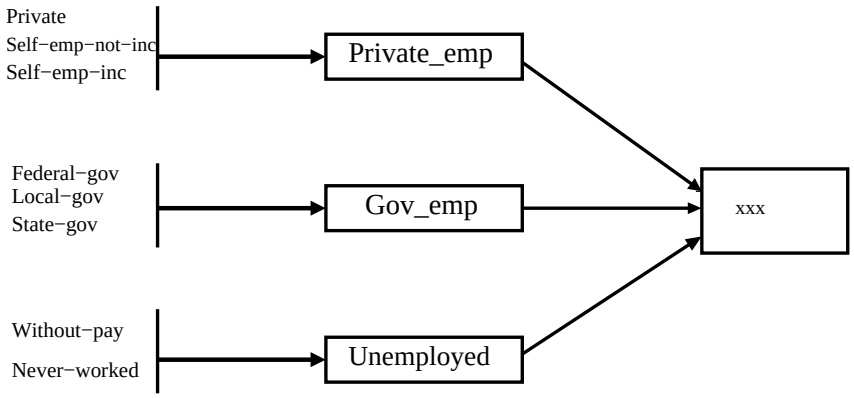

Figure 2.4: Value generalization hierarchy for workflow.

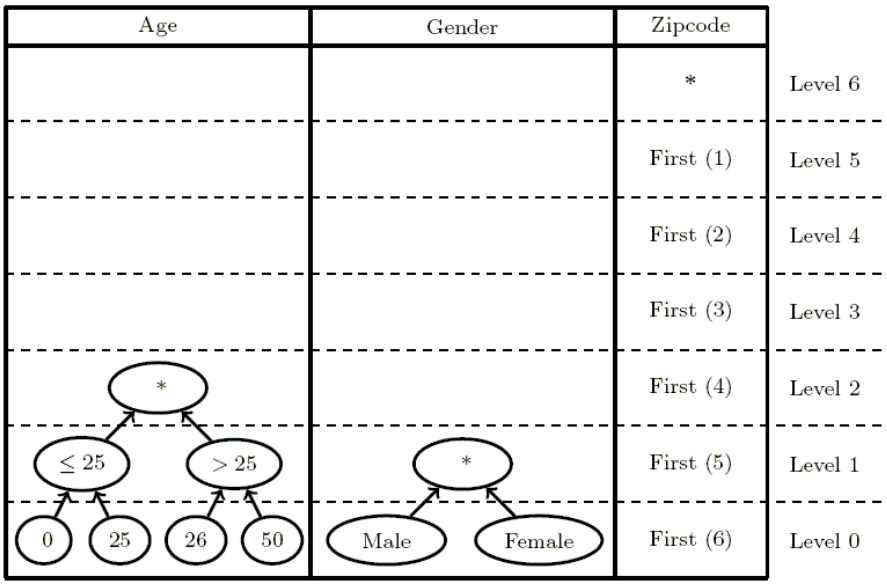

Figure 2.5: Generalization hierarchies.

Table 2.2: *2*-anonymization of Table 1.

Age	**Gender**	**Zipcode**
< 25	*	769***
≥ 25	*	769***
≥ 25	*	769***
< 25	*	769***
≥ 25	*	769***
< 25	*	743***
< 25	*	743***

2.3.1 Global Recording Algorithms

Datafly Algorithm

The Datafly algorithm [Sweeney (1997)] goes with the assumptions that the best solutions are the ones that are attained after generalizing the variables with the most distinct values (unique items). The search space is the whole lattice. However, this approach only goes through a few nodes in the lattice to find its solution. This approach is very efficient from a time perspective. Datafly uses a greedy algorithm to search the domain generalization hierarchy. At every step, it chooses the locally optimal move. One drawback with Datafly's approach is that it may become trapped in a local optimum [Cormen *et al.* (2001)].

Samarati Algorithm

Samarati's algorithm assumes that the best solutions in the lattice are the ones that result in a table having minimal generalizations [Samarati (2001)]. So, the solutions are available in the height that is minimal in a lattice. The algorithm is based on the axiom that if a node at level h, in domain generalization hierarchy satisfies k-anonymity, then all the levels of height higher than h also satisfy k-anonymity. In order to search the lattice and identify the the lowest level with the generalizations that satisfy $k-$anonymity with minimal suppression, Samarati used binary search. The algorithm goes through the lattice with a binary search, always cutting the search space in half. It goes down the level if a solution is found at that level, otherwise it goes up the lattice. Eventually, the algorithm finds the solution with the lowest height with the least generalizations. This

level ensures less information loss but time consumed is higher than Datafly.

Incognito Algorithm

Incognito [Lefevre *et al.* (2005)] implements a dynamic programming approach with an idea that if a subset of quasi-identifiers of a relation T is not k-anonymous then the relation T cannot be k-anonymous. The approach constructs generalization lattice of each individual subset of quasi-identifiers and traverses them by performing a bottom-up, breadth-first search. The number of generalization lattice constructed in case of Incognito for QIs of order r is 2^r. Thus Incognito algorithm is of order $\Omega(2^r)$ because at least one lattice is checked for k-anonymity in every generalization lattice.

Optimal Lattice Algorithm

El Emam *et al.* proposed an algorithm called Optimal Lattice Anonymization and showed that it outperforms Incognito [Emam *et al.* (2009)]. It use predictive tagging to prune the search space of the lattice. However, if global optimal k-anonymous lattice lie on or above the middle level of full domain generalized hierarchy, then the algorithm check all the middle level lattices for k-anonymity. We will show in Proposition 2 of Chapter 3, that checking only the middle level of full domain generalized hierarchy is exponential in number of QIs.

Flash Algorithm

Kohlmayer *et al.* proposed Flash algorithm for finding efficient, stable and optimal k-anonymity [Kohlmayer et al. (2012)]. They implemented their algorithm in a frame work, which holds all the data in a main memory with dictionary compression on these data. The maximum number of quasi-identifiers for the datasets considered by them is only nine. If the number of quasi-identifiers are very high, then it would be difficult to put all the data items in the main memory.

The algorithms listed above have their pros and cons. They have tradeoffs between utility of the dataset, information loss and computing time.

2.4 Closeness

The *k*-anonymity method can limit the disclosure risk of publishing data. It arrange the records of a table in such a way that at least k records exist in an equivalence class for the quasi-identifiers. Table 2.3 shows the inpatient data and Table 2.4 shows the 4-anonymous inpatient data. Though *k*-anonymity protects against identity disclosure, it does not provide sufficient protection against attribute disclosure [Aggarwal *et al.* (2005), Ninghui *et al.* (2007)]. Two attacks were identified, homogeneity attack and the background knowledge attack. Table 2.4 shows that, having a little knowledge of any of the non-sensitive identifiers, the sensitive attribute like disease can be easily determined as they are homogeneous in nature in the last equivalence class. As a countermeasure to these attacks, the *l*-diversity was introduced [Machanavajjhala *et al.* (2007), Ninghui *et al.* (2007)].

The *l*-diversity principle represents an important step beyond *k*-anonymity in protecting against attribute disclosure. The principle of *l*-diversity states that a table is said to possess *l*-diversity if every equivalence class of the table has atleast *l*-diversified values for the sensitive attribute. Tables 2.5 and 2.6 shows an example of a original table and its *3*-diverse table respectively. There are various instantiations of *l*-diversity namely, distinct *l*-diversity, entropy *l*-diversity and recursive (*c,l*)-diversity. A table is said to have distinct *l*-diversity if each equivalence class has at least l well-represented sensitive values. A table is entropy *l*-diverse if for every equivalence class Equation (2.1) holds true.

$$-\sum_{s \in S} P(QI,\ s)\ log(P(QI,\ s))\ \geq\ log\ l \tag{2.1}$$

where,

S is a sensitive attribute,

QI is the quasi-identifier and

$P(QI,\ s)$ is the fraction of records in a *qid* group having the sensitive value s.

A large threshold value l will definitely have low certainty of inferring a particular sensitive value in an equivalence class. The drawback of entropy *l*-diversity is that it does not handle

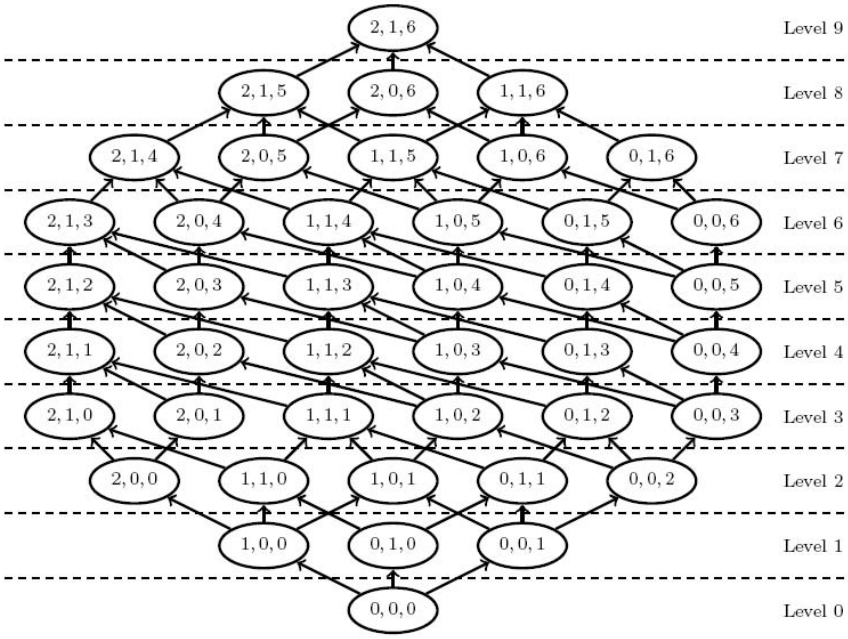

Figure 2.6: Lattice.

Table 2.3: Inpatient data.

#	**Zipcode**	**Age**	**Nationality**	**Disease**
1	13053	28	Russian	Heart Disease
2	13068	29	American	Heart Disease
3	13068	21	Japanese	Viral Infection
4	13053	23	American	Viral Infection
5	14853	50	Indian	Cancer
6	14853	55	Russian	Heart Disease
7	14850	47	American	Viral Infection
8	14850	49	American	Viral Infection
9	13053	31	American	Cancer
10	13053	37	Indian	Cancer
11	13068	36	Japanese	Cancer
12	13068	35	American	Cancer

probability based risk. The recursive (*c,l*)-diversity ensures that the most frequent value does not occur too frequently and less frequent values do not occur too rarely.

For an equivalence class, let r_i denote the number of times the i^{th} most frequent sensitive value appears in that equivalence class. Given a constant c, the equivalence class satisfies recursive (*c,l*)-diversity if the Equation (2.2) is satisfied.

$$r_1 < c(r_l + r_{l+1} + ... + r_m) \tag{2.2}$$

A table T satisfies recursive (*c,l*)-diversity if every equivalence class in T satisfies recursive (*c,l*)-diversity.

The *l*-diversity privacy model cannot prevent attribute disclosure. Two attacks are possible on a *l*-diversity table namely, skewness attack and similarity attack. When the overall distribution is skewed, satisfying *l*-diversity does not prevent attribute disclosure. When the sensitive attribute values in an equivalence class are distinct but semantically similar, an adversary can learn important information. The distributions that have the same level of diversity may provide very different levels of privacy. This is because there are semantic relationships among the attribute values, as different values have very different levels of sensitivity and privacy is affected by the relationship with the overall distribution. To prevent skewness attack, the *t*-closeness model was introduced [Ninghui (2007)]. This principle requires the distribution of a sensitive attribute in any group on the quasi-identifiers to be close to the distribution of the attribute in the overall table.

2.4.1 *t*-closeness

An observer has some prior observation, O_1 about the sensitive attributes of a table. After he or she observes the released table containing sensitive attributes, even if formally anonymized, there may be a change in the current observation. Let us consider this posterior observation as O_2. The information gain of the observer is the difference between these two observations.

Prior belief (represented as Q) depends on the distribution of a sensitive attribute in

Table 2.4: 4-anonymous inpatient data.

#	Zipcode	Age	Nationality	Disease
1	130**	<30	*	Heart Disease
2	130**	<30	*	Heart Disease
3	130**	<30	*	Viral Infection
4	130**	<30	*	Viral Infection
5	1485*	≥ 40	*	Cancer
6	1485*	≥ 40	*	Heart Disease
7	1485*	≥ 40	*	Viral Infection
8	1485*	≥ 40	*	Viral Infection
9	130**	3*	*	Cancer
10	130**	3*	*	Cancer
11	130**	3*	*	Cancer
12	130**	3*	*	Cancer

Table 2.5: Example of original table.

#	Zipcode	Age	Salary	Disease
1	47677	29	3K	gastric ulcer
2	47602	22	4K	gastritis
3	47678	27	5K	stomach cancer
4	47905	43	6K	gastritis
5	47909	52	11K	flu
6	47906	47	8K	bronchitis
7	47605	30	7K	bronchitis
8	47673	36	9K	pneumonia
9	47607	32	10K	stomach cancer

Table 2.6: Example of 3-diverse table.

#	Zipcode	Age	Salary	Disease
1	476**	2*	3K	gastric ulcer
2	476**	2*	4K	gastritis
3	476**	2*	5K	stomach cancer
4	4790*	≥ 40	6K	gastritis
5	4790*	≥ 40	11K	flu
6	4790*	≥ 40	8K	bronchitis
7	476**	3*	7K	bronchitis
8	476**	3*	9K	pneumonia
9	476**	3*	10K	stomach cancer

the whole table. Posterior belief (represented as P) depends on the distribution of the sensitive attribute value in an equivalence class. So, information gain is the difference between these two distribution can be denoted as (D[P,Q]).

Requring t-closeness would result in records of one equivalence class be grouped with the records of other equivalence class to make the distribution close to the overall distribution. This greatly reduces the utility of the data, as it hides the very information one wants to discover. This motivated for *(n,t)*-closeness model.

An equivalence class of a set, E_1 is said to have *(n,t)*-closeness if there exists a set E_2 of records that is a natural superset of E_1 such that E_2 contains at least n records, and the distance between the two distributions of the sensitive attribute in E_1 and E_2 is no more than a threshold t. A table is said to have *(n,t)*-closeness if all equivalence classes have *(n,t)*-closeness.

Ninghui *et al.* state that the metric used in the above methods for determining closeness of the dataset does not satisfy the properties of a distance metric [Ninghui *et al.* (2007)]. The t-closeness principle protects against attribute disclosure but not identity disclosure.

2.5 Privacy issues in Social Networks

Social networking and data mining has emerged as key areas of research interest in recent times. Social networks reflect the concept of small world representing many real-world instances [Travers and Milgram (1969)]. Social Network consists of social entities and ties between them [Wasserman and Faust (1994)]. Social entities, known as actors or nodes, can be an individual, corporate or an organization. Social ties, also called as edge, can be a kinship, social role, affective, cognitive, action, flow, transfer of material resources, distance, co-occurrence, etc. On the other hand data mining is the process of discovering interesting patterns from huge volumes of data. The intersecting research interest between social networks and data mining algorithms has posed problems like privacy and has received considerable attention as of the late. Qiang Yang *et al.* address privacy as one of the ten challenges of data mining [Qiang and Xindong (2006)]. Mining of data must be

done taking privacy into consideration [Qiang (2011), Aggarwal and Yu (2008)]. Privacy depends from person to person and may be in various forms like relational [Na *et al.* (2011)], influential [Jie *et al.* (2009a)] and emotional [Jie *et al.* (2009b)]. Efficient methods do exist to infer great treasure of knowledge in social networks [Gautam *et al.* (2008)].

The conventional security systems does not ensure privacy. Data encryption has its own limitation due to key sharing problem, computational cost and efficiency, trust violation and intransitiveness of trust. A definition in [Bin and Jian (2008)] states that general definition of privacy must be one that is measurable, of value and actionable. Privacy has three interrelated classes; secrecy, anonymity and solitude. Secrecy is concern about the information that other may gather anonymity about how the information is generalized. Solitude measures the degree of physical access.

Social network data pose a different and unique challenge in modeling privacy due to highly interrelated nodes. These challenges include modeling adversary background knowledge, calculating information loss and deciding anonymization method [Aggarwal and Wang (2010)]. Apart from modeling privacy, the utility of the published data in social network is attributed by degree, path length, transitivity, network reliance and infectiousness [Hay *et al.* (2008)].

2.5.1 Attacks on Social Networks

The attacks on social network data publishing can be broadly classified into two categories [Backstrom *et al.* (2007)]. (i) Active attack: The attacker creates new user accounts and edges in the original network and uses them to find targets and their relations in anonymized network. (ii) Passive attack: The attackers consider the uniqueness of small random sub-graphs embedded in a network. The attack is launched observing a particular node or sub-graph and re-identify them in the anonymized graph. Another classification of attacks can be made based on the target to which the attack is aimed. Such attacks are classified into three categories namely identity disclosure attack, link disclosure attack and content disclosure attack [Liu *et al.* (2008)].

Identity Disclosure Attack

This is a kind of attack in which the attacker tries to reveal the identity of the actors in the anonymized network. The solutions proposed for this are (i) k-candidate anonymity and graph randomization [Liu *et al.* (2008), Hay *et al.* (2007), Samarati and Sweeney (1998b)], (ii) k-degree anonymity and minimal edge modification [Liu *et al.* (2008)] and (iii) k-neighborhood anonymity and graph isomorphism [Zhou and Pei (2008)]. A variation of identity disclosure attack is neighborhood attack. In this type of attack, the attacker tries to identify the target (node or sub-graph) having a background knowledge of its neighbors. Liu *et al.* demonstrated that even if all the labels are removed from the network, an attacker will be able to reproduce the original network with the help of background knowledge [Liu *et al.* (2008)]. If all labels in the published data are removed then the utility is hampered.

Figure 2.7 shows a model social network. To publish this network, the first approach is to remove labels of all the edges and vertices as in Figure 2.8. But still the attack on this network is possible if the attacker knows the background information about the neighbor of the target node which is shown in Figure 2.9.

Background Knowledge 1: One node in Figure 2.8 has degree 4. Using this knowledge the attacker now identified the node 4 in the networks as shown in Figure 2.9.

Background Knowledge 2: Node 4 has one neighbor with degree one in Figure 2.9. So node 3 is discovered in Figure 2.10. Likewise, rest of the nodes in the network are revealed. This is shown in Figure 2.11.

Link Disclosure Attack

Link disclosure attack aims at revealing an important relation between a pair of actors. The relations between actors are very significant in social networks and signature property of the social network representation. Sensitive edge disclosure attack belongs to this category. To overcome this problem of link re-identification, Zheleave and Getoor proposed a concept of sensitive edge and observed edges to attain the balance between utility and privacy through series of algorithms named intactedges and partial edge

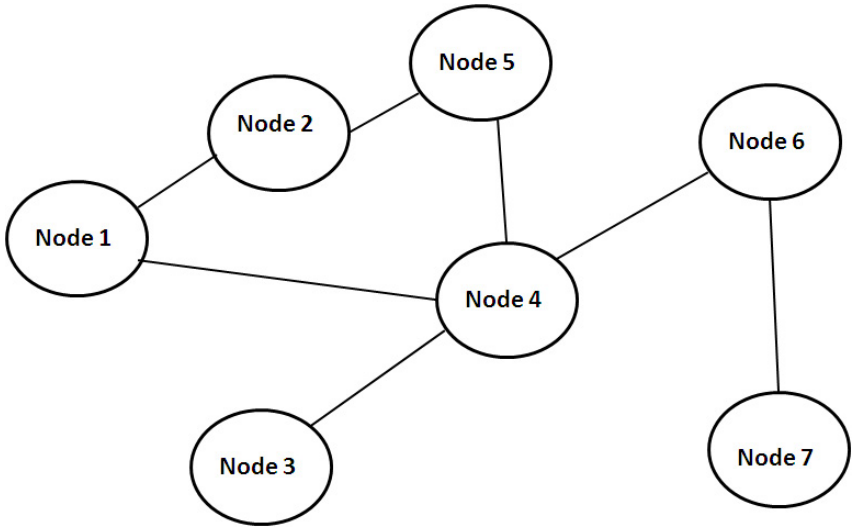

Figure 2.7: A social network.

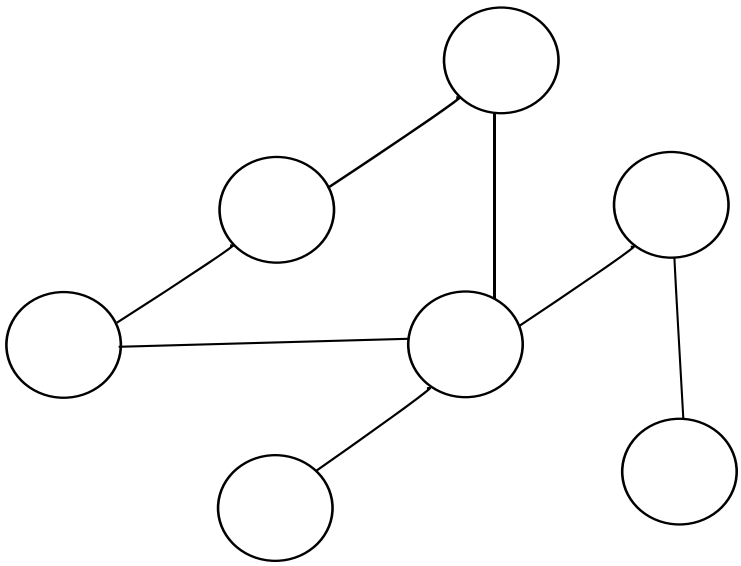

Figure 2.8: Label of the edges and vertex are removed.

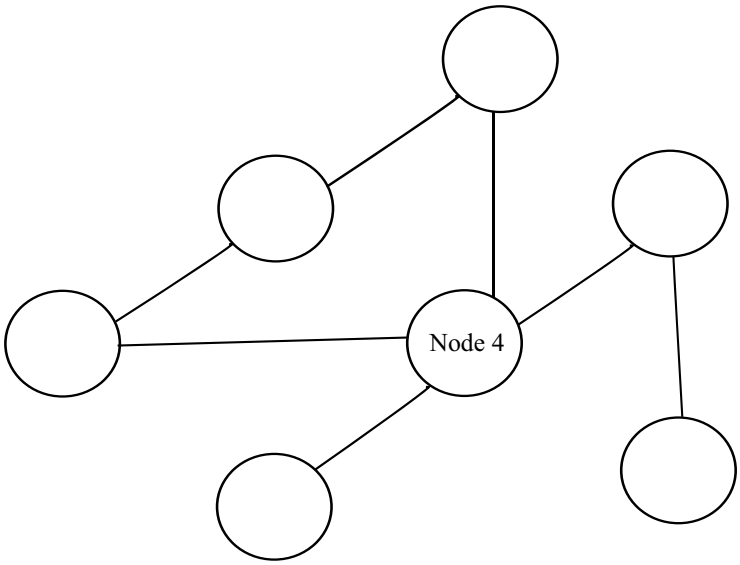

Figure 2.9: Node 4 identified.

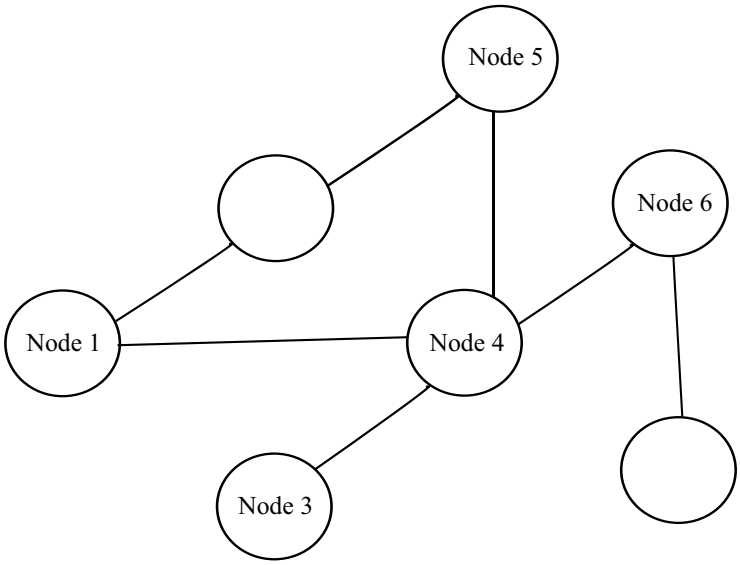

Figure 2.10: Node 3, 1, 5 and 6 identified.

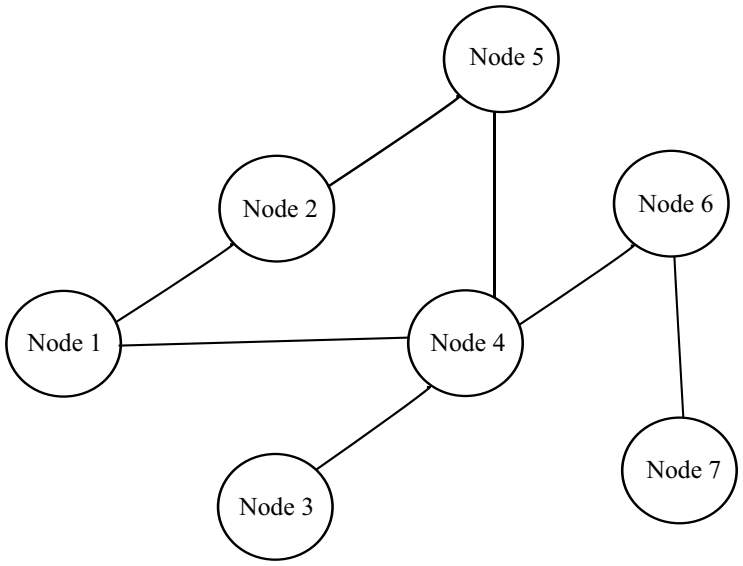

Figure 2.11: Entire network revealed.

removal [Zheleva and Getoor (2007)]. Other techniques found are privacy preserving link analysis, random perturbation for privacy relationship protection [Ying and Wu (2008)], cryptographic protocol for private relationships protection [Curminati *et al.* (2007)] and synthetic graph generation [Leskovec and Faloutsos (2007)].

Content Disclosure Attack

Sometimes private information associated with the network is revealed. These kinds of attacks generally launched by advertising organizations on online social networking sites. The activities of the users are notified to the friends associated with them. An example of the content disclosure attack is Facebook's Beacon service attack [Liu *et al.* (2008)].

2.5.2 Social Network Anonymization

The levels of privacy expected by each user in the network vary. A solution provided by Xiao and Tao is the concept of personalized anonymity [Xiao and Tao (2006)]. Here a user in a social network can specify the degree of privacy protection for his/her sensitive values.

The preferences of the users' can be collected from them when he or she is supplying the data. This scheme is focused on the interests of the users as a primary criterion. Therefore, high privacy can be ensured but may tend to provide very little utility of the published data.

A sub-graph matching scheme using minimum depth first search is proposed by Zhou and Pei which relates sub-graphs in a graph and anonymize the labels of these sub-graphs [Zhou and Pei (2008)]. Since the social networks have millions of nodes and finding the similar sub-graphs are very challenging in a big network, this scheme seems impractical in real life. It considers only one neighbor. On increasing the number of nodes to more than one, the size of neighbors grows exponential [Tripathy and Panda (2010)]. The anonymization approaches used by [Zhou and Pei (2008), Tripathy and Panda (2010)] statically anonymized the nodes. The concern for utility of the data is not emphasized in these schemes.

Liu *et al*. proposed two methodologies to solve the problem of sensitive edge disclosure attack [Liu *et al*. (2008b)]. They used the Gaussian randomization and greedy perturbation scheme. These two methods are computationally intensive and consume huge time. Still, this method is appreciated but they do not consider the dynamic nature of the weight associated with the edges.

Anonymizing dynamic structures like nodes in a social network are challenging. Algorithms available in the literature focus only on anonymity of nodes in a social network. Hence, the utility of these data after publishing is mostly neglected. Increase in privacy level decreases the utility of the data and vice versa. Optimal balance of utility and privacy is a NP hard problem. It is a fact that a reputed person in the society needs more privacy than the person with less reputation. Based on the social phenomena of importance of an user, we anonymize the social network. This gives a balancing of utility and privacy.

2.6 Conclusion

This chapter discussed basic background of various privacy models that can be used on the dataset before publishing them. Various attacks were also projected concerning to each of the privacy model.

Chapter 3

Achieveing *k*-anonymity with Improved Heuristics

3.1 Introduction

Linkage attack is one of the major covert channels for violating privacy. To address the problem of linkage attack, various techniques of statistical disclosure control are employed. One such approach called k-anonymity, works by reducing data across a set of *key variables* to a set of classes [Samarati and Sweeney (1998b)]. Other variations of k-anonymity can be found in [Sweeney (2002a,b), Aggarwal *et al*. (2005), Bayardo and Agrawal (2005), Gionis and Tassa (2007)]. In a k-anonymized dataset each record is indistinguishable from at least *k-1* other records. Therefore, an attacker cannot link the data records to population units with certainty thus reducing the probability of disclosure. However, preserving privacy through statistical disclosure control techniques reduce the utility of the data. Most of the techniques proposed in literature do not focus on the tradeoff issues between privacy and utility, information loss and computing time to achieve k-anonymity.

The method described in this chapter maintains a balance between computing time and information loss. An improved greedy heuristic was deployed to attain the balance of the tradeoffs. Unlike the heuristic used in Datafly, the improved greedy heuristic considers all the higher level generalization nodes that are associated to the current node

in the generalization lattice. The computing time is the time taken by a method to convert a given dataset to k-anonymity. The information loss is the loss occurred in the process of anonymization due to generalization of attributes. More details are provided in subsequent sections. Three metrics, namely information loss, computing time and level of generalization are used to compare the proposed algorithm with the exiting ones.

In the remainder of this chapter, Section 3.2 describes the related work. Algorithms that give full domain generalization with record suppression were considered. We observe that there is a tradeoff between anonymization achieved and computing time. Section 3.3 describes the details of the proposed improved greedy approach. Section 3.4 describes experiments conducted on two benchmark datasets. Section 3.5 gives the conclusions.

3.2 Related Work

Several algorithms have been developed with the purpose of making de-identified data k-anonymous [Ciriani *et al.* (2007), Truta *et al.* (2008), Benjamin *et al.* (2010), Aditya and Vijayalakshmi (2006), Jaideep *et al.* (2006), Iyengar (2002), Jacob and Tamir (2010), Bercic and Carlisle (2009)]. However, this chapter is only concerned with the methods that aim to achieve k-anonymity through full domain global recoding, hierarchical generalization and minimal suppression. Two of the most popular approaches that meet this specification are Sweeney's Datafly algorithm [Sweeney (1997)] and Samarati's algorithm [Samarati (2001)].

Samarati proposed the concept of the domain generalization hierarchy. The algorithm is based on the axiom that if a node at level h, in domain generalization hierarchy satisfies k-anonymity, then all the levels of height greater than h also satisfy k-anonymity. To exploit this, the algorithm uses a binary search over the levels of the domain generalization hierarchy, i.e., for a domain generalization hierarchy of height h, the search starts at a level of $h/2$ and if that level satisfies the k-anonymity then it searches at the level $h/4$. Otherwise, it searches at the level $3h/4$ and it goes on the same way till it finds the solution in the lowest possible level. This approach assumes that the optimal level is the

one with the least generalization and, within that level, it takes the node that has the minimum information loss as the solution.

Datafly uses a greedy algorithm to search the domain generalization hierarchy; at every step, it chooses the locally optimal move. One drawback with Datafly's approach in common with all such hill climbing type algorithms is that it may be trapped in a local optimum [Cormen *et al.* (2001)].

3.2.1 Problem Complexity

A combination of the values of a set of quasi-identifiers can be viewed as a hierarchical lattice. The k-anonymization algorithms need to search the solution space i.e., the full domain generalization lattice to find the *k-minimal* solution. In this section, we investigate how the size of the lattice (i.e., the number of nodes in the lattice) depends on the number of quasi-identifiers, r, and the height of generalization of each quasi-identifier.

Proposition 3.1 *Let the quasi-identifier set, $QIs = \{A_1, ..., A_r\}$ be the quasi-identifiers for a private table, T with corresponding domain generalized hierarchy, $DGH = \{DGH_{A_1}, ..., DGH_{A_r}\}$ and let $H = \{h_1, ..., h_r\}$ be the corresponding height of domain generalized hierarchy such that $h_i = height(DGH_{A_i})$, for all $1 \leq i \leq r$. Then size of the full domain generalized hierarchy, $DGH_{<A_1,...,A_r>}$ is given by*

$$|DGH_{<A_1,...,A_r>}| = \prod_{1}^{r}(h_i + 1) \tag{3.1}$$

Proof : The problem of finding the number of nodes at level n of the lattice of full domain generalization is the same as the problem of finding the number of solutions to the equation:

$$x_1 + x_2 + ... + x_r = n \tag{3.2}$$

subject to the condition

$$0 \leq x_1 \leq h_1, 0 \leq x_2 \leq h_2, ..., 0 \leq x_r \leq h_r \tag{3.3}$$

So, the number of nodes at level n is equal to the coefficient of x^n in

$$(x^0 + x^1 + ... + x^{h_1}) \times (x^0 + x^1 + ... + x^{h_2}) \times ... \times (x^0 + x^1 + ... + x^{h_r}) \tag{3.4}$$

This is because the number of ways in which sum of r integers in Equation (3.2), subject to the conditions (3.3) equals n is the same as the number of times x^n appears in Equation (3.4). Also, height($\text{DGH}_{<A_1,...,A_r>}$) = $h_1 + h_2 + ... + h_r = h_{max}$, since maximum value attained by n in Equation (3.1) is h_{max}, according to the condition (3.3).

Let

$$(x^0+x^1+...+x^{h_1})\times(x^0+x^1+...+x^{h_2})\times...\times(x^0+x^1+...+x^{h_r}) = C_0x^0+C_1x^1+...+C_{h_{max}}x^{h_{max}} \tag{3.5}$$

Where, C_0, C_1, ... , $C_{h_{max}}$ is the number of nodes in the level 0,1, ... ,h_{max} of the lattice respectively. We are interested to find the total number of nodes in the lattice which will be $C_0 + C_1 + ... + C_{h_{max}}$. So, if in Equation (3.5) we let x=1 we get,

$$C_0 + C_1 + ... + C_{h_{max}} = (h_1 + 1) \times (h_2 + 1) \times ... \times (h_r + 1) = \prod_1^r (h_i + 1) \tag{3.6}$$

■

From Proposition 3.1 it can be shown that the size of full domain generalized hierarchy is exponential on the number of QIs of a database.

Proposition 3.2 *Let $f(h, r)$ denote the number of nodes in $DGH_{<A_1,...,A_r>}$, where QI = {A_1, ..., A_r} are quasi-identifiers for a private table T. Then $f(h, r) = O((h+1)^r)$ where h is the average height of the domain generalized hierarchy of a quasi-identifier in the quasi-identifier set.*

Proof : Let $h = \{h_1, ..., h_r\}$ be the corresponding height of domain generalized hierarchies, DGH = DGH_{A_1}, ..., DGH_{A_r} and let h_{avg} be the average height of the domain

generalized hierarchies, DGH. That means, $h_{avg} = \frac{h_1+h_2+...+h_r}{r}$. The number of nodes in $\text{DGH}_{<A_1,...,A_r>}$ is given by,

$$f(h,r) = (h_1+1) \times (h_2+1) \times ... \times (h_r+1)[From Proposition 3.1] \tag{3.7}$$

and using arithematic mean - geometric mean inequality, we can write the above formula as follows:

$$(\sqrt[r]{(h_1+1) \times (h_2+1) \times ... \times (h_r+1)}) \leq \frac{(h_1+1)+(h_2+1)+...+(h_r+1))}{r} \tag{3.8}$$

Therefore Equation (3.7) can be written as

$$f(h,r) \leq (\frac{(h_1+1)+(h_2+1)+...+(h_r+1)}{r})^r \tag{3.9}$$

$$\leq (h_{avg}+1)^r \tag{3.10}$$

$$= O((h_{avg}+1)^r) \tag{3.11}$$

■

Thus, Proposition 3.2 shows that the solution space is exponential with respect to r (the number of QIs). Given this, as r becomes large, it becomes impractical to exhaustively search the lattice for a *k-minimal* solution. Thus, the lattice should be searched heuristically in a manner that optimizes the tradeoff between computing time and information loss. Samarati's algorithm used binary search to choose levels of the lattice to be searched. In the next few paragraphs, we show that even if binary search is used, Samarati's algorithm is exponential in r. Datafly is polynomial of first order in r, but results in high information loss. Our proposed algorithm is polynomial of second order in r and maintains the best tradeoff between computing time and information loss.

If we assume that QI contains only the attributes having the domain generalized hierarchy of height, h, then, from Equation (3.4), the number of nodes at level n of the full domain generalized hierarchy (FDGH) is the coefficient of x^n in the expansion of $(x^0 + x^1 + ... + x^h)^r$ since, $h_1 = h_2 = ... = h_r = h_{avg} = h$. The expansion of the above expression is given by,

$$(x^0 + x^1 + ... + x^h)^r = \sum_{n=0}^{rh} \left(\begin{array}{c} r \\ n \end{array} \right)_{h+1} x^n \tag{3.12}$$

$\left(\begin{array}{c} r \\ n \end{array} \right)_{h+1}$ denotes the number of integer compositions of the non-negative integer n with r parts $x_1, ..., x_r$ each from the set $\{0, 1, ..., h\}$ and allows the representation,

$$\left(\begin{array}{c} r \\ n \end{array} \right)_{h+1} = \sum_{\substack{r_0+...+r_h=r \\ 0.r_0+1.r_1+...+h.r_h=n}} \left(\begin{array}{c} r \\ r_0, r_1, ..., r_h \end{array} \right) \tag{3.13}$$

where $\left(\begin{array}{c} r \\ r_0, r_1, ..., r_h \end{array} \right)$ is a multinomial coefficient, defined as $\frac{r!}{r_0! r_1! ... r_h!}$, for non-negative integers $r_0, ..., r_h$. We can verify the Equation (3.13) by noting that for real numbers $y_0, ..., y_h$, the following holds [Weisstein (2009)].

$$(y_0 + y_1 + ... + y_h)^r = \sum_{\substack{r_0,...,r_h \geq 0 \\ r_0+...+r_h=r}} \left(\begin{array}{c} r \\ r_0, r_1, ..., r_h \end{array} \right) y_0^{r_0} ... y_h^{r_h} \tag{3.14}$$

Then setting $y_i = x^i$ for all $0 \leq i \leq h$, we get

$$(x^0 + x^1 + ... + x^h)^r = \sum_{\substack{r_0,...,r_h \geq 0 \\ r_0+...+r_h=r}} \left(\begin{array}{c} r \\ r_0, r_1, ..., r_h \end{array} \right) x^{0.r_0} + ... x^{h.r_h} \tag{3.15}$$

Comparing coefficients of the right-hand sides of Equations (3.12) and (3.15), we get (3.13). Now we prove that central coefficient $\left(\begin{array}{c} r \\ \lfloor \frac{h_r}{2} \rfloor \end{array} \right)_{h+1}$ is the largest from the remaining coefficient in the expansion of $(x^0 + x^1 + ... + x^h)^r$. This means that the

number of nodes in the middle level of the FDGH is the highest among all levels.

Lemma 3.1 *Let $r \geq 0$ and $h \geq 0$ be integers. For all integers n such that $0 \geq n \geq hr$,*

$$\left(\begin{array}{c} r \\ n \end{array} \right)_{h+1} \leq \left(\begin{array}{c} r \\ \lfloor \frac{hr}{2} \rfloor \end{array} \right)_{h+1} \tag{3.16}$$

The proof can be found in [Steffen (2012a)].

Now we investigate the lower and the upper bound of the central coefficient.

Lemma 3.2 *In the expansion of expression $(x^0 + x^1 + ... + x^h)^r$, the central coefficient $\left(\begin{array}{c} r \\ \lfloor \frac{hr}{2} \rfloor \end{array} \right)_{h+1}$ satisfies*

$$\frac{(h+1)^r}{hr+1} \leq \left(\begin{array}{c} r \\ \lfloor \frac{hr}{2} \rfloor \end{array} \right)_{h+1} \leq (h+1)^r \tag{3.17}$$

Proof : The sum of all coefficients in the expansion of the expression is $(h+1)^r$, which can be obtained by setting $h_1 = h_2 = ... = h_r = h$ in Proposition 3.1. The central coefficient is the largest amongst all coefficients, which is shown in Lemma 3.1. Since the total number of terms in the expansion of the expression is $(hr+1)$, we get the lower bound of $\frac{(h+1)^r}{hr+1}$ and the upper bound of $(h+1)^r$. Thus, $\frac{(h+1)^r}{hr+1} \leq \left(\begin{array}{c} r \\ \lfloor \frac{hr}{2} \rfloor \end{array} \right)_{h+1} \leq (h+1)^r$ ■

Recently, Steffen has found a sharper approximation of the central binomial coefficient by generalizing the famous Stirling's approximation to the central binomial coefficient [Steffen (2012b)]. In the following lemma, we write $a_k \sim b_k$ as a short-hand for $\lim_{k\to\infty} \frac{a_k}{b_k} = 1$.

Lemma 3.3 *For all fixed h,*

$$\left(\begin{array}{c} r \\ \lfloor \frac{hr}{2} \rfloor \end{array} \right)_{h+1} \sim \frac{(h+1)^r}{\sqrt{2\pi r \frac{(h+1)^2-1}{12}}} \tag{3.18}$$

The proof can be found in [Steffen (2012b)].

3.3 Proposed Method for *k*-anonymization

Sweeney's Datafly starts checking from the lowest level of the domain generalization hierarchy and checks for the k-anonymity. If it is not satisfied then it generalizes the most differentiated attribute, stepping into the next level of hierarchy and repeating until it is satisfied that k-anonymity has been achieved. It checks only one attribute in each level, which guarantees that a solution will be found in polynomial time, but not necessarily the optimal one.

Analyzing the algorithms (Samarati's and the Incognito algorithms), we observe that both give theoretically optimal solutions but not in polynomial time, whereas the Datafly algorithm finds a solution in a polynomial time but is not necessarily optimal.

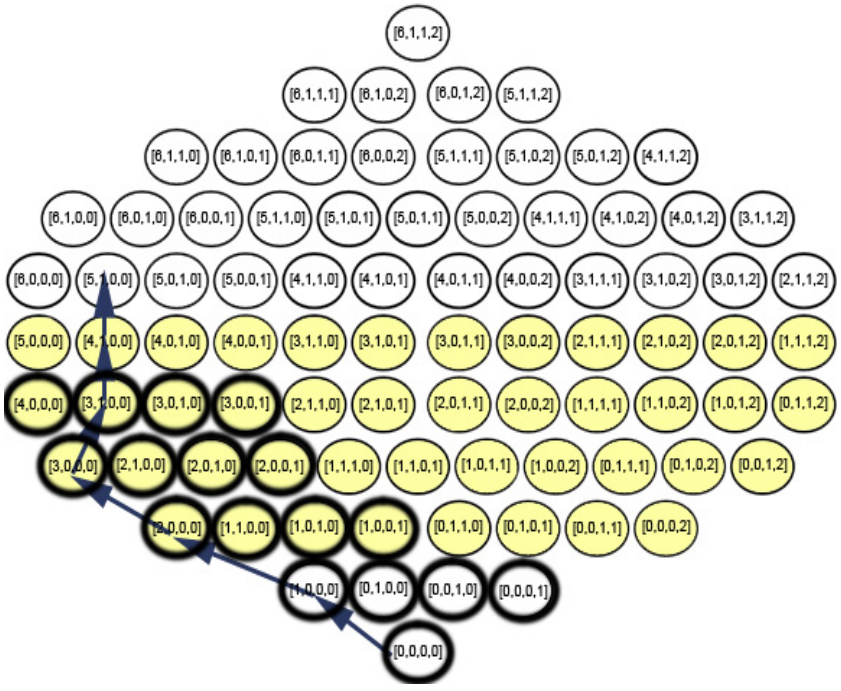

Figure 3.1: A visualization of the three processing strategies: the circles with yellow fillings represent Samarati's method; the arrow represents the strategy for Datafly and the dark circles represent the improved greedy method.

The greedy heuristic that Datafly uses simply generalizes the most differentiated

attribute. Here, we propose an improved greedy heuristic which gives better solutions than the heuristics of the Datafly algorithm. The algorithm starts by checking for k-anonymity from the lowest level. If it is not satisfied, the k-anonymity is checked by generalizing one attribute at a time simultaneously implementing the minimum suppression. If, by generalizing an attribute, the k-anonymity is satisfied, the attribute is generalized and declared as the solution to the first phase. If after generalizing all the attributes, none of them satisfies k-anonymity, the case that is the closest to k-anonymity is selected to be generalized. In case of ties the most differentiated attribute is selected. This process continues until a solution is found. In Figure 3.3, the circles with yellow fillings represent Samarati's method, the arrow represents the strategy for Datafly and the dark circles represent the improved greedy method.

Theorem 3.1 *The number of nodes in the full domain generalized hierarchy that needs to be checked for k-anonymity in Samarati's algorithm is exponential in the number of quasi-identifiers.*

Proof Samarati's algorithm uses binary search on the levels of the lattice with the nodes in the middle level being searched first. For simplicity we assume $h_1 = h_2 = ... = h_r = h$, then the number of nodes in the middle level of the lattice is given by the central coefficient $\begin{pmatrix} r \\ \lfloor \frac{h_r}{2} \rfloor \end{pmatrix}_{h+1}$, where $QI = \{A_1, ..., A_r\}$ is the quasi-identifiers and $H = \{h_1, ..., h_r\}$ is the corresponding height of domain generalized hierarchy. In Lemma 3.2 and Lemma 3.3 we have shown that the central coefficient is exponential in r. Thus, searching only middle level of the lattice has lower bound $\frac{(h+1)^r}{(hr+1)}$ and the upper bound $(h+1)^r$.

■

Therefore, as the value r becomes large, applying binary search on the levels of the lattice becomes increasingly less effective in reducing the number of potential nodes that need to be checked for k-anonymity.

Theorem 3.2 *The number of nodes in the full domain generalized hierarchy that needs to be checked for k-anonymity in the proposed greedy algorithm is polynomial of second*

order in the number of quasi-identifiers and Datafly algorithm is polynomial of first order in the number of quasi-identifiers.

Proof Let $QI = \{A_1, ..., A_r\}$ be the quasi-identifiers and $H = \{h_1, ..., h_r\}$ be the corresponding height of domain generalized hierarchy. The proposed algorithm starts checking nodes from the lowest level, 0 of the full domain generalized hierarchy, $\text{DGH}_{<A_1,...,A_r>}$. In the worst case (at level h_i), r possible generalizations are checked for k-anonymity from which the generalization closest to k-anonymity is selected. This worst case value is obtained by generalizing the attributes from set QI one at a time from the selected node closest to k-anonymity at level h_{i-1}. Thus in the worst case, the algorithm goes to the highest level, h_{max} of the hierarchy, $\text{DGH}_{<A_1,...,A_r>}$ and the number of nodes searched in our proposed algorithm is given by $O(r \times h_{max})$. In Proposition 3.1, we indicated that highest level, h_{max} is given by $h_{max} = h_1 + h_2 + ... + h_r$. In Proposition 3.2, we indicated that $h_{avg} = \frac{h_1+h_2+...+h_r}{r}$. The number of nodes searched is given by $O(r^2 \times h_{avg})$. If we assume $h_1 = h_2 = ... = h_r = h$, it becomes $O(r^2 \times h)$. Thus, the proposed algorithm is polynomial of second order in the number of quasi-identifiers.

The Datafly algorithm checks only one generalization at level h_i of the hierarchy $\text{DGH}_{<A_1,...,A_r>}$. In the worst case, the algorithm goes to the highest level, h_{max}. So, the number of nodes searched by Datafly algorithm is given by, $O(h_{max}) = O(r \times h_{avg})$. If we assume $h_1 = h_2 = ... = h_r = h$, it becomes $O(r \times h)$ which means the algorithm is polynomial of first order in the number of quasi-identifiers. ■

The details of the proposed algorithm are given in Algorithm 3.1. It starts by constructing a full domain generalisation hierarchy. The algorithm then starts by checking for k-anonymity in the lowest level of the full domain generalization hierarchy which has only one node *i.e.*, the node which has all the quasi-identifiers without any generalization. If k-anonymity is satisfied, then the dataset is k-anonymous as it is and needs no further anonymization. But if k-anonymity is not satisfied, then k-anonymity is checked by generalizing one attribute at a time. If none of the nodes at a given level of the full domain generalization hierarchies satisfies k-anonymity, then the node that is closest

Algorithm 3.1 *Improved_Greedy*($\mathcal{T}, \mathcal{QI}$)

{ $\mathcal{T}$: A table to be k-anonymized, $\mathcal{QI}$: a set of n quasi-identifier attributes }
n: number of quasi-identifiers
k: desired anonymity
m: suppression limit
node: array of n integers
array_node: array of nodes
array_anonymity: array of integers
array_index: array of integers
dtree: two dimensional array /* each row corresponding to a level of DGH and column to the node in the particular level */
Construct the domain generalised hierarchy using Samarati's algorithm for the given tables
node ← array of n zeros
temp ← anonymity(node)
if $temp \geq k$ **then**
  return *node* /* T is anonymous as it is given */
end if
while $temp < k$ **do**
  array_node ← array of nodes obtained by generalizing each quasi-identifier at a time
  array_anonymity ← empty array
  for each level in array_levels append anonymity(node) to array_anonymity
  array_index ← max(array_anonymity)
  if $len(array\_index) = 1$ **then**
    node ← array_node[array_index[0]]
    temp ← array_anonymity[array_index[0]]
  else
    index ← diverse(array_index
    node ← array_node[index]
    temp ← array_anonymity[index]
  end if
end while
return *node*
Output $\mathcal{D}\infty$

to k-anonymity is selected (in case of tie, the node which has generalized the most differentiated attributes is selected) and k-anonymity is again checked by generalizing one attribute at a time in the selected node. The previous step is repeated until a node is found which is k-anonymous. Theorem 3.2 shows that the complexity of the proposed algorithm is polynomial.

Four functions are used in Algorithm 3.1. The function anonymity() takes the node in the domain generalization hierarchy as input and outputs the anonymity of the particular node using suppression limit m. The function max() takes an integer array as input and outputs the array of indices of all the integers with the largest value in the input array. The function len() takes an array as an input and outputs the number of elements in the input array. Finally, diverse() is a function which takes an array of integers as an input. It considers the nodes in array_node with indices in array_index and outputs the index of the node which generalizes the most diverse attribute (*i.e.*, the attribute with the most distinct values as used in Datafly algorithm).

When applying disclosure control to any data product, we need to be mindful of the impact on the utility of that product. If privacy were our only concern, then utility would tend to become zero. In the literature, damage to utility is most frequently operationalized as information loss of which there are many different definitions. According to Domingo Ferrer, information loss can be obtained by comparing the original data to the masked one, the closer the data is to the original, the lower is the information loss [Domingo-Ferrer and Vicent (2003)]. Xu *et al*. gave a similar definition stating that the information loss measures "how well the generalized tuples approximate the original ones" [Xu *et al*. (2006)]. A general-purpose metric that is considered here is *iloss*.

Iloss, proposed by Xiao and Tao, is a metric to find the information loss of transforming a value to a more general one [Xiao and Tao (2006)]. It assumes that all raw values are at the leaves of the taxonomic tree. The formula to calculate *iloss* is given in the following equations [Benjamin *et al*. (2011)]. Equation (3.19) computes the information loss of a domain value when generalized. Equation (3.20) computes the information loss for a tuple level and Equation (3.21) computes information loss for the

whole table.

$$Iloss(V_g) = \frac{|V_g| - 1}{|D_A|} \tag{3.19}$$

$$Iloss(r) = \sum_{V_g \in r} (w_i \times Iloss(V_g)) \tag{3.20}$$

$$Iloss(T) = \sum_{r \in T} Iloss(r) \tag{3.21}$$

where,

$|V_g|$ is the number of domain values that are descendants of V_g,

$|D_A|$ is the number of domain values in the attribute A of V_g and

w_i is a positive constant specifying the penalty weight of attribute A_i of V_g

3.4 Results and Discussion

The three algorithms were tested on two datasets obtained from UCI Machine Learning Repository: the adult dataset [Adult Dataset (1996)] and CUPS dataset [CUPS (1998)]. The adult dataset has 32,561 records and 15 attributes of which eight attributes (namely Age (3), Profession (2), Education (2), Marital status (2), Position (2), Race (1), Sex (1), Country (3)) were considered to be quasi-identifiers[1]. The CUPS dataset has 96367 records and 479 attributes of which, five attributes (namely Age (4), Gender (3), Income (4), Cluster (4) and Domain (3)) were considered to be quasi-identifiers. While we endeavored to maximize the number of attributes that we used as quasi-identifiers, the data quality of the remaining attributes was poor. So, they were discarded. However, the results presented here are not dependent on the particular attributes chosen. Similar patterns of results can be obtained regardless of the cross classification of quasi-identifiers.

Comparing the time complexities of the algorithms, Samarati's algorithm has an exponential time complexity (shown in Theorem 3.1) and both the greedy approaches discussed have polynomial time complexities (shown in Theorem 3.2)). From the time

[1]The numbers in the brackets indicate the maximum height in the hierarchy.

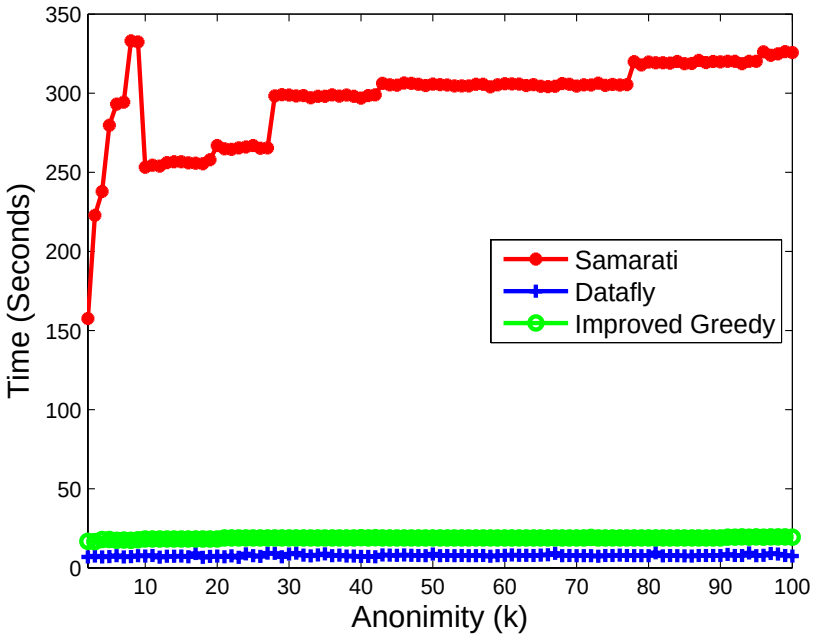

Figure 3.2: Anonymity *vs* time for Adult dataset.

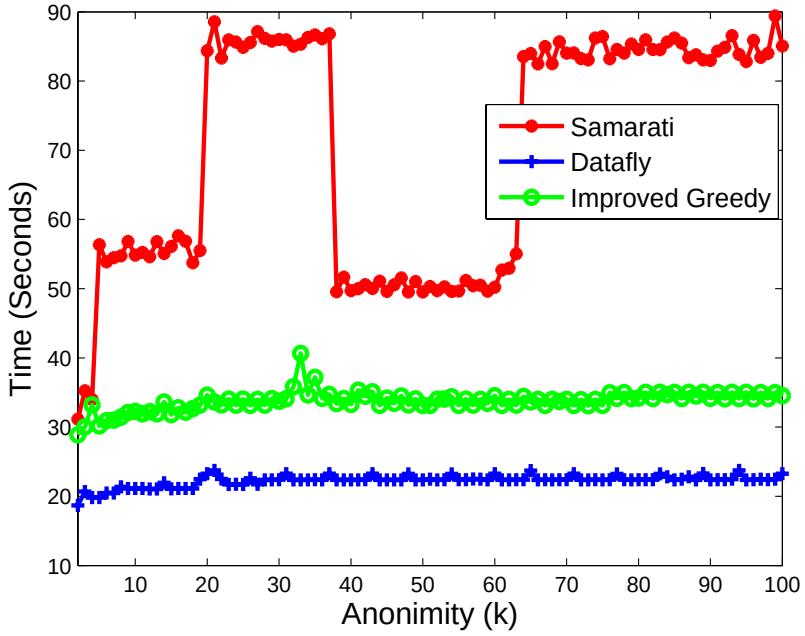

Figure 3.3: Anonymity *vs* time for CUPS dataset.

complexities, it is obvious that Datafly will have the minimum time of execution followed by proposed improved greedy algorithm. Figures 3.2 and 3.3 compare the execution times of the Datafly, Samarati and improved greedy algorithms for the two datasets. It can be observed from Figures 3.2 and 3.3 that the execution time of Samarati's algorithm is the highest. The execution time of Samarati's approach varies non-monotonically. This is because the execution time of Samarati's algorithm also depends on the order in which the quasi-identifier is arranged in the domain generalization hierarchy. The execution times of the proposed and Datafly algorithms are found to be almost the same with Datafly being the faster of the two.

The information loss of the k-anonymized datasets for the three algorithms are plotted in Figures 3.4 and 3.5. It can be observed that Samarati's algorithm has the lowest information loss. This is because Samarati's algorithm searches for the optimal solution in the hierarchy using brute force. So, the ideal would be to achieve the same information loss as Samarati's algorithm but with lower time complexity. The Datafly algorithm, having least time complexity, suffers from very high information loss and the loss is unacceptable if the value of k is large. The improved greedy algorithm has time complexity comparable to Datafly and iloss values that are comparable to those of Samarati's algorithm.

Figures 3.6 and 3.7 shows the level of generalization at which a given level of k-anonymity was achieved for the three algorithms. The level of generalization for the improved greedy algorithm is similar to the Samarati's algorithm for the Adult dataset although slightly inferior for the CUPS dataset. In both cases the improved greedy algorithm performs markedly better than Datafly. The improved greedy algorithm has similar information loss to Samarati's algorithm so it may seem surprising that it performs slightly worse in terms of the level of generalization. However, as Samarati's algorithm uses brute force method to find the solution in the lowest possible level, it optimizes with respect to level of generalization metric only. But the information loss within a level of a full domain generalization hierarchy is not the same for all the nodes. The improved greedy heuristic gives more emphasis on choosing the nodes *within a level*. So, the nodes chosen *at a given level* by our algorithm have, on average, less information loss than those

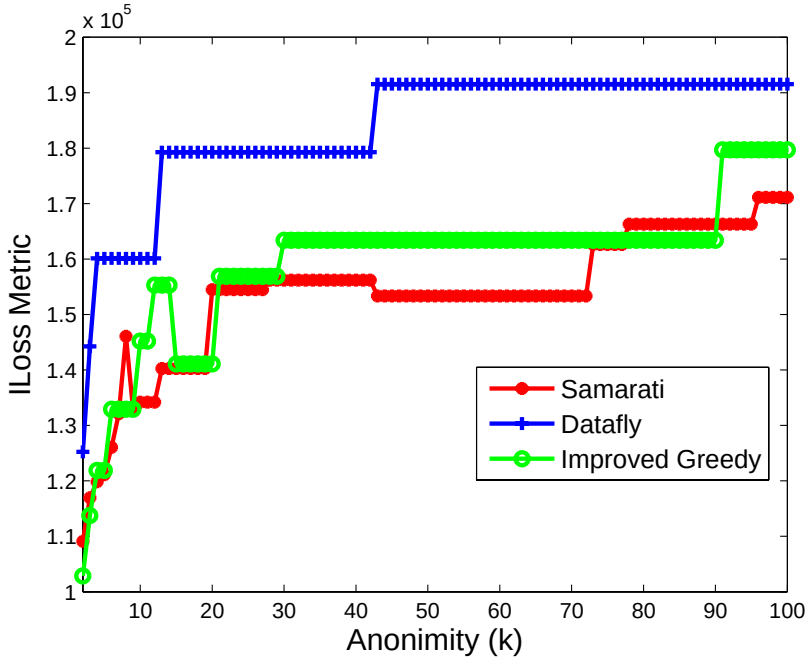

Figure 3.4: Anonymity *vs* iloss for Adult dataset.

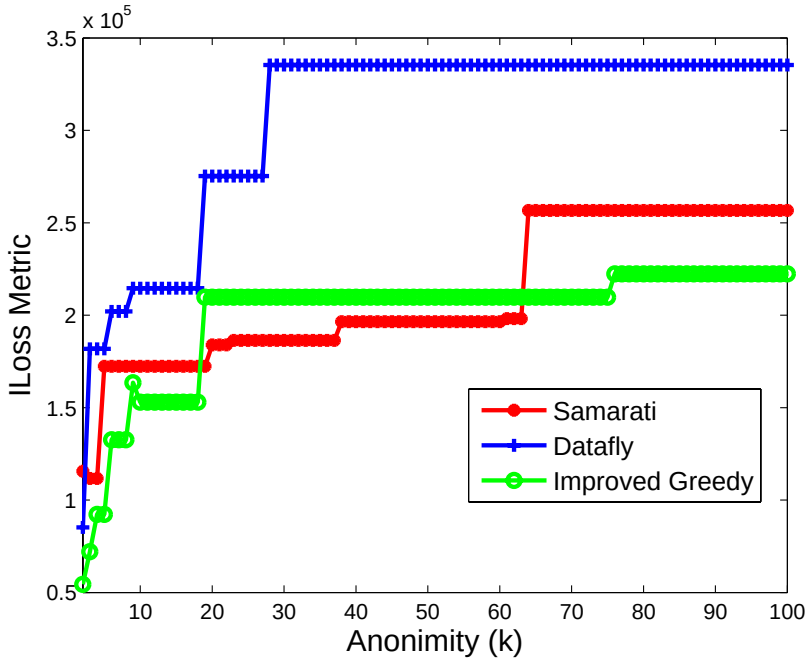

Figure 3.5: Anonymity *vs* iloss for CUPS dataset.

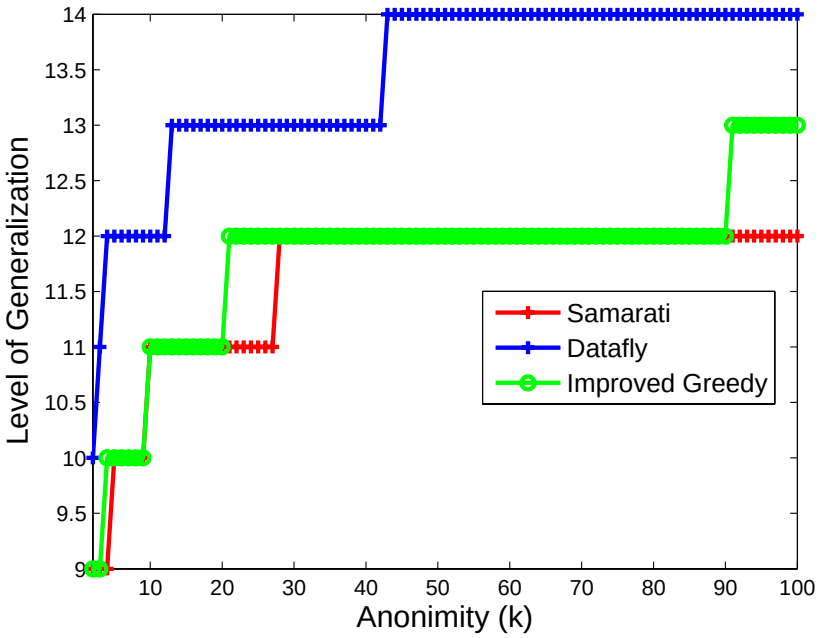

Figure 3.6: Anonymity *vs* level of generalization for Adult dataset.

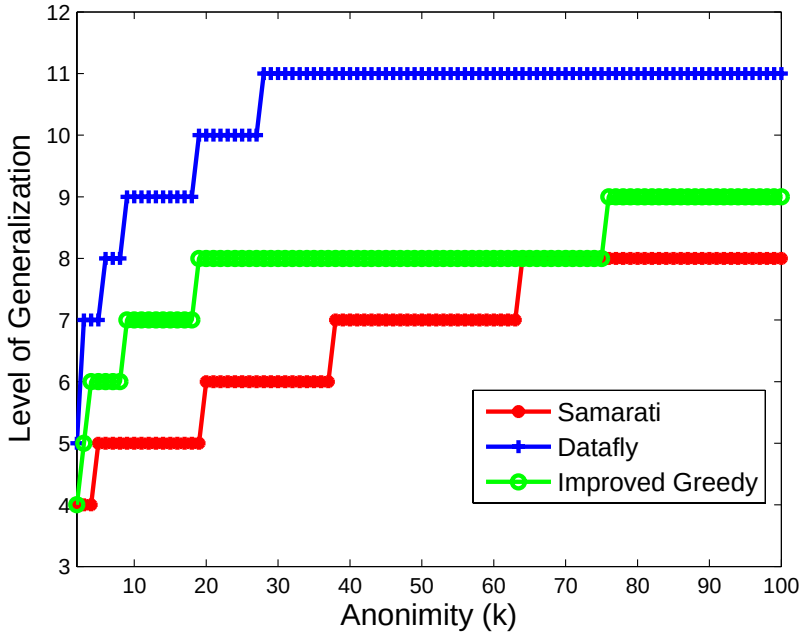

Figure 3.7: Anonymity *vs* level of generalization for CUPS dataset.

chosen by Samarati's algorithm but when we consider only the level of generalization the Samarati's algorithm performs better. Overall the information loss of Samarati's algorithm and proposed algorithm is comparable.

Figure 3.8 shows the execution times of three algorithms with varying the number of quasi-identifiers from three to eight for $k = 10$. This analysis was done for adult dataset. We can infer that the execution time of Samarati's approach was highest as expected. Its execution time is found to increase exponentially with the increase in the quasi-identifiers.

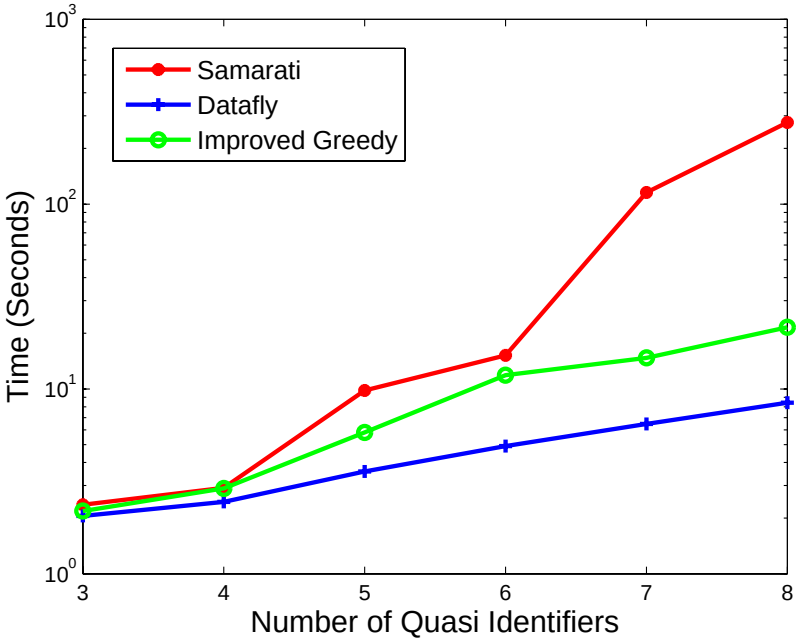

Figure 3.8: Quasi-identifier *vs* time for Adult dataset for k=10.

3.5 Observations

Datafly algorithm takes less time but huge information loss. Samarati's Algorithm takes more time and less information loss. The improved greedy heuristic consumes less time than Samarati and less amount of information loss than Datafly. The improved greedy algorithm maintains a balance between computing time and information loss and privacy.

3.6 Conclusion

Before the release of any data product, there is a need to consider a balance between utility and privacy. In this chapter, we have proposed an improved greedy heuristic for k-anonymization that maintains a better balance between privacy and information loss than other similar algorithms. Metrics like iloss, computing time and level of generalization were used to compare the new algorithm with the established Datafly and Samarati's algorithms. On balance, the new greedy algorithm performed better than the established ones.

Chapter 4

Determining k for k-anonymity

4.1 Introduction

Publishing anonymized data need to ensure certain degree of utility. When utility is guaranteed then privacy is hampered and vice versa. There is a need for balancing between utility and privacy of the data. This chapter deals with experimental study of balancing for utility and privacy when data is released and finding the best value of k for k-anonymization.

4.2 Related Work

Anonymization of the data is to be done so that individual information cannot be inferred. The first of the kind of anonymization algorithm proposed for preserving privacy while publishing data is k-anonymity [Sweeney (2002a)]. The algorithm only converts a dataset to k-anonymized dataset if the input k is provided. The direction of finding the best value of k for a dataset is not explored. First work in that direction is given in [Jun and Meng (2008)]. The algorithm is called as one pass k-means algorithm (OKA). This algorithm was proposed by Jun and Meng in 2008. It is derived from the standard K-means algorithm

and runs in one iteration. This algorithm has two stages first is the clustering stage and second is the adjustment stage.

Clustering Stage: Clustering stage proceeds by sorting all the records and then randomly picking N records as seeds to build clusters. Then for each record r remaining in the dataset, algorithm checks to find a cluster, C for which this record is the closest and assigns the record to the cluster and updates its centroid. The difference between the traditional K-means algorithm and OKA is that in OKA whenever a record is added to the cluster its centroid is updated. This improves the assignments in future because the centroid represents the real centre of the cluster. In OKA the records are first sorted according to the quasi-identifiers thus making sure that similar tuples are assigned to the same cluster.

Adjustment Stage: In the clustering stage the clusters that are formed can contain more than k tuples and there can be some clusters containing less than k tuples. Therefore, when these clusters are anonymized it will not satisfy the condition for k-anonymity. These clusters need to be resized to contain at least k tuples. The goal of this adjustment stage is to make the clusters contain at least k records while minimizing the information loss. This algorithm first removes the extra tuples from the clusters and then assigns those tuples to the clusters having less than k tuples. The removed tuples are farthest from the centroid of the cluster and while assigning the tuples to the clusters it checks the cluster which is closest to the tuple before assigning it, thus minimizing the information loss. If no cluster contains less than k tuples and some records are left they are assigned to this respective closest clusters.

4.3 Proposed Approach

Organizations cannot refrain from publishing information of the stakeholders. They need to publish information by the guidelines of the privacy policy. If the privacy policy is permitting to publish the information, then they need to be anonymized beforehand. The most practical method is the $k-$anonymity model. Choosing the value of k is a difficult task because it is dataset dependant. The proposed method has two steps. First step is

to construct the clusters and the second step is to run the clusters through the proposed algorithm (Algorithm 4.1.). With the availability of the quasi-identifiers in the dataset, attacks may be possible on $k-$anonymity. Quasi-identifiers are covert channels for the attacks on $k-$anonymity. It is advisable to make the quasi-identifiers as generalized as possible. The following approach is proposed for lessening the attacks. Once the table T is organized into clusters with k tuples using OKA, apply generalization hierarchy of quasi-identifier on the clusters to form a privacy preserving $k-$anonymized table. This algorithm uses the output of OKA. The generalization hierarchy constructed should be complete and map all possible values of the attribute to a single value.

Algorithm 4.1 Enhanced_Privacy ($\mathcal{P}, \mathcal{DGH}$)

1: { $\mathcal{P}$: an adjusted partitioning ($P1toPk$)of table $\mathcal{T}$ and $\mathcal{DGH}$: Domain Generalization Hierarchy }
2: QI: Quasi-identifier
3: P_i: Partition i of $\mathcal{T}$
4: **for** each partition P_i of $\mathcal{T}$ **do**
5: **for** each QI in P_i **do**
6: **if** attribute values for partition Pi are not same **then**
7: use Generalization hierarchy to generalize
8: **end if**
9: **end for**
10: **end for**
11: Exit
12: Output: Privacy enhanced k-anonymized table $\mathcal{T}'$

The parameters to check utility and privacy depend from data to data. To determine the extent of privacy preserved by the dataset, we counted the number of attributes whose values are completely suppressed. The percentage of privacy can be computed with the Equation (4.1).

$$Percentage\ of\ Privacy\ =\ \frac{Number\ of\ Supressed\ Values}{Number\ of\ quasi-identifiers} \times 100 \qquad (4.1)$$

To determine the utility of the dataset, Decision stump algorithm for classification is used. Decision stump is a machine learning model consisting of a single-level decision tree

with a categorical or numeric class label. The results produced by WEKA tool [Mark *et al.* (2009)] show percentage of tuples that can be correctly classified using the algorithm. We have considered the average of all the percentages obtained from using various attributes in the Decision stump algorithm for classfication of correct instances.

4.4 Results and Discussion

The experiment was conducted on Adult dataset from UCI machine learning repository which comprise of 45,222 instances with known values. It contains numerical as well as categorical attributes. It contains 15 attributes with its properties. Clustering of the dataset is done using WEKA. The K-means clustering algorithm was deployed for clustering. Out of the total 15 attributes of the dataset only 5 (age, education, native-country, race and workclass) were considered as quasi-identifiers and the rest as sensitive attributes. Generalization was performed on the clustered dataset. These generalization hierarchies are used for anonymizing evenly clustered data. For age which is a numerical attribute, mean of all the tuple values was taken. The other attributes are generalized as shown in the Section 2.2.3 of Chapter 2.

4.4.1 Experiment 1

In the first experiment six attributes (age, education, marital status, occupation, race and native-country) were considered for our analysis. Randomly 1000 tuples from the dataset are selected for anonymization to determine how utility varies with privacy. Age, education, race and country are considered as quasi-identifiers and other two as sensitive attributes. Clusters were formed using WEKA tool according to the value of k. As described in Section 4.2, the clusters produced may contain less than k tuples, thus an adjustment was done so that each cluster contains at least k tuples. After adjusting the clusters, $k-$anonymization is performed based on the generalization hierarchy. The k-anonymization algorithm based on OKA was run to generalize the adjusted clusters.

For evaluating utility, we performed the classification mining on the k-anonymized

dataset. Classification was performed by using WEKA tool considering native-country as classification variable. Percentage of correctly classified tuples as the utility of the dataset was taken for various values of k. Privacy was calculated by counting the number of tuples which are generalized to highest level of generalization. Privacy and utility was calculated by varying the value of k. The balancing point between utility and privacy is the point where privacy and utility curves intersect or tend to converge. Figure 4.1 shows the variation of utility and privacy with k. It clearly follows from the Figure 4.1 that on increasing the value of k privacy provided by the dataset increases but utility decreases. For this sample dataset the balancing point comes between k=8 and k=9, and utility of the dataset at balancing point is around 60%.

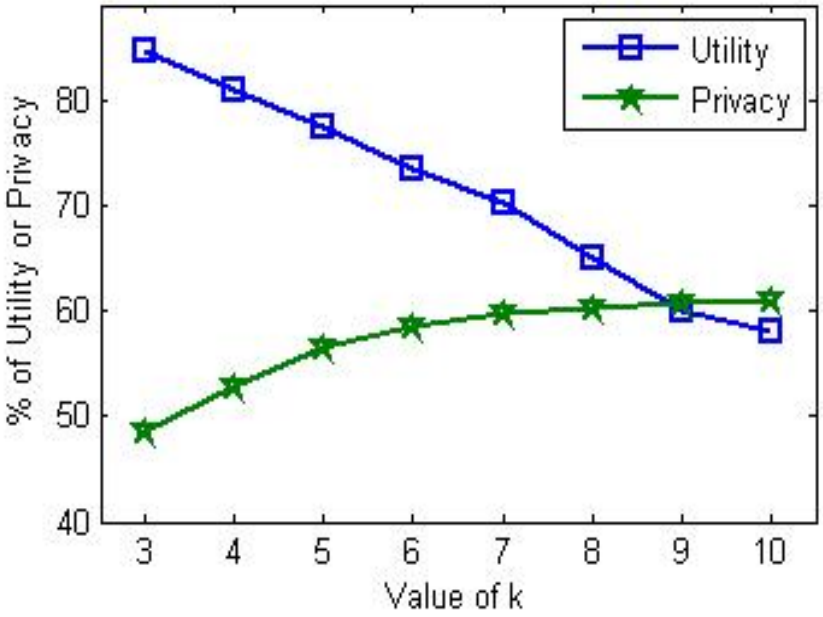

Figure 4.1: Variation of utility and privacy with anonymization.

4.4.2 Experiment 2

In the second experiment all 15 attributes were considered for analysis, to study the effect of more number of attributes on the privacy and the utility of the k-anonymized dataset. Again randomly selected 1000 tuples from the dataset for anonymization to determine

how utility varies with privacy. Age, work class, education, race and native-country are considered as quasi-identifiers and all other attributes as sensitive attributes. Similar steps were followed as in experiment 1 to study the variation of utility and privacy on varying k value. Figure 4.2 shows the variation of utility and privacy with k. For this sample dataset the balancing point comes between k=11 and k=12, and utility of the dataset at balancing point is around 52%. Thus on increasing the number of quasi-identifiers considered for analysis the balancing point falls down and values of k at which balance is achieved increases.

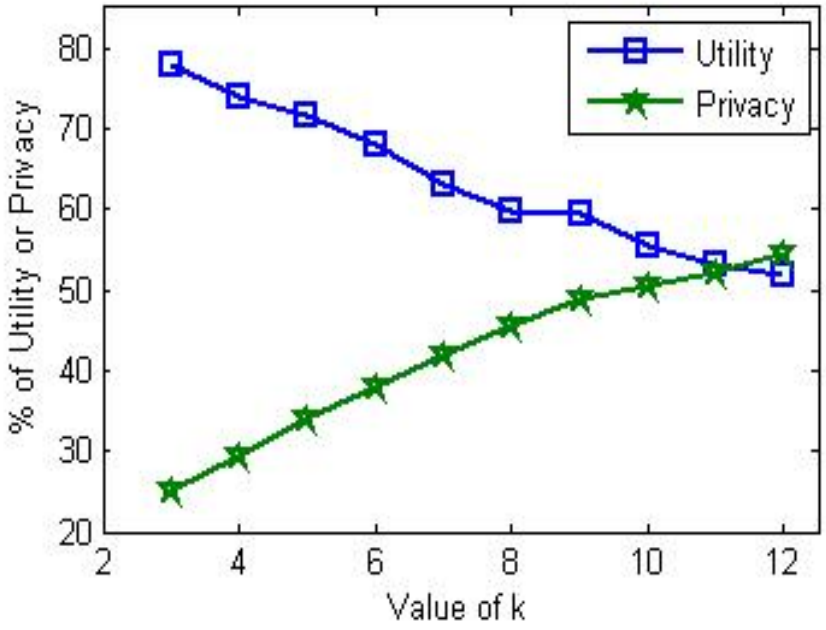

Figure 4.2: Variation of utility and privacy with anonymization.

4.4.3 Experiment 3

In this experiment 3000 tuples were taken from the adult dataset considering all 15 attributes, to study the effect of more number of tuples on the privacy and the utility of the k-anonymized dataset. Age, work class, education, race and native-country are considered as quasi-identifiers and all other attributes as sensitive attributes. Similar steps were followed as in Experiment 1 to study the variation of utility and privacy on

varying k value and shown in Figure 4.3. For this sample dataset the balancing point comes between k=10 and k=11, and utility of the dataset at balancing point is around 50%.

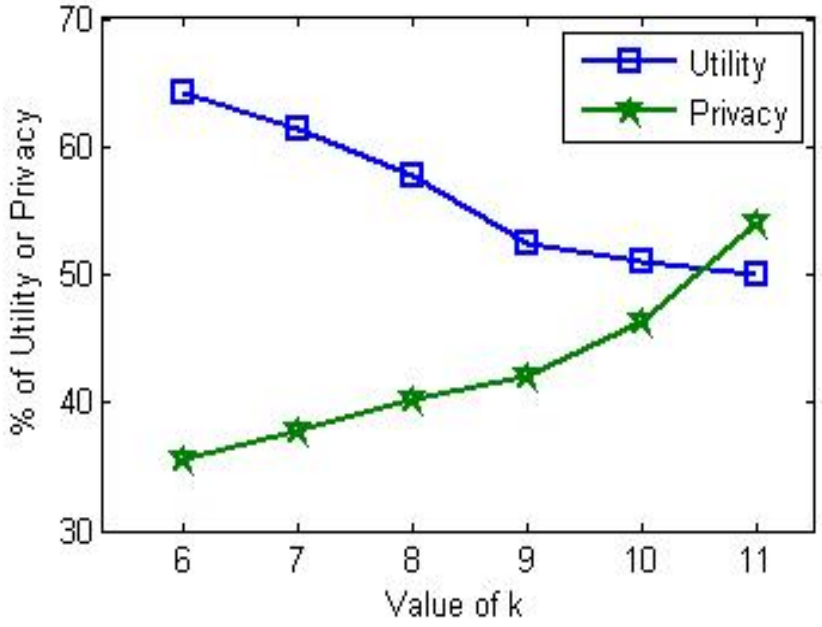

Figure 4.3: Variation of utility and privacy with anonymization.

4.5 Conclusion

There is a tradeoff between privacy and utility. On conducting the experiments we found that the balancing point between utility and privacy depends on the dataset and value of k cannot be generalized for all datasets such that utility and privacy are balanced. On varying the number of sensitive attributes in a dataset the balancing point varies. If the number of quasi-identifiers increases balancing point moves down and balance between utility and privacy occurs at a higher value of k. Thus if a dataset contains more number of quasi-identifiers then the utility as well as privacy attained at balancing point will be less than the dataset having fewer quasi-identifiers. The number of tuples in the dataset has an effect on the balancing point. Increase in the number of tuple slightly shifts the balancing point and the value of k for which balancing occurs. Thus an approximate

prediction can be made on the balancing point for huge dataset by conducting experiment on a sample dataset.

Chapter 5

Enhanced *t*-closeness for Publishing Nominal Data

5.1 Introduction

It has been discussed in previous chapters that publishing data in the public domains by removing sensitive attributes do not guarantee privacy. Even if one removes the attributes like SSN, name, etc., which are used to identify an individual, sensitive data can be revealed using linking attacks. These threats make the individuals hide their information or provide tampered information to protect their privacy.

To publish data with proper limitation of disclosures, Sweeney introduced k-anonymity [Sweeney (2002a)]. This model states that each equivalence class should contain at least k records with respect to the quasi-identifiers. While k-anonymity protects from identity disclosure, it does not provide sufficient protection against attribute disclosure. This has been recognized by several authors [Aggarwal *et al.* (2005), Ninghui *et al.* (2007)]. Two attacks namely, homogeneity attack and the background knowledge attack were identified. To overcome these attacks, the l-diversity privacy model was introduced [Machanavajjhala *et al.* (2007), Ninghui *et al.* (2007)].

The l-diversity principle overcomes the lack of diversity in the k-anonymous table for the sensitive attributes. It protects from attribute disclosure. The principle of l-diversity

Table 5.1: Example of original table.

#	**Zipcode**	**Age**	**Salary**	**Disease**
1	47677	29	3K	gastric ulcer
2	47602	22	4K	gastritis
3	47678	27	5K	stomach cancer
4	47905	43	6K	gastritis
5	47909	52	11K	flu
6	47906	47	8K	bronchitis
7	47605	30	7K	bronchitis
8	47673	36	9K	pneumonia
9	47607	32	10K	stomach cancer

Table 5.2: Example of 3-diverse table.

#	**Zipcode**	**Age**	**Salary**	**Disease**
1	476**	2*	3K	gastric ulcer
2	476**	2*	4K	gastritis
3	476**	2*	5K	stomach cancer
4	4790*	≥ 40	6K	gastritis
5	4790*	≥ 40	11K	flu
6	4790*	≥ 40	8K	bronchitis
7	476**	3*	7K	bronchitis
8	476**	3*	9K	pneumonia
9	476**	3*	10K	stomach cancer

states that a table is said to be *l*-diversity if every equivalence class of the table has at least *l*-diversified values for sensitive attributes. Table 5.1 shows an original table and Table 5.2 shows a *3*-diverse table. The *l*-diversity is insufficient to prevent attribute disclosure. Two attacks are possible on *l*-diversity. One of the attacks is skewness attack. When the overall distribution is skewed, satisfying *l*-diversity does not prevent attribute disclosure. The other possible attack is similarity attack. When the sensitive attribute values in an equivalence class are distinct but semantically similar, an adversary can learn important information. The distributions that have the same level of diversity may provide very different levels of privacy. This is because there are semantic relationships among the attribute values, as different values have very different levels of sensitivity and privacy is affected by the relationship with the overall distribution. To address the limitations of *k*-anonymity and *l*-diversity, *t*-closeness was introduced [Ninghui (2007)].

The *t*-closeness principle states that the distance between the distribution of a sensitive attribute in a class and the distribution of the attribute in the whole table is not more than a threshold *t*.

Section 5.2 gives related work of *t*-closeness. We discuss about *t*-closeness, (*n,t*)-closeness and the distance measure used. Achieving *t*-closeness for preserving privacy without downgrading the utility is shown in Section 5.3. Identity disclosure can reveal sensitive values. Attribute disclosure can occur with or without identity disclosure. It has been recognized that even disclosure of false attribute information may cause harm [Lambert (1993)]. So identity disclosure is a serious concern in privacy and can often cause attribute disclosure. Section 5.4 uses a data reconstruction approach for dealing identity disclosure. Section 5.5 gives the conclusions.

5.2 Related Work

Privacy is measured by the information gain of an observer. Before seeing the released table, the observer has a prior belief about the sensitive attribute value. After seeing the released table, the observer has a posterior belief about the sensitive attribute value. So information gain computed as the following equation:

$$Information\ gain = posterior\ belief - prior\ belief \tag{5.1}$$

Prior belief is influenced by the distribution of the sensitive attribute value in the whole table, Q. Posterior belief is influenced by the distribution of the sensitive attribute value in the equivalence class, P. So, information gain is the difference between the two distribution , P and Q. Let it be denoted as (D[P,Q]).

An equivalence class has *t*-closeness if D[P,Q] $\leq$ *t*. A table has *t*-closeness if all equivalence classes have *t*-closeness. For a table, if D[P,Q] is less, then information gained by the observer is less and there is less privacy risk and vice versa. The distance metric used in *t*-closeness is the earth mover distance [Ninghui (2007), Ninghui *et al.* (2007)]. This is the minimal amount of work needed to transform one distribution to another by

moving the distribution mass. It can be represented by the following equation:

$$D[P,Q] = WORK(P,Q,F) = \sum_{i=1}^{m}\sum_{j=1}^{m} d_{ij} \times f_{ij} \tag{5.2}$$

where,

P = $(p_1, p_2, ..., p_m)$,

Q = $(q_1, q_2, ..., q_m)$,

d_{ij} is the ground distance between element i of P and element j of Q and

F = $[f_{ij}]$ is the flow of mass from element i of P to element j of Q that minimizes the overall work

Here an example to find t-closeness [1]. Let Q = {3k, 4k, 5k, 6k, 7k, 8k, 9k, 10k, 11k } and P = {3k, 4k, 5k }. If we have an ordered list $\{v_1, v_2...v_m\}$, then the ground distance between i and j is $\frac{|i-j|}{m-1} = \frac{|i-j|}{8}$. One optimal mass flow that transforms P to Q is to move $\frac{1}{9}$ probability mass across pair. So, D[P,Q] = $(5k \rightarrow 11k)$, $(5k \rightarrow 10k)$, $(5k \rightarrow 9k)$, $(4k \rightarrow 8k)$, $(4k \rightarrow 7k)$, $(4k \rightarrow 6k)$, $(3k \rightarrow 5k)$, $(3k \rightarrow 4k) = \frac{1}{9} \times \frac{(6+5+4+4+3+2+2+1)}{8} =$ 0.375.

An equivalence class of a set,E_1 is said to have *(n,t)*-closeness if there exists a set E_2 of records that is a natural superset of E_1 such that E_2 contains at least n records, and the distance between the two distributions of the sensitive attribute in E_1 and E_2 is no more than a threshold t. A table is said to have *(n,t)*-closeness if all equivalence classes have *(n,t)*-closeness.

5.3 Proposed *t*-closeness

The challenge in privacy preserving data publishing is to have a good tradeoff between privacy and utility. The t-closeness property provides privacy in terms of attribute disclosure [Ninghui *et al.* (2007)]. The distance metric measures the probability distance between P and Q [Ninghui (2007), Ninghui *et al.* (2007)]. In publishing data, P and Q

[1] We consider the example from Reference [Ninghui *et al.* (2007)]

are the two probability distributions where P represents the probability distribution of the an equivalence class of the table and Q represents the probability distribution of the whole table. To satisfy the t-closeness property the distance between P, Q should not exceed t.

Information loss is an unfortunate consequence of anonymization. In order to make the anonymized data as useful as possible, it is required to reduce the information loss as much as possible. Generally to achieve better privacy in t-closeness, reduce the distance between P and Q through anonymization thereby causing the degradation of utility. A concern in data publishing is to provide necessary information through publishing and at the same time minimize the advisories' knowledge.

The goal of this section is to obtain good utility with the same level of privacy as that of (n,t)-closeness. The distance metric deployed in (n,t)-closeness is earth mover distance and does not satisfy the properties of distance metrics like trainagle inequality, probability scaling, etc [Ninghui *et al.* (2007)]. A study on various distance metrics made us aware of the fact that Hellinger distance satisfies all these properties [Derpanis (2008), Comaniciu *et al.* (2003)]. So, we incorporate Hellinger distance metric in the method of anonymizing a table to (n,t)-closeness, and study the performances.

5.3.1 Hellinger Distance Method

Let $P(x)$ and $Q(x)$ represent two multinomial populations, each consisting of N classes with respective probabilities $P(x = 1)$, . . ., $P(x = N)$ and $Q(x = 1)$, . . ., $Q(x = N)$. Since $P(x)$ and $Q(x)$ represent probability distributions, the following equation holds true.

$$\sum_{x \in N} P(x) \quad = \quad \sum_{x \in N} Q(x) = 1 \tag{5.3}$$

The Hellinger distance is given in Equation (5.4).

$$D = \sqrt{(1 - BC)} \tag{5.4}$$

where,

BC is the Bhattacharyya coefficient.

The BC coefficient for the discrete variables can be given by

$$BC(P, Q) = \sum_{x \in X} \sqrt{P(x)Q(x)} \tag{5.5}$$

where,

$P(x)$ is the probability of x in one equivalence class and

$Q(x)$ is the probability of x in whole table.

Let us illustrate with an example the performance of Hellinger distance method and earth mover distance method to achieve a $t-$closeness where t=0.1. For the benchmark Adult dataset, the values attained by using Hellinger distance method and earth mover distance were 0.204 and 0.241 respectively.

Anonymization reacts on the privacy and utility of the dataset. For a dataset if higher level of anonymizations are used the privacy is attained but utility of the data is hampered as higher level of privacy is inversely proportional to information loss. Since P and Q are the probabilities, the distance between two probabilities can be very well measured by using Hellinger method. This method improves the utility of the data, by satisfying t-closeness property.

For the dataset in Table 5.3, the probability of occurring cancer in the dataset $Q(x)$ is 700/3000 i.e. 0.23, while the probability of cancer among individual in the first equivalence class $P(x)$ is 0.5 for $t = 0.1$. So the Hellinger distance is near when compared to the other method. With this comparison, one can conclude that Hellinger method gives better result.

Table 5.3: Original patients table.

#	Zipcode	Age	Disease	Count
1	47673	29	cancer	100
2	47674	21	flu	100
3	47605	25	cancer	200
4	47602	23	flu	200
5	47905	43	cancer	100
6	47904	48	flu	900
7	47906	47	cancer	100
8	47907	41	flue	900
9	47603	34	cancer	100
10	47605	30	flue	100
11	47602	36	cancer	100
12	47607	32	flue	100

The time required to compute distance is really important because the dataset used in the data publishing were having large number of records. Figure 5.3 shows the computing time for the Hellinger metric and earth mover distance metric. As the number of records increases the performance of (n, t) deteriorates than $t-$closeness with Hellinger distance method. This is due to the computation of earth mover distance metric.

5.3.2 Overcoming Identity Disclosure

The $t-$closeness and (n,t)-closeness do not deal with identity disclosure [Ninghui (2007), Ninghui *et al.* (2007)]. A privacy model needs to handle attribute disclosure and identity disclosure efficiently. We agree that by satisfying t-closeness property one can prevent identifying the sensitive information. However, there is a need of a technique that can do better with identity disclosure. Identity disclosure is being a serious concern in privacy. The (n,t)-closeness can be added with k-anonymity to handle identity disclosure, but that would be an overhead. This section proposes an approach to overcome identity disclosure. The proposed approach is data reconstruction method. This method has two steps: randomization with discrete medians and swapping with second order distribution.

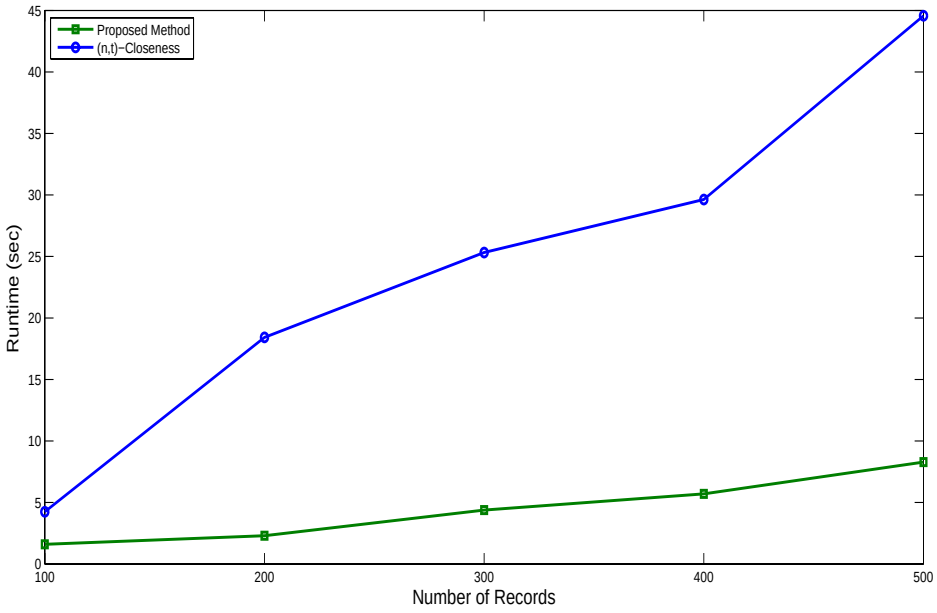

Figure 5.1: Computing time of the two algorithms.

Randomization with discrete median

For each numeric quasi-identifier attribute, a supervised discretization method is used to divide the numeric values into groups [Motwani and Xu (2007)]. The goal of this method is to preserve the relationships between the sensitive attributes and the numeric attributes after discretization. The method recursively splits an attribute to minimize the class entropy and uses a minimum description length criterion to determine the stopping condition. The algorithm evaluates the information gain on each of the potential cut points, and choose the one with the maximum value to split. This process is repeated recursively until a stopping criterion is reached (e.g., the subset contains a single class only). Groups are formed based on the cut points. The value of a numeric attribute is subsequently set for each instance as the median value in corresponding group. The process can be illustrated with an example given in Table 5.4. Let us consider age as a quasi-identifier attribute. Its values can be divided into two groups. Group 1 : age ≤ 34; with median = 28; Group 2 : age > 34; with median = 40; Then the age values are set to 28 for the first six instances, and 40 for the last six instances. Therefore the anonymized

table for Table 5.4 is Table 5.5.

Table 5.4: Example of original table.

ID	Age	Maritial Status	BP	Blood Type	Test Result
1	23	Never married	75/120	O	Negative
2	23	Never married	66/113	O	Positive
3	27	Never married	74/115	A	Negative
4	28	Never married	77/128	AB	Negative
5	32	Married	72/125	B	Negative
6	33	Married	93/147	O	Negative
7	35	Married	75/124	AB	Positive
8	37	Divorced	95/142	O	Negative
9	40	Widow(er)	88/146	A	Positive
10	43	Divorced	110/155	O	Positive
11	45	Married	90/140	O	Positive
12	45	Divorced	104/145	B	Positive

Table 5.5: Anonymized table.

ID	Age	Maritial Status	BP	Blood Type	Test Result
1	28	Never married	75/120	O	Negative
2	28	Never married	66/113	O	Positive
3	28	Never married	74/115	A	Negative
4	28	Never married	77/128	AB	Negative
5	28	Married	72/125	B	Negative
6	28	Married	93/147	O	Negative
7	40	Married	75/124	AB	Positive
8	40	Divorced	95/142	O	Negative
9	40	Widow(er)	88/146	A	Positive
10	40	Divorced	110/155	O	Positive
11	40	Married	90/140	O	Positive
12	40	Divorced	104/145	B	Positive

Swapping with second order distribution

For each nominal quasi-identifier attribute, use a data swapping method given by Reiss to mask the value of the attribute [Reiss (1984)]. With this method, a part of the current values of the quasi-identifier attributes are replaced with new values such that the lower order statistical distributions of the masked data will be close to those of the original data. We apply a second order feedback algorithm, which computes the second order

frequency distributions of the original data and swaps the data such that the new data have approximately the same distributions [Reiss (1984)]. For classification problems, it is important to maintain the second order statistics between the class attribute and each of the quasi-identifier attributes. This is because in many classification techniques, such as decision tree and naive Bayes methods, classification models are built on such second order statistics. The method is implemented using the proposed swapping algorithm (Algorithm 5.1).

Algorithm 5.1 *Swapping*($\mathcal{F}_t$)

1: $\{\mathcal{F}_t$: Set of all Frequency tables, $0 \leq$ t ≤ 2 $\}$
2: D : Database
3: **for** i=1 to N **do**
4: Choose(i, 1, $f_0(1)$)
5: **for** j=1 to V **do**
6: $p2 \leftarrow 0$
7: **for** k = 1 to j-1 **do**
8: $p1 \leftarrow f_1(i, k, j)$
9: $p2 \leftarrow f_2(p1, p2)$
10: **end for**
11: *Choose*(i, j, f_3(p2, j))
12: **end for**
13: **end for**
14: Exit
15: Output D : An N $\times$ V database consistent with F_t

The steps in Algorithm 5.1. states that for each tuple in the table, choose the elements based on probability distribution $f_0(v)$. Various function used in Algorithm 5.1 are $f_0()$, $f_1()$, $f_2()$, $f_3()$ and $Choose()$. The function, $Choose(i, j, p)$ sets $D_{i,j} = 0$ with probability p, or $D_{i,j} = 1$ otherwise. More details about the functions are given in Equations (5.6) to (5.9).

$$f_0(v) = \frac{F_1[v = 0]}{F_0} \tag{5.6}$$

$$f_1(i,j,k) = \begin{cases} \frac{F_2[(v_j=D_{i,j})\times(v_k=0)]}{F_1[(v_j=D_{i,j})]} & \text{if } F_1[(v_j = D_{i,j})] \neq 0 \\ 1/2 & \text{otherwise} \end{cases} \tag{5.7}$$

$$f_2(p1,p2) = p1 + p2 \tag{5.8}$$

$$f_3(p2,n) = \frac{p2}{n-1} \tag{5.9}$$

5.4 Results and Discussion

The main goal of the experiments are to study the effect of the data utility, privacy on real data and investigate the effectiveness of proposed approach with the incorporation of Hellinger method in *t*-closeness privacy model and then applying identity disclosure techniques.

Table 5.6: Adult dataset used in the experiment.

Attribute	**Type**	**Number of Values**
Age	Numeric	72
Work class	Categorical	7
Education	Categorical	16
Marital Status	Categorical	7
Race	Categorical	5
Gender	Categorical	2
Occupation	Sensitive	14

All experiments were conducted using adult dataset of UCI machine learning repository [Asuncion and Newman (2007)]. We used seven attributes of the dataset as shown in Table 5.6. Occupation was considered as the sensitive attribute and the other six fields were treated as quasi-identifiers. After eliminating the missing value records, total valid records taken from the UCI repository for our experiment are 30,717.

The experiment compares the utility of the (n, t)-closeness and proposed approach.

The anonymized data is evaluated with two parameters namely discernibility metric cost and propensity score. The discernibility metric measures the number of tuples that are indistinguishable from each other. It attempts to capture in a straight forward way the desire to maintain discernibility between tuples as much as is allowed for a given value of t. This metric assigns a penalty to each tuple which is the function of indistinguishable tuples in the transformed table. If an unsuppressed tuple falls into an induced equivalence class of size j, then that tuple is assigned a penalty of j. If a tuple is suppressed, then it is assigned a penalty of $|D|$, the size of the input dataset. This penalty reflects the fact that a suppressed tuple cannot be distinguished from any other tuple in the dataset. The metric can be mathematically given as follows:

$$C_{DM}(g,k) = \sum_{\forall |E| \geq k} |E|^2 + \sum_{\forall |E| \geq k} |D||E| \tag{5.10}$$

where,

E is a set refer to the equivalence classes of tuples in D induced by the anonymization g and

C_{DM} is the cost of discernability metric.

The first sum computes penalties for each non-suppressed tuple and the second for suppressed tuples.

The variations in the utility have an vital role in privacy preserving data publishing. As the anonymization hides the information which is to be utilized. The utility, being inverse to anonymization, makes it poor when the anonymization levels are increased. From this it is clear that the anonymization should be less if one needs good utility. However without reducing privacy, utility should be provided. Here privacy is maintained by satisfying t-closeness property. That property will be satisfied when the distance between P, Q is less than 0.1, so the variation with Hellinger is less than (n,t)-closeness and both the metrics will be satisfying by making distance reduced through anonymization of table. The variations are plotted in Figure 5.2.

Figure 5.2 shows the discernibility metric cost of both models. It can be observed from Figure 5.2 that the proposed approach outperforms (n,t)-closeness privacy models.

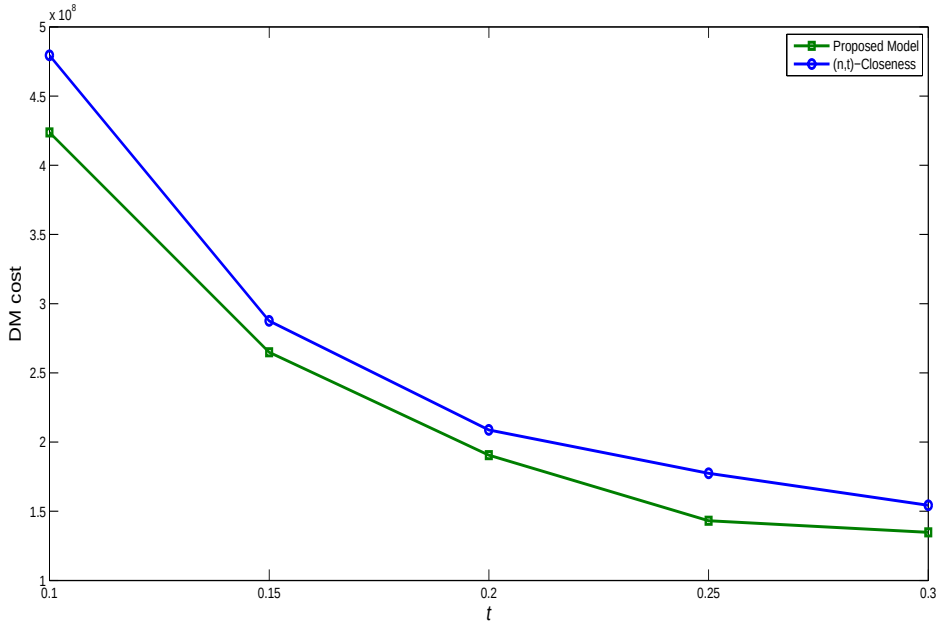

Figure 5.2: Discernibility metric cost.

This ensures that the proposed model has better utility with privacy.

Another metric used for comparison is propensity score [Reiss (1984), Woo *et al.* (2009), Rosenbaum and Rubin (1983)] for both anonymized data of proposed model and $(n,\ t)$-closeness model. The similarity of the propensity scores for the masked and original observations can be assessed in numerous ways, for example, by comparisons of their percentiles in each group as in Equation (5.11).

$$U_P = \frac{1}{N}\sum_{i=1}^{N}[p_i - c]^2 \tag{5.11}$$

where,

N is the total number of records in the merged dataset,

p_i is the estimated propensity score for unit i,

c equals the proportion of units with masked data in the merged dataset and

U_P is the utility of the propensity score.

The comparative results are shown in Table 5.7. Propensity score metric suggests an approach for measuring data utility. When two distributions are similar, the distributions

of the original and anonymized data are similar, so data utility is relatively high. From Table 5.7, one can draw the conclusion that the utility of proposed model is efficient as the score is low.

Table 5.7: Comparison of propensity scores.

Utility Measure	**(n,t)closeness**	**Proposed Model**
Propensity Score	17.947	0.25

5.5 Conclusion

Trading between privacy and utility should be good. Existing privacy models have some drawbacks and are unable to balance privacy and utility. This chapter proposes an approach for good trading between privacy and utility without degrading the privacy. All the experiments that have been conducted consider only nominal attributes and no categorical attributes were considered. The model satisfies t-closeness property using Hellinger distance method; therefore attribute disclosure is well handled. However to address identity disclosure, an algorithm is proposed using data reconstruction technique. Various experiments were conducted showing the proposed models outperform the existing models for balancing utility and privacy.

Chapter 6

Prominence-Based Anonymization of Social Networks

6.1 Introduction

Social networks allow millions of users to create profiles and share personal information. They resemble a small world simulating many real-world situations [Travers and Milgram (1969)]. Social network analysis has a wide variety of applications, such as in business, marketing, entertainment, etc. The proliferation of social network sites resulted in numerous problems to the world. Advances in the social network analysis of published data poses problems related to privacy and have received considerable attention as of late. Privacy is person-specific and has various forms like relational, influential and emotional. Efficient methods exist to analyze the great treasure of information in social networks [Gautam *et al.* (2008), Ralph and Alessandro (2005), George *et al.* (1993)]. Paradigms such as social network federation [Chao *et al.* (2012)] add more fuel to the burning problem of privacy issues in social networks.

Data must be published to share useful information. Privacy Preserving Data Publishing (PPDP) is an approach to publishing data in a hostile environment, while preserving the privacy of individuals involved in a network [Benjamin *et al.* (2011), Duan *et al.* (2005)]. PPDP techniques include generalization with suppression, anonymization

with permutation and perturbation [Benjamin *et al.* (2011)]. These techniques use various mathematical and statistical properties to mask actual data. Performing mathematical operations on actual data may transform actual values into some other values (anonymization), group similar items and combine them into one category (generalization) or generate synthetic data. Any PPDP technique should take utility into consideration. George *et al.* discuss various issues regarding utility and risks [George *et al.* (2011)].

Social network data poses a different and unique challenge in modeling privacy due to highly interrelated nodes. These challenges include modeling adversary background knowledge, calculating information loss and determining the suitability of anonymization techniques [Aggarwal and Wang (2010)]. The utility of published data in social networks is affected by degree, path length, transitivity, network reliance and infectiousness [Hay *et al.* (2008)].

In this chapter, we propose a method for anonymizing users in a social network. The anonymization of social network nodes was based on domain generalization and sensitivity of the nodes. The rest of the chapter is organized as follows. The following section provides a background and a review of existing methods used to preserve privacy. The proposed method for anonymization is described in Section 6.3. Section 6.4 presents the simulation results, and Section 6.5 concludes the chapter.

6.2 Background and Related Work

This section describes various attacks on social networks, centrality measures that determine the importance of nodes and related work.

6.2.1 Attacks on Social Networks

Three types of attacks can be launched in an anonymized social network namely, identity disclosure attack, link disclosure attack and content disclosure attack [Liu *et al.* (2008)]. These attacks can be either active or passive. In active attack, an attacker creates an

account and targets the nodes and their relationships. In passive attack, the uniqueness of small random sub-graphs is analyzed and the target nodes are re-identified [Backstrom *et al.* (2007)].

Identity disclosure attack aims at identifying a node in an anonymized network. Solutions proposed for this are k-candidate anonymity and graph randomization [Liu *et al.* (2008), Hay *et al.* (2007), Samarati and Sweeney (1998b)], k-degree anonymity and minimal edge modification [Liu *et al.* (2008)] and k-neighborhood anonymity and graph isomorphism [Zhou and Pei (2008)]. Removing labels of the network will not prevent identity disclosure [Liu *et al.* (2008)]. Another method of identity disclosure is through face recognition [Gross (2005), Liu and Maes (2005)]. Image anonymization may also be possible with blurring, but one may post any image in a network and later have a repudiation problem. To overcome such problems, the methods proposed by Preda and Vizireanu can be deployed [Preda and Vizireanu (2010, 2011a,b)].

Link or sensitive edge disclosure attack aims at revealing an important relationship between a pair of nodes. Zheleave and Getoor proposed the concept of sensitive edge and observed edges to attain a balance between utility and privacy through a series of algorithms called intact edges and partial edge removal [Zheleva and Getoor (2007)]. Other techniques include privacy preserving link analysis, random perturbation for privacy relationship protection [Ying and Wu (2008)], cryptographic protocol for private relationships protection [Curminati *et al.* (2007)] and synthetic graph generation [Leskovec and Faloutsos (2007)].

Content disclosure attack aims at revealing the private information associated with a network, such as e-mails, blogs, etc. These types of attacks are generally launched by advertising organizations on online social networking sites; one recent example is Facebook's Beacon service [Liu *et al.* (2008)].

6.2.2 Centrality Measures

A common method used to understand networks and their nodes in a static design is to evaluate the location of nodes in the network in terms of strategic position. Node

centrality measures help to determine the importance of a node in a network. Bavelas initially investigated the formal properties of centrality as the relationship between structural centrality and influence on group processes [Bavelas (1948)]. He proposed several centrality concepts. Later, Freeman argued that centrality is an important structural factor affecting leadership, satisfaction and efficiency [Freeman (1979)]. To quantify the importance of an user (node) in a social network, various centrality measures have been proposed over the years [Bavelas (1950), Freeman (1979), Scott (1991), Alireza *et al*. (2011), Krackhardt (2010), Sabidussi (1966)].

The simplest and easiest way of measuring a node's centrality is by counting the number of other nodes connected directly to the node. This *degree* of a node can be regarded as a measure of local centrality [Scott (1991)]. Nodes with a high degree centrality imply popularity and leadership [Bavelas (1950), Freeman (1979), Krackhardt (2010)].

The degree centrality formula is given as

$$C_D(i) = \frac{d(i)}{n-1} \tag{6.1}$$

where,

$C_D(i)$ is degree centrality of node i,

$d(i)$ is the number of edges connected to the node and

n is the total number of nodes in the network.

The closeness centrality [Sabidussi (1966)] is a measure of global centrality in terms of distance among various nodes. The sum of the distances between a node and other nodes is $farness$. Its inverse is closeness. Closeness centrality of node i can be defined as follows:

$$C_C(i) = \frac{n-1}{\Sigma_{j=1}^{n} d(i,j)} \tag{6.2}$$

where,

$C_C(i)$ is closeness centrality of node i,

$d(i, j)$ is the shortest distance from node i to node j and

n is the total number of nodes in the network.

Another global measure of centrality is betweenness [Freeman (1979)]. This measure describes the number of times a particular node lies *between* the various other nodes in a network. Nodes with high betweenness centrality play the role of a broker or gatekeeper in connecting nodes and subgroups. So, they can most frequently control information flow in a network. The betweenness of node i is given as follows:

$$C_B(i) = \sum_{j<k} \frac{P_{jk}(i)}{P_{jk}} \tag{6.3}$$

where,

$C_B(i)$ is betweenness centrality of node i and

P_{jk} is the number of shortest paths between nodes j and k.

Page rank is a more refined measure of the prominence of a node in a directed network. Each node in a social network is treated as a page. The importance of a page i is determined by how much prominent a page j that has an outlink to i. The formula is given as follows:

$$P(i) = \sum_{j,i \in E} \frac{P(j)}{O_j} \tag{6.4}$$

where,

$P(i)$ is the page rank of node i,

O_j is the number of outlinks of page j and

E is the set of edges in the network.

6.2.3 Related Work

Each user of social networks expects different levels of privacy. The anonymization of a given node depends on the importance of the node. Xiao and Tao provide the option of allowing a node to determine its own privacy level [Xiao and Tao (2006)]. However,

this approach has drawbacks, such as a reduction in the utility of published data because most nodes set their privacy level to the maximum. This motivates us to anonymize the privacy level of nodes based on social reputation. The privacy level is denoted by the sensitivity index, which is calculated according to the centrality or prestige, depending on the nature of the network.

Zhou and Pei proposed an anonymization scheme, which deploys sub-graph matching using a minimum depth-first search tree [Zhou and Pei (2008)]. A sub-graph matching scheme is used to find the related sub-graphs in a graph and anonymize the labels of these sub-graphs. The drawback of this scheme is that it cannot be practically implemented in actual social networks with thousands of edges and of nodes. Furthermore, it considers only one neighbor. Upon increasing the number of nodes to more than one, the number of neighbors grows exponentially. In a recent study by Tripathy and Panda, the neighborhood size was increased to three [Tripathy and Panda (2010)]. Still, these approaches [Zhou and Pei (2008), Tripathy and Panda (2010)] cannot be applied due to very large size of the social networks, making them practically impossible. The anonymization approach reported in [Zhou and Pei (2008), Tripathy and Panda (2010)] statically anonymizes nodes. This reduces the utility of published data.

Liu *et al*. proposed two methodologies to solve the problem of sensitive edge disclosure attack [Liu *et al*. (2008b)]. They used the Gaussian randomization and greedy perturbation schemes. These two methods are computationally intensive. For a practical network, the complexity is very high. These two methods do not consider the dynamic nature of the weight associated with the edges.

In the next section, we discuss the proposed scheme for anonymizing social networks.

6.3 Proposed Scheme for Anonymizing Social Networks

Social network data are of two types: structural data and composite data. Structural variables hold the relationships between nodes, and composite variables hold the attributes

of the node. The domain in each of these variables has some generalization hierarchy. A taxonomy tree can be constructed with the domain generalization hierarchy. For the conventional problems of data anonymization, Sweeney proposed a method using domain generalization hierarchy [Sweeney (2002b)].

6.3.1 Problem Formulation

Given a finite set of entities (S) = $\{S_1, S_2, \ldots, S_n\}$, its edges (T) = $\{T_1, T_2, \ldots, T_k\}$, an entity domain generalization hierarchy (DGH_E) = $\{E_1, E_2, \ldots, E_l\}$ and edges domain generalization hierarchy (DGH_T) = $\{R_1, R_2, \ldots, R_m\}$, an anonymized set of entities (S^1)= $\{S_1^1, S_2^1, \ldots, S_p^1\}$, where $p < n$, with a generalization for DGH_E or DGH_T as a function on A. That is for each f : A $\rightarrow$ B, is a generalization where B $\in \{A_1, A_2, \ldots, \mathrm{A}_x\}$ in a generalization hierarchy of $A_0 \rightarrow A_1 \rightarrow .. \rightarrow \mathrm{A}_x$. The function imposes a linear ordering from A to B. A_0 is the base domain and A_x is the highest domain level. The goal is to generalize A to B depending on the sensitivity index of each entity S_i. Generalization of DGH_E implies generalization of DGH_T and vice versa.

The above paragraph states, given a set of entities (nodes) and edges with their generalization hierarchy, the network is anonymized by promoting the nodes or edges to a higher level of generalization according to the taxonomy tree of the domain generalization hierarchy. The generalization must be made based on the sensitivity index of each node.

6.3.2 Process for Anonymization

The step-by-step process for anonymization of the nodes is social network is given below.

Step 1. Generate a taxonomy tree for the quasi-identifiers.

Step 2. Compute the sensitivity index of each node.

Step 3. Generalize each node according to the sensitivity index.

Step 4. Form super nodes from the generalized vertex.

Details of the above steps are provided in the subsequent paragraphs.

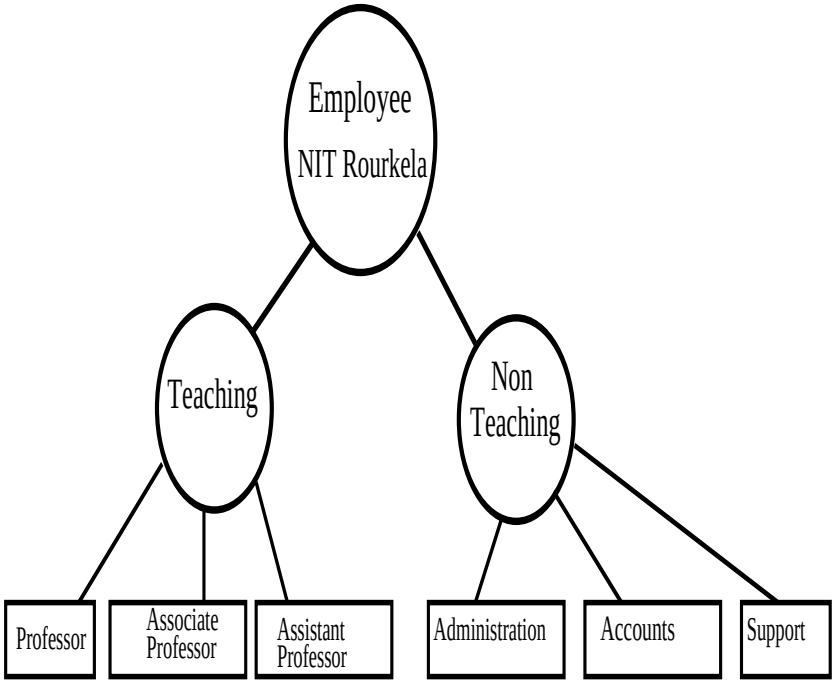

Figure 6.1: Sample tree.

Generating a Taxonomy Tree

A taxonomy tree is a hierarchical representation of domain values. A sample taxonomy tree regarding job profiles at NIT Rourkela is shown in Figure 6.1. To create such a tree, both structural and composite data are required. A composite variable of a node has many attributes. All of the domain generalizations of quasi-identifiers should be considered while building the taxonomy tree. Quasi-identifiers are the set of attributes that determine the sensitive data.

Computing Sensitivity Index

The sensitivity index indicates the importance of a node in a network. To evaluate the sensitivity index, one needs to determine the importance of a node. This idea is borrowed from the social phenomenon that, if a person is famous or important, the number of people linked to him or her is high. It is also evident that in social networking sites such as Twitter, popular people are followed by many people. Centrality measures are used to determine the sensitivity index of nodes. The details of various centrality measures were

discussed in Section 6.2.2.

Algorithm 6.1 is used to determine the sensitivity index of all the nodes. The inputs to this algorithm are a dataset ($\mathcal{D}$) and level of generalization ($\mathcal{L}$). This algorithm computes the masked centrality ($\mathcal{MC}$) of each node, creates an anonymizing matrix ($\mathcal{AM}[][]$) using Algorithm 6.2 and determines the anonymization level of each node in $\mathcal{AM}$ using Algorithm 6.3.

The masked centrality of node i, represented by $\mathcal{MC}[i]$, is computed and depends on the category to which a given network belongs. Social networks can be classified into four categories depending on their directional and weight properties: (i) Unweighted Undirected Network, (ii) Unweighted Directed Network, (iii) Weighted Undirected Network and (iv) Weighted Directed Network. More details about the centrality measure deployed to compute $\mathcal{MC}$ in the proposed scheme are provided in Section 6.4. From a global prospective, the prominence of a node depends on the direction of its edges. Algorithm 6.4 is proposed to fulfill this criterion.

Generalizing Nodes

The set of values of a particular attribute can be addressed as a domain. Given two domains S0 and S1, S0 $\leq$ S1 describes the fact that the values of the attributes in S1 are more generalized. There are several schools of thought regarding generalization ontologies [Samarati and Sweeney (1998b)]. Figures 2.2, 2.3 and 2.4 of Chapter 2 shows sample value generalization hierarchies.

Once the sensitivity index is computed, a node is generalized according to the index. The generalizations of these nodes must follow the domain generalization hierarchy.

Formation of super nodes

After the generalization step described above, there will be many nodes with the same label. One can merge these nodes to remove redundancy and maintain the structure of the network. The other reason for merging nodes is to eliminate attacks such as neighborhood and sensitive edge disclosure.

Algorithm 6.1 *Compute_Sensitivity_Index*($\mathcal{N}$,$\mathcal{D}$,$\mathcal{L}$)

{$\mathcal{N}$: Number of nodes, $\mathcal{D}$: Dataset containing source and destination nodes, $\mathcal{L}$: Taxonomy level of the dataset }
$\mathcal{SI}$[]: Sensitivity Index
$\mathcal{AM}$[][]: Anonymization Matrix
$\mathcal{MC}$[]: Masked Centrality
|.|: Cardinality
$Temp_1[], Temp_2[]$: Temporary Variables /* Both are single dimension arrays */
for $i = 1$ to $\mathcal{N}$ **do**
    $Temp_1[i] \leftarrow Centrality(i)$ /* Centrality(i) is a function for computing the centrality of a node, i */
    $Temp_1[i] \leftarrow Temp_1[i]$ * order of $\mathcal{D}$
    $Temp_2[i] \leftarrow Temp_1[i]$ + Mean of the centrality
    $\mathcal{MC}$[i] $\leftarrow round(Temp_2[i])$
    /* round(i) is a function for computing the upperbound of the value i */
end for
$\mathcal{AM}[][] \leftarrow Create\_Anonymization\_Matrix(\mathcal{MC}[], \mathcal{L})$
for $i = 1$ to $\mathcal{N}$ **do**
    $\mathcal{SI}$[i] $\leftarrow Find\_Anonymization\_Level(\mathcal{MC}$[i],$\mathcal{AM}[][])$
    return $\mathcal{S}$[]
end for
Output $\mathcal{S}$[]: Array consisting of sensitivity index of each node

Algorithm 6.2 *Create_Anonymization_Matrix*($\mathcal{MC}$[],$\mathcal{L}$)

{$\mathcal{MC}$[]: Masked Centrality, $\mathcal{L}$: Taxonomy level of the dataset }
DC[]: Array to store distinct centrality values
A, B, C: To store intermediate values
$DC[] \leftarrow distinct\_centrality(\mathcal{MC}[])$
$A \leftarrow |DC[]|$
$B \leftarrow ceil(A/\mathcal{L})$
$DC[] \leftarrow sort(|DC[]|)$
for $j = 1$ to $\mathcal{L}$ **do**
    for $k = 1$ to B **do**
        AM[j][k] $\leftarrow DC[i]$ /* AM[] is the anonymization matrix of dimension L x B */
        $i = i + 1$
    end for
end for
return $AM[][]$
Exit
Output $AM[][]$: Anonymization matrix

Algorithm 6.3 *Find_Anonymization_Level*($\mathcal{MC}[node]$,$\mathcal{AM}[][]$)

1: {$\mathcal{MC}[node]$: Masked Centrality of *node*, $\mathcal{AM}[][]$: Anonymization Matrix }
2: *row* : Number of rows in $\mathcal{AM}[][]$
3: *col* : Number of columns in $\mathcal{AM}[][]$
4: **for** i=1 to *row* **do**
5: **for** j=1 to *col* **do**
6: **if** $\mathcal{AM}[i][j] = \mathcal{MC}[node]$ **then**
7: return i
8: **end if**
9: **end for**
10: **end for**
11: Exit

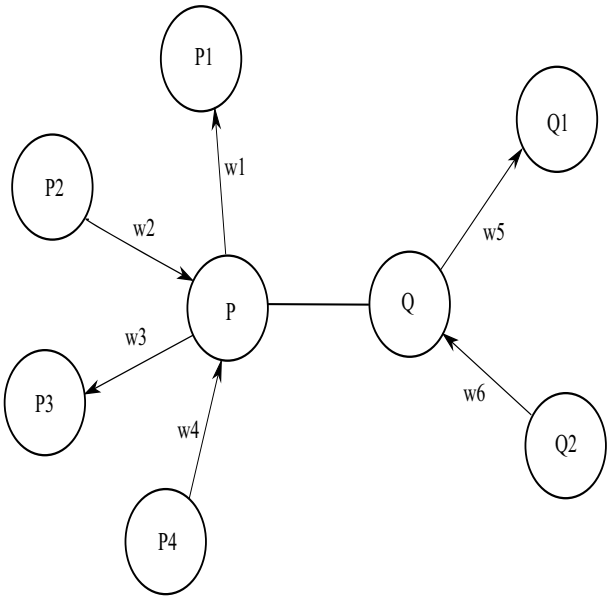

Figure 6.2: Before merging.

Algorithm 6.4 *Node_Rank*($\mathcal{D}$, *itr*)

1: { $\mathcal{D}$: Dataset containing source and destination nodes, *itr* : Number of iterations chosen heuristically }
2: e_i: Entropy
3: $\mathcal{N}$: Number of nodes
4: $Temp_1[], Temp_2[], Temp_3[]$: Temporary variables
5: *entropy*(): Function to calculate entropy of a distribution
6: **for** $i = 1$ to $\mathcal{N}$ **do**
7: $Temp_1[i] \leftarrow Centrality(i)$ /* Centrality(i) is a function for computing the centrality of a node, i */
8: $Temp_2[i] \leftarrow$ Indegree(i)
9: $Temp_3[i] \leftarrow$ Outdegree(i)
10: **end for**
11: $e_1 \leftarrow 0$
12: $e_2 \leftarrow 1$
13: **for** $l = 1$ to itr **do**
14: **if** $e_1 \neq e_2$ **then**
15: **for** $i = 1$ to $\mathcal{N}$ **do**
16: **if** $Temp_1[i] \neq 0$ **then**
17: $count \leftarrow Temp_2[i]$
18: **for** $j = 1$ to count **do**
19: $Temp_1[i] \leftarrow Temp_1[i] + \frac{Temp_2[j]}{Temp_3[j]}$
20: **end for**
21: **end if**
22: **end for**
23: $e1 \leftarrow e2$
24: $e2 \leftarrow entropy(Temp_1[])$
25: **end if**
26: **end for**
27: Exit
28: Output $Temp_1[]$

Figures 6.2 and 6.3 illustrates merging of nodes that are neighbors which have the same node label. When nodes are merged, their degrees and weights are also merged.

Assuming P and Q are the nodes to be merged, the weight of each edge in this new network is calculated using Equation (6.5) and all of the edges are rewritten.

$$Weight^{'}(i,j) = \frac{Weight_{i,j}}{\sum Weights} \tag{6.5}$$

If weights and directions associated with each edge, then the following formula can be used to compute the new weights.

$$W_{in} = W_2 + W_4 + W_6 \tag{6.6}$$

$$W_{out} = W_1 + W_3 + W_5 \tag{6.7}$$

Through this operation, actual information is very effectively protected without destroying much utility. The new values provide information about the fraction of the total weight directed in or out of the network. Thus, a viewer can get an idea about the percentage of a transaction or material flow but never be able to know what the exact amount is, which is a very effective way to protect against sensitive edge disclosure attack.

Algorithm 6.5 ($Merge\_Nodes$) is invoked when two neighborhood nodes have the same label.

6.4 Results and Discussion

The dataset used for simulation is the Wiki-votes dataset available in the Stanford Large Network Dataset Collection [Stanford University (2008)].

6.4.1 Taxonomy Tree

Given a dataset, there are no computational methods to generate a taxonomy tree. Based on real-life scenarios, one can assume what the taxonomy tree should be using simple common sense. Table 6.1 shows the statistics of the Wiki-votes dataset. As no

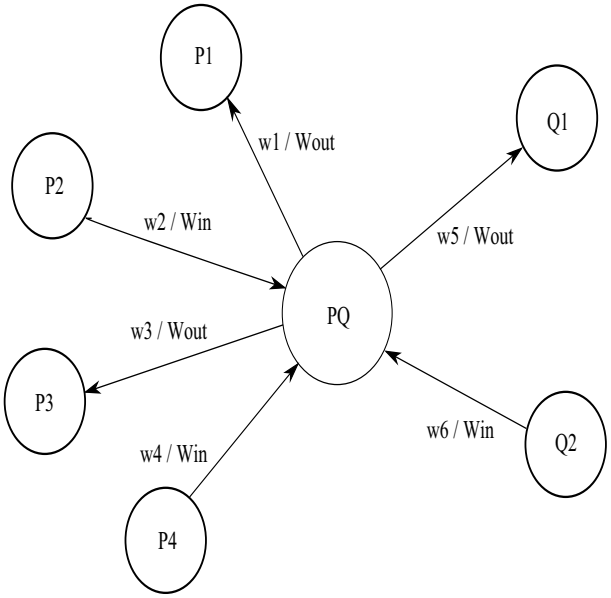

Figure 6.3: After merging.

Algorithm 6.5 $Merge\_Nodes(\mathcal{N}_1, \mathcal{N}_2, AdjMat)$

{ $\mathcal{N}_1, \mathcal{N}_2$: Node to be merged, $AdjMat$: Adjacency matrix of the network }
k: Temporary variable
W: Weights
Find the Neighbor of $\mathcal{N}_1 and \mathcal{N}_2$
Copy Neighbors of $\mathcal{N}_1 and \mathcal{N}_2 into \mathcal{N}_1\mathcal{N}_2$
Merge $\mathcal{N}_1(W) \cup \mathcal{N}_2(W) - \mathcal{N}_1(W) \cap \mathcal{N}_2(W)$
Remove Redundancy from neighbors of $\mathcal{N}_1\mathcal{N}_2$
Remove $\mathcal{N}_1$ and $\mathcal{N}_2$ from the AdjMat
Exit
Output $\mathcal{N}_1\mathcal{N}_2$: New node formed by merging $\mathcal{N}_1$ and $\mathcal{N}_2$

structural information is available about the dataset, the node labels are used to create the taxonomy tree. The hierarchy level is created by dividing the node value by 10. A sample generalization of a few nodes is depicted in Figure 6.4, with a hierarchy of level five. Level one is the root node that is fully generalized.

6.4.2 Computing Sensitivity Index

The sensitivity index of each node was computed using Algorithm 6.1. As mentioned in the earlier sections, the sensitivity of a node depends on its prominence. Different centrality measures are used to determine the prominence based on the type of network. The centrality measures deployed and the results obtained are shown in subsequent paragraphs.

Unweighted Undirected Network

This is the simplest type of network. Hence, the degree centrality measure could be the best to anonymize the network.

The degree distribution of the original network and the distribution of the network with masked centrality is shown in Figure 6.5 and Figure 6.6.

We have obtained the following results by applying degree centrality to Algorithm 6.1. maximum degree = 1167, maximum centrality = 0.14067, maximum masked centrality = 141, Number of distinct masked centrality = 67.

For this type of network, one can even deploy other centrality measures. For example, we consider two more measures, namely betweenness centrality and a hybrid centrality of degree and betweenness.

Betweenness is an index of a position's potential for gaining control over the communication in a network. The leader identification is predicted by the control-based measure of centralization. A person in such a position with high betweenness can influence others by withholding or distorting information in transmission. The formula for betweenness is given in Equation (6.3). Figure 6.7 shows the distribution of the nodes with betweenness centrality.

The results attained are: maximum betweenness centrality = 0.0646, minimum

Table 6.1: Description of Wiki-votes dataset.

Details	**Statistics**
Nodes	8297
Edges	103689
Nodes in Largest WCC	7066 (0.993)
Edges in Largest WCC	103663 (1.000)
Nodes in Largest SCC	1300 (0.183)
Edges in Largest SCC	39456 (0.381)
Average Clustering Coefficient	0.2089
Number of Triangles	608389
Fraction of Closed Triangles	0.1255
Diameter (Longest shortest path)	7
90-percentile effective Diameter	3.8

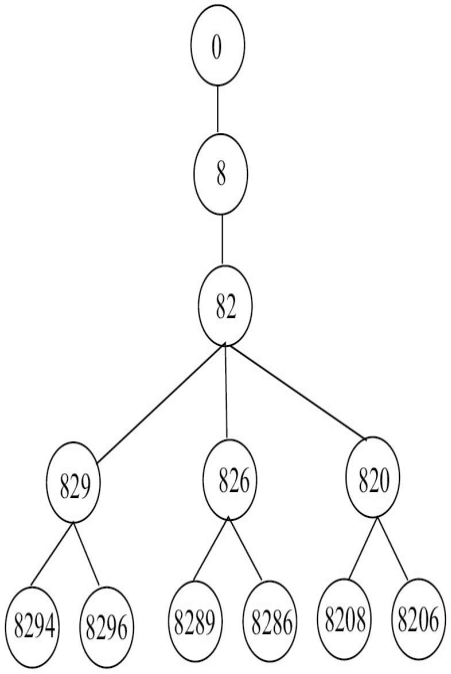

Figure 6.4: Taxonomy tree.

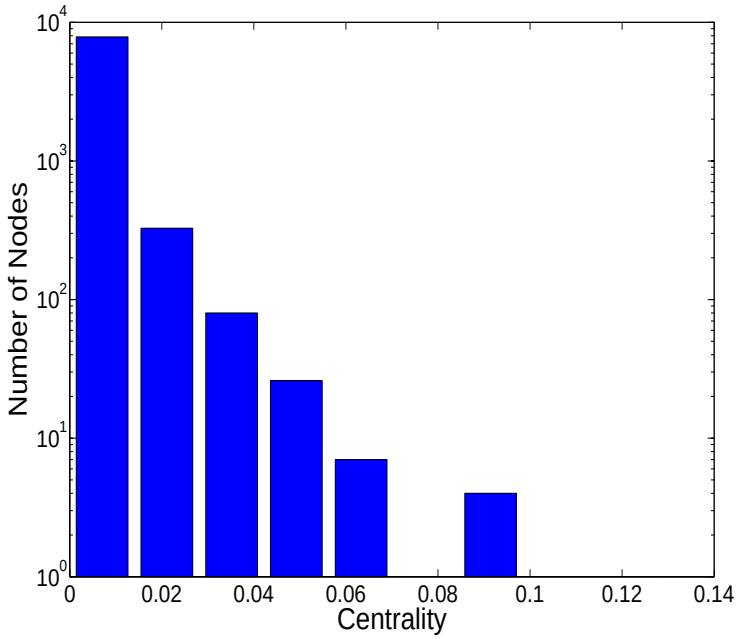

Figure 6.5: Original network.

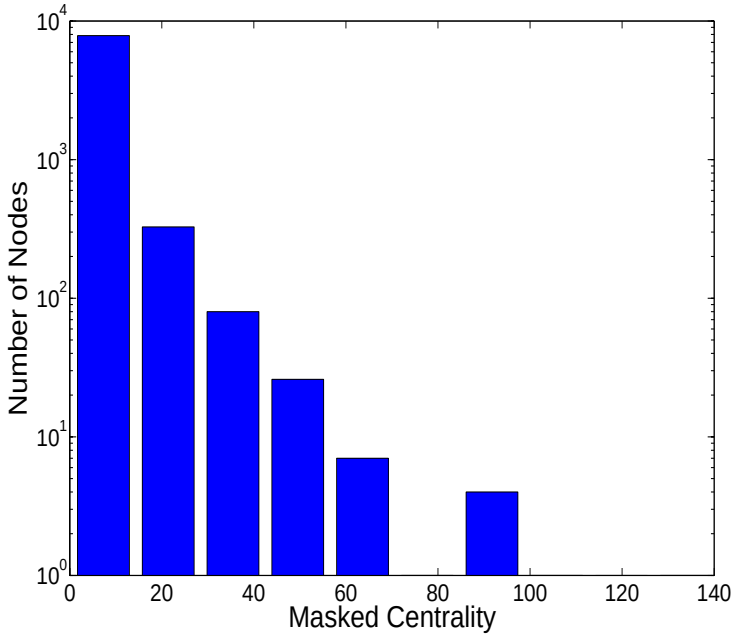

Figure 6.6: Masked network.

Betweenness centrality = 0, max normalized centrality = 1000, Number of distinct normalized centrality = 127.

The degree centrality is easy to compute but local in nature. The betweenness centrality is global in nature but does not consider direct influence. One needs to consider both local and global importance. We combine both centralities to create a hybrid centrality. The following formula is used for hybrid centrality.

$$Comb(i) = a \times D(i) + b \times B(i) \tag{6.8}$$

where,

i is a node,

$Comb(i)$ is the combined centrality of degree and betweenness,

$D(i)$ is the degree centrality of i,

$B(i)$ is the betweenness centrality of i and

a, b are tuning parameters.

The values of degree centrality and betweenness centrality are normalized such that they vary between 0 to 1. The values of a and b are determined by the data publisher. It depends on the network type. We considered $a + b$=1.

The horizontal axis of Figure 6.8 is the combined centrality, and the vertical axis is the number of nodes. A total of nine experiments were performed using different values of the tuning parameters varying from 0.1 to 0.9. The combined centrality was calculated using the Equation (6.8).

Unweighted Directed Network

This type of network shows relationships that are not symmetric, such as Employer-Employee and Producer-Consumer relationships. To rank the nodes, one needs to consider both the indegree and outdegree. The page rank algorithm [Page *et al*. (1999)] operates using a reference model that has been specially tailored and used in Algorithm 6.4 ($Node\_Rank$). By using page rank, we ensure that a link from an important node

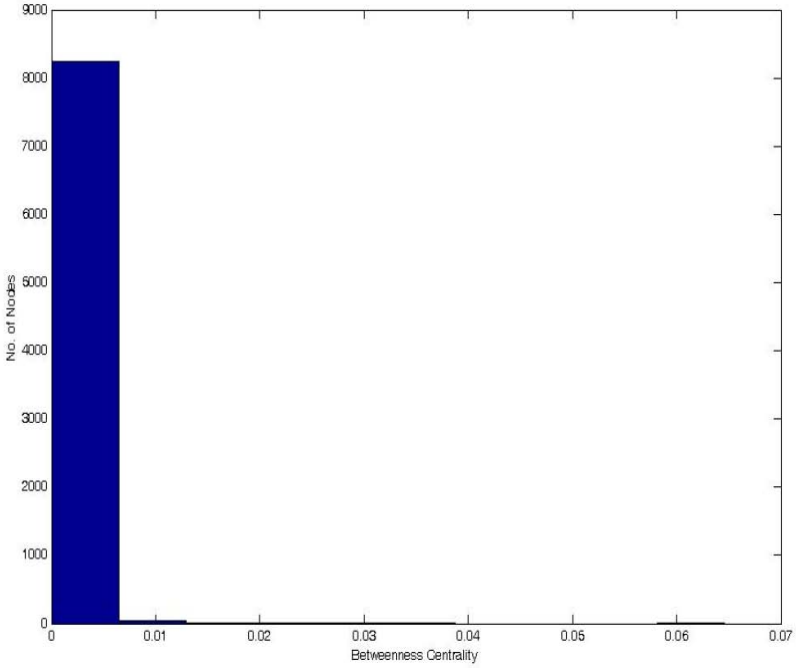

Figure 6.7: Betweenness centrality distribution.

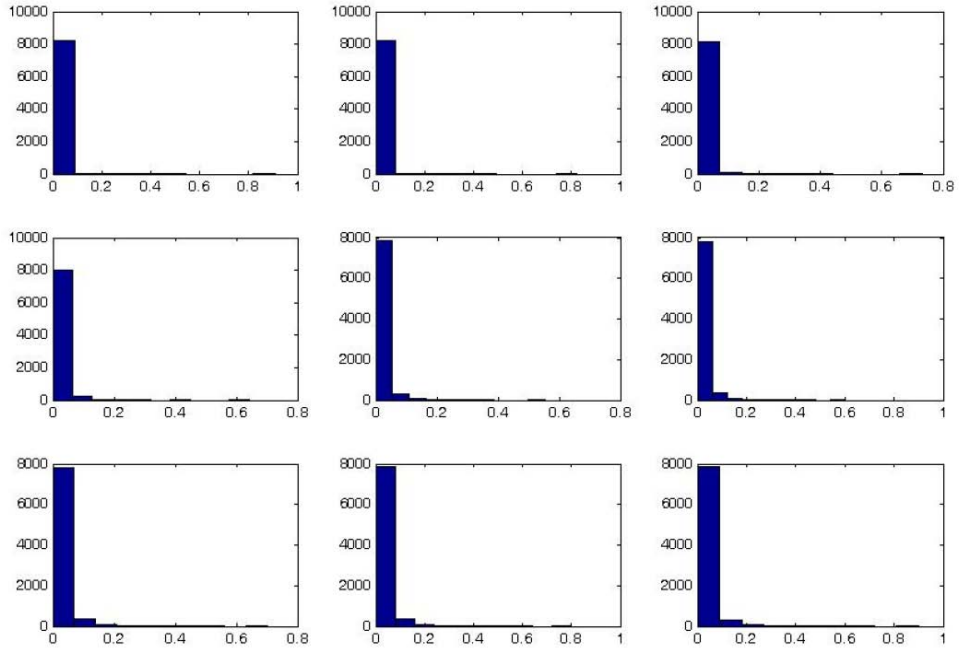

Figure 6.8: Combined centrality distribution.

fetches a higher rank than a link from a less important node. This simulates real-life experience.

The modification made in the page rank algorithm is due to the nature of the dataset. The first modification made is in assigning the initial rank. The initial rank assigned in Algorithm 6.4 is the degree centrality instead of the probability of choosing a link from all available links. The second modification is in choosing the converging criteria for this iteration. Page rank use eigen values. Algorithm 6.4 uses the concept of entropy due to fast convergence and less expensive computation. Once the ranks are calculated, the nodes can be assigned by their anonymization level using Algorithm 6.2 and Algorithm 6.3.

Figure 6.9 shows the variation in entropy with respect to the number of iterations. Using Algorithm 6.4, the Wiki-votes dataset was used and it gets converged after 14 iterations. Figure 6.10 shows the power law distribution of the node rank, through which the utility of the dataset is preserved.

Results attained are: maximum rank = 10826, minimum rank = 0, Number of distinct ranks = 2465.

Weighted Undirected Networks

Weighted networks simulate quantified relations. These weights can describe the level of interaction between two nodes or the amount of money or material flow between nodes. The weight indicates the level of closeness or mutual importance that nodes share. Weight Centrality (WC) is calculated using the following equation:

$$WC = Centrality \times Weight \tag{6.9}$$

For this type of network, the weight centrality is used in Algorithm 6.1 to compute the centrality. Figure 6.11 shows the rank, which satisfies the power law distribution.

Results attained are: maximum rank = 111, minimum rank = 0, Number of distinct rank = 55.

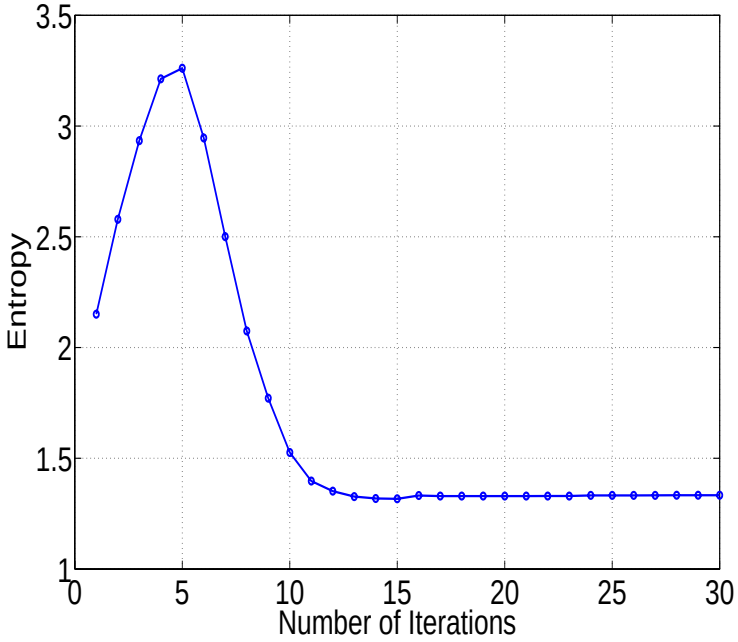

Figure 6.9: Convergence of rank for unweighted directed network.

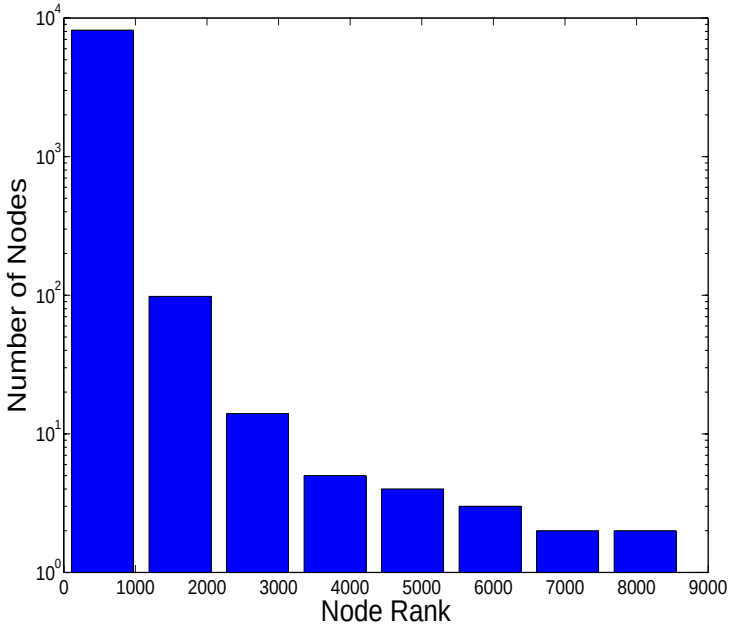

Figure 6.10: Node rank showing power law distribution.

Weighted Directed Network

Weighted and directed networks represent quantified and asymmetrical relationships. This type of network simulates traffic information, telecommunication networks, etc. The measure of node prominence is weight centrality. The weight centrality is calculated as given in Equation (6.10).

$$WC = DC \times WR \tag{6.10}$$

$$WR(forRN) = \frac{InWeight}{OutWeight} \tag{6.11}$$

$$WR(forPN) = \frac{OutWeight}{InWeight} \tag{6.12}$$

where,

WC = Weight Centrality, DC = Degree Centrality, WR = Weight Ration,

RN = Reference Networks and PN = Production Networks

The calculation of the Weight Ratio depends on the nature of the network. Equation (6.11) is used to determine importance, where the weight coming into the network is considered to be important. Equation (6.12) is used for networks in which the outward weight from the network is considered important.

To rank nodes in this type of network, we consider both the weight and directional features. Thus, we modify Algorithm 6.4 by providing the initial node ranks with WC. The node rank is computed by using Equation (6.13). Apart from this, the algorithm remains the same as Algorithm 6.4.

$$Noderank(P) = \sum_{Q} \frac{Rank(Q) \times Weight_{Q \to P}}{\sum Weight_{Q \to AllNeighbors}} \tag{6.13}$$

where

Q represents all of the nodes that have an outlink to node P.

Figure 6.12 shows the variation in entropy with the number of iterations. The Wiki-votes dataset requires 30 iterations to converge. Figure 6.13 shows a histogram

of weighted node rank, which satisfies a power law distribution.

Results attained are: maximum rank = 3.1932×10^{52}, minimum rank = 0, no. of distinct rank = 1529.

6.4.3 Generalizing the Nodes

After determining the sensitivity index, we generalize the nodes. The labels of the nodes are modified based on the index. In this study, while generalizing the nodes, the domain generalization hierarchy shown in Figure 6.4 was followed.

The generalization of nodes or edges resulted in a loss of information. The metric chosen to calculate information loss is *iloss* is given in Section 3.3 of Chapter 3.

Degree Centrality

For simplification, we keep w_i=1 in the formula of *iloss*. This approach is used for all four types of networks, and *iloss* by using degree centrality is calculated using Equation (6.1).

Table 6.2 shows the information loss calculated for different types of networks.

Betweenness Centrality

The weight (w_i) considered here also is 1. Table 6.3 shows the values of *iloss* calculated through customized privacy, where the sensitivity index of the nodes is determined using betweenness centrality.

Combined Centrality

Here, we keep w_i=1. Table 6.4 shows the *iloss* due to customized generalization using combined centrality for different values of the tuning parameters to determine the sensitivity index of nodes for an unweighted and undirected network and a weighted and undirected network.

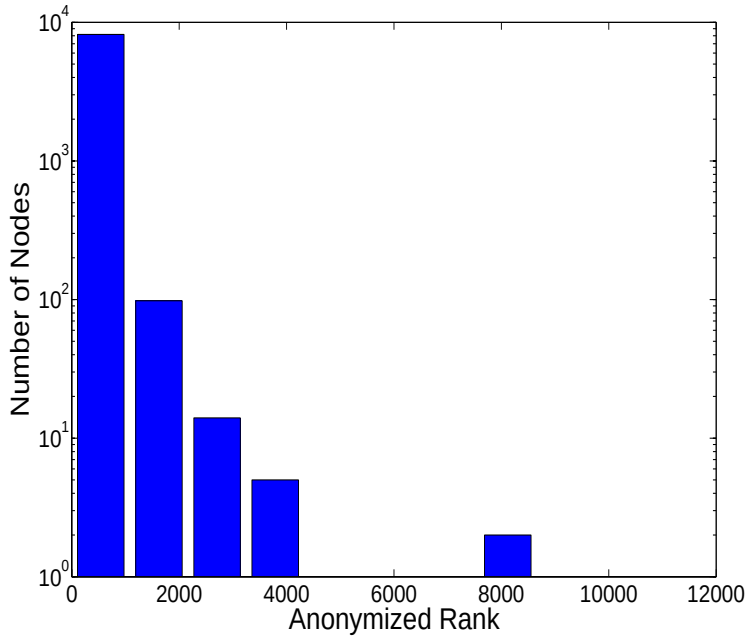

Figure 6.11: Weighted masked weighted centrality.

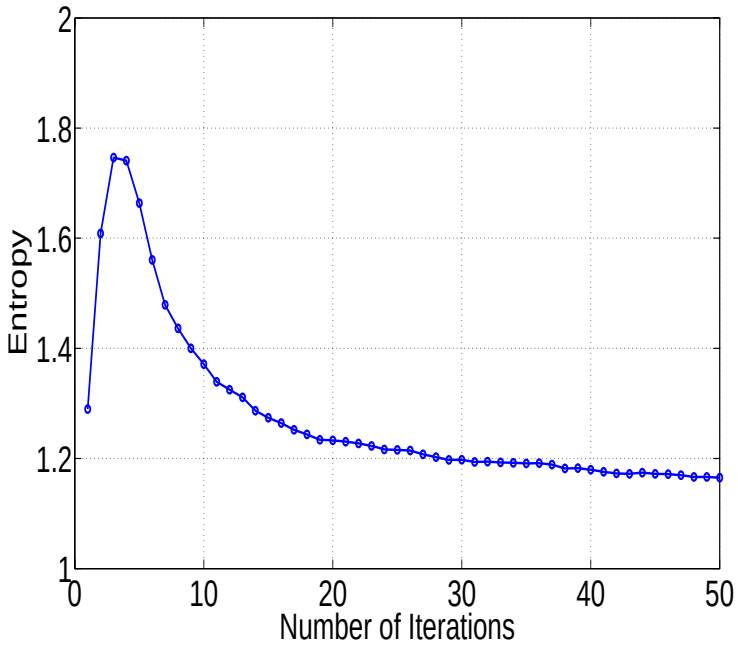

Figure 6.12: Variation in entropy with number of iterations for weighted directed network.

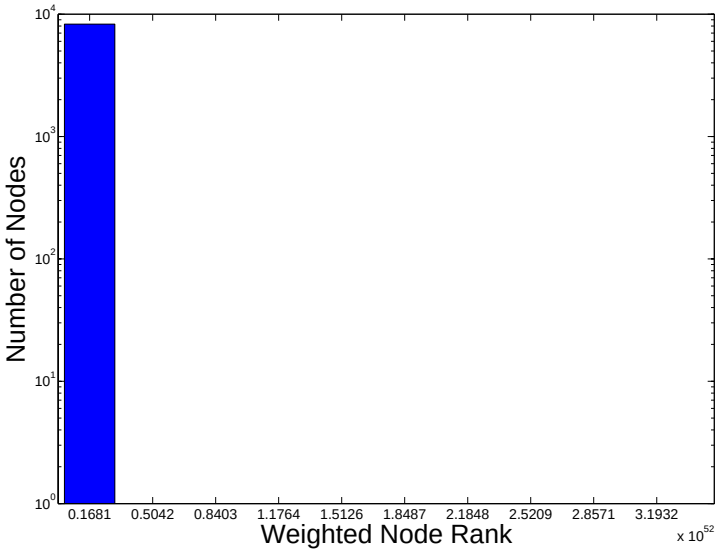

Figure 6.13: Weighted node rank.

Table 6.2: Iloss for different types of networks

Network	**Iloss**
Undirected Unweighted	1428
Directed Unweighted	1793
Undirected Weighted	1451
Directed Weighted	1791

Table 6.3: Iloss for different types of networks using betweenness centrality

Network	**Iloss**
Undirected Unweighted	1442
Undirected Weighted	1411

Table 6.4: Iloss for various values of a

a	**Iloss (Unweighted Undirected)**	**Iloss (Weighted Undirected)**
0.1	1455	1410
0.2	1480	1410
0.3	1498	1413
0.4	1521	1418
0.5	1528	1422
0.6	1537	1422
0.7	1540	1422
0.8	1551	1424
0.9	1553	1426

Table 6.5: Node merging

	Original		**Merged**	
	Nodes	Edges	Nodes	Edges
Undirected	7115	103689	6093	58426
Directed	7115	103689	6109	54579

6.4.4 Formation of Super Node

After the anonymization is complete, many nodes with the same label exist in the network. The nodes can be merged to generalize and remove redundancy while maintaining the structure of the network. The main aim of merging nodes is to group as many similar nodes as possible such that attacks such as neighborhood and sensitive edge disclosure are not possible. Table 6.5 lists the counts of nodes and edges before and after merging.

6.5 Conclusion

Protecting privacy is an important problem associated with social networks. Nodes must be anonymized to ensure privacy. This chapter presents a generalization technique used to anonymize social network data in a hostile environment and maintain a suitable level of privacy while retaining the utility of the data. This can be ensured with the power law distribution of the masked nodes. Four types of networks were considered. The generalization can be tailored according to the sensitivity of the nodes to ensure privacy. The information loss metric, *iloss*, is used to verify the amount of information lost due to generalization. The proposed approach assumes that the viewers are of a static trust level.

Chapter 7

Conclusions and Future Works

7.1 Summary of Contributions of the Thesis

In this thesis, we investigate the various privacy preserving data publishing techniques and study on the privacy and utility issues of the existing mechanisms. We proposed various mechanisms to improve the utility of the data while still preserving the privacy. Contributions of the thesis are summarized as follows.

Before the release of any data product, there is a need to consider a balance between utility and privacy. It is observed that the algorithms that achieve k-anonymity has tradeoff between utility and privacy. An improved heuristic is proposed for achieving k-anonymity. The proposed algorithm has a better balance between computing time and information loss. The proposed algorithm was compared with the established Datafly and Samarati's algorithms. The benchmarked datasets namely, Adult dataset and CUPS datasets, are used for experiments. On balance, the new greedy algorithm performed better than the established ones.

In order to achieve utility and privacy best value of k is required. A scheme is proposed to determining k that has a balance between utility and privacy.

The k-anonymity scheme suffers from homogeneity attack. As a result, the l-diverse scheme was developed. The l-diversity scheme suffers from background knowledge attack. To address this problem, t-closeness scheme was proposed. The draw back with this

scheme is that, the distance metric deployed in constructing a table, satisfying t-closeness, does not satisfy the distance characteristics. In this thesis, we have deployed an alternative distance metric for constructing the t-closeness table. In addition to this, an improved t-closeness was proposed to preserve the identity disclosure.

The k-anonymity, l-diversity and t-closeness schemes can be used to anonymize the dataset before publishing. This is generally in a static environment. There are situations where data need to be published in a dynamic environment like social network. Anonymizing social networks poses great challenges. In this thesis, we propose a scheme to anonymize the users depending on their prominence. Prominence of a node was decided by the centrality and prestige measures.

7.2 Scope for Future Work

The proposed improved greedy heuristic algorithm for balancing utility and privacy is confined only to the full domain generalization with local recording. As a scope of future work, efficient and better heuristics and optimization techniques can be applied to determine the optimal node in the generalization lattice. These strategies may also be investigated with the full domain generalization with local recording, partial domain generalization with local recording and partial domain generalization with global recording.

Further research can also be focused in the direction of deterining global methods to find the best value of k, l and t values. In addition to this, always the quasi-identifiers are assumed within the given set of variables of table. In real-life scenario, it is very difficult to find which attribute would be a potential quasi-identifier. Determining all the possible potential quasi-identifiers for a table and be researched further.

New distance measures can be used in finding t-closeness for balancing the utility and privacy. In this thesis only nominal attributes were considered. As a scope for future work, suitable categorical distance measures are yet to be fitted for improving the utility of the data and preserving privacy.

New centrality measures can be adopted for deciding the improtance of the user in

online social networks. Research can also be focused in the direction of reducing the time for anonymizing the social networks.

Bibliography

Aditya B. and Vijayalakshmi A., *Information Systems Security*, Advances in Database System, LNCS, Springer, Vol. 4332, 2006.

Adult Dataset of UCI Machine Learning Databases, 1996.
ftp://ftp.ics.uci.edu/pub/machine-learning-databases/adult.

Aggarwal C. C. and Wang H., *Managing and mining graph data*, Advances in Database System, Springer, Vol.40, pp. 422–449, 2010.

Aggarwal C. C. and Yu P. S., *Privacy preserving data mining: Models and algorithm*, Advances in Database System, Springer, Vol. 34, 2008.

Aggarwal G., Feder T., Kenthapadi K., Motwani R., Panigrahy R., Thomas D. and Zhu A., *Anonymizing tables*, Proceedings of 10th International Conference on Database Theory, Edinburgh, UK, Springer, pp. 246–258, 2005.

Alireza A., Liaquat H. and Loet L., *Betweenness centrality as a driver of preferential attachment in the evolution of research collaboration networks*, Journal of Informetrics, Vol. 6, No. 3, pp. 403–412, 2012.

Asuncion A. and Newman D. J., UCI Machine Learning Repository, 2007.
http://archive.ics.uci.edu/ml/.

Australia, The Australian Privacy Charter,
Australian Privacy Charter Group, Law School, University of New South Wales, Sydney, 1994.

Backstrom L., Dwork C. and Kleinberg J. M., *Anonymized social network hidden pattern and structural steganography*, Proceedings of 16th International Conference on World Wide Web, Alberta, Canada, ACM, pp. 181–190, 2007.

Bavelas A., *A mathematical model for group structures*, Human Organization, Vol. 7, No.1, pp. 16–30, 1948.

Bavelas A., *Communication patterns in task oriented groups*, Journal of Acoustical Society of America, Vol. 22, No. 6, pp. 271–282, 1950.

Bayardo B. and Agrawal R., *Data privacy through optimal k-anonymity*, Proceedings of the 21st International Conference on Data Engineering, Tokyo, Japan, IEEE, pp. 217–228, 2005.

Bellaachia A. and Anasse B., *SFLOSCAN: A Biologically-Inspired Data Mining Framework for Community Identification in Dynamic Social Networks*, IEEE Symposium Series on Computational Intelligence, Swarm Intelligence, IEEE, pp.1-8, 2011.

Benjamin C. M. F., Ke W., Rui C. and Yu P. S., *Preserving data publishing: A survey of recent developments*, ACM Computing Surveys, Vol. 43, No. 4, pp. 1–53, 2010.

Benjamin C. M. F., Ke W., Ada W. and Yu P. S., *Introduction to Privacy Preserving Data Publishing Concepts and Techniques*. CRC, 2011.

Bercic B. and Carlisle E. G., *Identifying personal data using relational database design principles*, International Journal of Law and Information Technology, Vol. 17, No. 3, pp. 233–251, 2009.

Bin Z. and Jian P., *Preserving privacy in social networks against neighborhood attacks*, Proceedings of IEEE 24th International Conference on Data Engineering, IEEE, pp. 506–515, 2008.

Calcutt D., *Report of the committee on privacy and related matters*, HMSO, 1990.

Chao W., Yike G. and Zhou B., *Social networking federation: A position paper*, Computers and Electrical Engineering, Elsevier, Vol. 38, No. 2, pp. 306–329, 2012.

Ciriani V., De C., Vimercati S., Foresti S. and Samarati P., *k-anonymity. secure data management in decentralized systems*, Advances in Information Security, Springer, Vol. 33, pp. 323–353, 2007.

Comaniciu D., Ramesh V. and Meer P., *Kernel-based object tracking*, Transactions on Pattern Analysis and Machine Intelligence, IEEE, Vol. 25, No. 5, pp. 564–577, 2003.

Cormen T. H., Leiserson C. E. and Rivest T. L., *Introduction to Algorithms*, MIT Press, Vol. 1, 2001.

CUPS: Ismail P. E., KDD CUP Data, 1998.
http://kdd.ics.uci.edu/databases/kddcup98/kddcup98.html.

Curminati B., Ferrari E. and Perego A., *Private relationship in social networks*, Proceedings of 23rd International Conference on Data Engineering Workshop, Istanbul, Turkey, IEEE, pp. 163–171, 2007.

Dalenius T., *An Empirical Study of the Problem of Quasi-identifiers*, Technical Report, Brown University, 1986.

Derpanis K. G., *The bhattacharyya measure*, Mendeley Computer, Vol. 1, No. 4, pp. 1990–1992, 2008.

DM Moratorium Act of the Senate of the United States 108th Congress, 2003,
http://thomas.loc.gov/cgi-bin/query/z?c108:S.188.

Domingo-Ferrer J. and Vicent T., *Risk assessment in statistical microdata protection via advanced record linkage*, Journal of Statistics and Computing, Vol. 13, No. 1, pp. 343–354, 2003.

Duan Y., Wang J., Kam M. and Canny J., *Privacy preserving link analysis on weighted graph*, Computation and Mathematical Organisation Theory, Vol. 11, No. 2, pp. 141–159, 2005.

Emam K. E., Dankar F. K., Issa R., Jonke R. E., Amyot E., Cogo E., Corriveau J. P., Walker M., Chowdhury S., Vaillancourt R., Roffey T. and J. Bottomley., *A globally optimal* k-*anonymity method for de-identification of health data*, Journal of American Medical Informatics Association, Vol. 16, No. 5, pp. 670–682, 2009.

Freeman L. C., *Centrality in social networks: Conceptual clarification*, Social Networks, Vol. 1, pp. 215–239, 1979.

Gautam D., Nick K., Manos P. and Sushruth P., *Efficient sampling of information in social networks*, Proceedings of ACM workshop on Search in social media, California, pp. 67–74, 2008.

George T. D., Thomas B. J. and Virginia A. D. W., *Private Lives and Public Policies*, National Academy Press, 1993.

George T. D., Mark E. and Salazar J. J., *Statistical Confidentiality: Principles and Practice*, Springer, 2011.

Gionis A. and Tassa T., *k-anonymization with minimal loss of information*, Transactions on knowledge and data engineering, IEEE, pp. 206–219, 2009.

Gross R., *Re-identifying facial images*, Technical report, Institute for Software Research International, Carnegie Mellon University, 2005.

Hay M., Mekalu G., Jenson D., Weis P. and Srivastava S., *Anonymizing social networks - a technical report*, Technical report, University of Massachusate, MA, 2007.

Hay M., Mikalu G., Jensen D. and Weis P., *Resisting structural re identification in an anonymized social network*, VLDB Endowment, Vol. 1, No. 1, pp. 102-114, 2008.

Huangmao Q., Ana M., Slobodan V. and Jie W., *A connectivity-based popularity prediction approach for social networks*, Proceedings of IEEE International Conference on Communications, IEEE, pp. 2098-2102, 2012.

Human Rights Law, The universal declarative of human rights, 1948. http://www.un.org/en/documents/udhr/.

Iyengar V., *Transforming data to satisfy privacy constraints*, Proceedings of Eighth ACM SIGKDD international conference on Knowledge discovery and data mining, Edmonton, Canada, ACM, pp. 279–288, 2002.

Jacob G. and Tamir T., *Efficient anonymizations with enhanced utility*, Transactions on Data privacy, Vol. 3, No. 3, pp. 149–175, 2010.

Jaideep V., Christopher W. C. and Yu M. Z., *Privacy Preserving Data Mining*, Springer, New York, 2006.

Jie T., Jimeng S., Chi W. and Zi Y., *Social influence analysis in large-scale networks*, Proceedings of 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, pp. 807–816, 2009a.

Jie T., Yuan Z., Jimeng S., Jinghai R., Wenjing Y., Yiran C., and Fong A. C. M., *Quantitative study of individual emotional states in social networks*, Transactions on Affective Computing, IEEE, Vol. 3, No. 2, pp. 132-144, 2009b.

Jun L. and Meng C. W., *An efficient clustering method for k-anonymization*, Proceedings of the International workshop on Privacy and Anonymity in information society, ACM, pp. 46–50, 2008.

Junqiang L., *Privacy Preserving Data Publishing: Current Status and New Directions*, Information Technology Journal, Asian Network for Scientific Information, Vol. 11, No. 1, pp. 1–8, 2011.

Kohlmayer F., Prasser F., Eckert C., Kemper A. and Klaus A. K., *Flash: Efficient, stable and optimal* k-*anonymity*, Proceedings of ASE/IEEE International

Conference on Privacy, Security, Risk and Trust, Amsterdam, The Netherlands,, IEEE, pp. 708–717, 2012.

Kok S. W. and Myung H. K., *Privacy-preserving frequent itemsets mining via secure collaborative framework*, Security and Communication Networks, Vol. 5, No. 3, pp.263–272, 2012.

Krackhardt D., *Social networks*, Encyclopedia of group processes and intergroup relations, pp. 817–821, 2010.

Krishna A. and Valli K. V., *Attribute based anonymity for preserving privacy*, Proceedings of Second International Workshop on Trust Management in P2P Systems, Sprigner LNCCIS, pp. 572–579, 2011.

Lambert D., *Measures of disclosure risk and harm*, Journal of Official Statistics, Vol.9, No.2, pp. 313–331, 1993.

Lefevre K., Dewitt D. J. and Ramakrishnan R., *Incognito: Efficient full-domain k-anonymity*, Proceedings of ACM SIGMOD, Baltimore, Maryland, USA, ACM, pp. 49–60, 2005.

Leskovec J. and Faloutsos C., *Scalable modeling of real graphs using kronecker multiplication*, Proceedings of International Conference on Machine Learning, Corvallis, OR, ACM, pp. 497–504, 2007.

Liu H. and Maes P., *Interestmap: Harvesting social network profiles for recommendations*, Proceedings of Beyond Personalization Workshop, San Diego, California, USA, January 2005.

Liu K., Kamalika D., Grandison T. and Kargupta H., *Preserving Data Analysis on Graphs and Social Networks*, Next Generation Data Mining, CRC Press, 2008.

Liu L., Jie W., Jinze L. and Jun Z., *Privacy preserving in social networks against sensitive edge disclosure*, Technical Report Technical Report CMIDA-HiPSCCS 006-08, Department of Computer Science, University of Kentucky, KY, 2008b.

Louis B. and Samuel W., *The Right to Privacy*, Harvard Law Review, pp. 193–220, 1890.

Machanavajjhala A., Kifer D., Gehrke J. and Venkatasubramaniam M., *l-diversity: Privacy beyond k-anonymity*, Transactions on Knowledge Discovery of Data, ACM, Vol. 1, No. 3, pp. 1–52, 2007.

Mark H., Eibe F., Geoffrey H., Bernhard P., Peter R. and Ian H. W., *The weka data mining software: An update*, SIGKDD Explorations, Vol. 11, No. 1, pp. 10–18, 2009.

Ming H. and Jian P., *A Survey of Utility-based Privacy-Preserving Data Transformation Methods*, Advances in Database Systems, Springer, Vol. 34, pp. 207–237, 2008.

Mohammad A. K. and Somayajulu D. V. L. N., *A data perturbation method by field rotation and binning by averages strategy for privacy preservation*, Proceedings of the 9th International Conference on Intelligent Data Engineering and Automated Learning, Springer LNCS, pp. 250–257, 2008.

Motwani R. and Xu Y., *Efficient algorithms for masking and finding quasi identifiers*, Conference on Very Large Data Bases, 2007.

Na L., Nan Z. and Sajal K. D., *Preserving relation privacy in online social network data* IEEE Internet Computing, IEEE, Vol. 15, No. 3, pp. 35–42, 2011.

Nergiz M. E. and Christopher C., *δ-presence without complete world knowledge*, Transactions on Knowledge and Data Engineering, IEEE, Vol. 22, No. 6, pp. 868–883, 2010.

Ninghui L., *t-closeness: Privacy beyond k-anonymity and l-diversity*, Proceedings of IEEE 23rd International Conference on Data Engineering, IEEE, pp. 106–115, 2007.

Ninghui L., Tiancheng L. and Venkatasubramaniam S., *Closeness : A new privacy measure for data publishing*, IEEE Knowledge and Data Engineering, IEEE, Vol. 22, No. 7, pp. 943–956, 2007.

Page L., Brin S., Motwani R. and Terry W., The pagerank citation ranking: Bringing order to the web, Technical Report, Stanford University, 1999.

PHIPA, Personal health information protection act, 2004.
www.e-laws.gov.on.ca/navigation?file=home&lang=en.

Pin L., *Utility-Based Anonymization for Continuous Data Publishing*, Proceedings of the 2008 IEEE Pacific-Asia Workshop on Computational Intelligence and Industrial Application, Washington, USA, IEEE, Vol. 2, pp. 290-295, 2008.

Preda R. O. and Vizireanu D. N., *A robust digital watermarking scheme for video copyright protection in the wavelet domain*, Measurement, Vol. 43, No. 10, pp. 1720–1726, Dec 2010.

Preda R. O. and Vizireanu D. N., *A robust wavelet based video watermarking scheme for copyright protection using the human visual system*, Journal of Electronic Imaging, Vol. 20, No, 13022, 2011a.

Preda R. O. and Vizireanu D. N., *Quantization based video watermarking in the wavelet domain with spatial and temporal redundancy*, International Journal of Electronics, Vol. 98, No. 3, pp. 393–405, 2011b.

Purdam K., Elliot M. J. and Mackey E., *The regulation of the personal: Individual data use and identity in the UK*, Journal of policy studies, Vol. 25, No. 4, pp. 267–282, 2004.

Qiang Y., *Three challenges in data mining*, Frontiers of Computer Science in China, Springer, Vol. 4, No. 3, pp. 324–333, 2011.

Qiang Y. and Xindong W., *10 challenging problems in data mining research*, International Journal of Information Technology and Decision Making, Vol. 5, No. 4, pp. 597–604, 2006.

Rajput D. S., Ramjeevan S. T. and Thakur G. S., *Rule generation from textual data by using graph based approach*, International Journal of Computer Science, Vol. 31, No. 9, pp. 36–43, 2011.

Ralph G. and Alessandro A., *Information revelation and privacy in online social networks (the facebook case)*, ACM Workshop on Privacy in the Electronic Society, ACM, pp. 71–80, 2005.

Raymond C. W. W., Ada W. C. F., Ke W. and Jian P., *Anonymization-based attacks in privacy-preserving data publishing*, Transactions on Database Systems, ACM, Vol. 34, No. 2, pp. 8–46, 2009.

Reiss S. P., *Practical data-swapping: The first steps*, Transactions on Database Systems, ACM, Vol. 9, No. 1, pp. 20–37, 1984.

Rob H. and Stephen E. F., *Privacy-preserving record linkage*, Privacy in Statistical Databases, Springer, pp. 269–283, 2010.

Robert E. S., *Ben Franklin's Web Site: Privacy and Curiosity from Plymouth Rock to the Internet*, Privacy Journal, 2004.

Rosenbaum P. R. and Rubin D. B., *The central role of the propensity score in observational studies for causal effects*, Biometrika, Elsevier, Vol. 70, No. 2, pp. 41–55, 1983.

Sabidussi G., *The centrality index of a graph*, Psychometrika, Vol. 31, No. 3, pp. 581–603, 1966.

Samarati P., *Protecting respondents' identity in microdata release*, IEEE Transactions on Knowledge and Data Engineering, IEEE, Vol. 13, No. 6, pp. 1010–1027, 2001.

Samarati P. and Sweeney L., *Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression*, Proceedings of the IEEE Symposium on Research in Security and Privacy, Oakland, CA, IEEE, 1998a.

Samarati P. and Sweeney L., *Generalizing data to provide anonymity when disclosing information*, Proceedings of the seventeenth ACM SIGACT-SIGMOD-SIGART symposium on Principles of database systems, Seattle, WA, USA, ACM, pp. 188–189, 1998b.

Scott J., *Social Network Analysis*, MIT, Cambridge,Mass, 1991.

Simon D., *Big Brother: Britain's web of surveillance and the new technological order*, Pan, London, pp. 23, 1996.

Stanford University, Wikipedia vote network, 2008.
http://snap.stanford.edu/data/wiki-Vote.html.

Steffen E., *Asymptotic normality of integer compositions inside a rectangle*, CoRR, 2012a.

Steffen E., *Stirlingś approximation for central polynomial coefficients*, CoRR, 2012b.

Sweeney L., *Guaranteeing anonymity when sharing medical data, the datafly system*, Proceedings of AMIA Annual Fall Symposium, Washington, DC, pp. 1–5, 1997.

Sweeney L., *Datafly: A system for providing anonymity in medical data*, Proceedings of 11th International conference on Database Security XI: Status and Prospects, pp. 356–381, 1998.

Sweeney L., *k-anonymity: a model for protecting privacy*, International Journal on Uncertainty, Fuzziness and Knowledge based Systems, World Scientific, Vol. 10, No. 5, pp. 557–570, 2002a.

Sweeney L., *Computational Disclosure Control for Medical Microdata: The Datafly System*, National Academy Press, 2007.

Sweeney L., *Achieving k-anonymity privacy protection using generalization and suppression*, International Journal on Uncertainty, Fuzziness and Knowledge-based Systems, World Scientific, Vol. 10, No. 5, pp. 571–588, 2002b.

Travers T. and Milgram S., *An experimental study of the small world problem*, Sociometry, Vol. 32, pp. 425–443, 1969.

Tripathy B. K. and Panda G. K., *A new approach to manage security against neighborhood attacks in social networks*, Proceedings of International Conference on Advances in Social Networks Analysis and Mining, pp. 264–269, 2010.

Truta T. M., Campan A., Abrinica M. and Miller J., *A comparison between local and global recoding algorithms for achieving microdata psensitive k-anonymity*, Acta Universitatis Apulensis, Vol. 15, No. 15, pp. 213–233, 2008.

U.S. Department of Health and Human Services, Health insurance portability and accountability act, 1996.
www.hhs.gov/ocr/privacy/

Verma M. and Toshniwal D., *Privacy preserving sequential pattern mining in progressive databases using noisy data*, Proceedings of 13th International Conference on Information Visualisation, IEEE, pp. 456–460. 2009.

Verykios V. S., Bertino E., Fovino I. N., Provenza L. P., Saygin Y. and Theodoridis Y., *State of the art in privacy preserving data mining*, ACM SIGMOD Record, Vol. 33, No. 1, pp. 1–57, 2004.

Wasserman S. and Faust K., *Social Network Analysis Method and Application*, Cambridge University Press, 1 edition, 1994.

Weisstein E. W., *Multinomial series*, MathWorld- A Wolfram Web Resource, 2009.

Woo M. J., Jerome P. R., Anna O. and Alan F. K., *Global measures of data utility for microdata masked for disclosure limatation*, CYLAB, The Journal of Privacy and Confidentiality, Vol. 1, 2009.

Xiao X. and Tao Y., *Personalized privacy preservation*, SIGMOD Conference, Chicago, IL, USA, ACM, pp. 229–240, 2006.

Xu J., Wei W., Pei J., Wang X., Shi B. and Fu A. W., *Utility-based anonymization using local recoding*, 12th ACM SIGKDD international conference on Knowledge discovery and data mining, Philadelphia, USA, ACM, pp. 785–790, 2006.

Ying X. and Wu X., *Randomizing social networks*, International Conference on Data Mining, Atlanta GA, pp. 739–750, 2008.

Yung M. L. and Lin S. Y., *A diffusion mechanism for social advertising over microblogs*, Decision Support Systems, Vol. 54, No. 1, pp. 9-22, 2012.

Zheleva E. and Getoor L., *Preserving the privacy of the sensitive relationship in graph data*, International Workshop on Privacy, Security and Trust in KDD, San Jose CA, 2007.

Zhou B. and Pei J., *Preserving privacy in social networks*, 24th International Conference on Data Engineering(ICDE '08), Canun, Mexico, pp. 506–515, 2008.

Publications out of the work

International Journals

1. **Korra Sathya Babu**, Sanjay Kumar Jena, Nitin Reddy Ranabothu, Nitesh Kumar and Mark Elliot, Achieving k-anonymity Using Improved Greedy Heuristics for Relational Databases, **Transactions on Data Privacy**, IIIA-CSIC, UNESCO Chair in Data Privacy. (Accepted on 25^{th} January 2013)

2. **Korra Sathya Babu** and Sanjay Kumar Jena, *Anonymizing Social Networks: A Generalization Approach*, **Computers & Electrical Engineering**, Elsevier, DOI: 10.1016/ j.compeleceng.2013.01.020. (Accepted on 20^{th} January 2013)

3. **Korra Sathya Babu** and Sanjay Kumar Jena, *Enhancing* t*-closeness for Utility and Privacy*, **International Journal of Trust Management and Computer Communications**, Inderscience Publishers. (Accepted on 08^{th} January 2013)

4. **Korra Sathya Babu**, Sanjay Kumar Jena and Jalaka Hota, *Privacy Preserving Social Networking*, **International Journal of Computational Engineering**, Inderscience Publishers. (Accepted on 05^{th} May 2012 and article in press)

International Conference

1. **Korra Sathya Babu** and Sanjay Kumar Jena, *Balancing Utility and Privacy for k-anonymity*, Proceeding of the 19^{th} International Workshop on Identity: Security, Management & Applications, (ID 2011), July 22-24, Kochi, Kerala, Springer LNCCIS, pp. 1-8, 2011, DOI: 10.1007/978-3-642-22714-1_1.

Other publications of the author

International Journals

1. **Korra Sathya Babu** and Sanjay Kumar Jena, *Privacy Preservation by Quantization*, **International Journal of Data Mining Knowledge Engineering**, Vol. 5, No. 3, 2012, Pages 139-158, 2012, DOI: DMKE032012004.

Book Chapter

1. Saroj Kumar Panigrahy, Debasish Jena, **Korra Sathya Babu** and Sanjay Kumar Jena, *On the Privacy Protection of Biometric Traits: Palmprint, Face and Signature*, Edition 1 of Communications in Computer and Information Science, Vol. 40, Contemporary Computing, Part 4, Springer, pp. 182-193, 2009, DOI: 10.1007/978-3-642-03547-0_18.

2. Kumar Dhiraj, Santanu Kumar Rath and **Korra Sathya Babu**. *FCM for Gene Expression Bioinformatics Data*, Edition 1 of Communications in Computer and Information Science, Vol. 40, Contemporary Computing, Part 10, Springer, pp. 521-532, 2009, DOI: 10.1007/978-3-642-03547-0_50.

3. **Korra Sathya Babu**, Saroj Kumar Panigrahy and Sanjay Kumar Jena. *Web Usage Mining: An Implementation View*, Edition 1 of Communications in Computer and Information Science, Advances in Computing, Communication and Control, Part 1, Vol. 125, pp. 131-136, 2011, DOI: 10.1007/978-3-642-18440-6_16.

International Conference

- Khaitan Pranav, M Satish Kumar, **Korra Sathya Babu** and Sanjay Kumar Jena. *Improved Query Plans for Unnesting Nested SQL Queries*, Proceedings of 2nd International Conference on Computer Science and its Applications, 2009. CSA '09, South Korea, Dec. 10-12, IEEE, pp. 1-6, 2009, DOI: 10.1109/CSA.2009.5404288.

- Khaitan Pranav, **Korra Sathya Babu**, Sanjay Kumar Jena and Banshidhar Majhi, *Approximation Algorithms for Optimizing Privacy and Utility*, Proceedings of 2nd International Conference on Computer Science and its Applications, 2009. CSA '09, South Korea, Dec. 10-12, IEEE, pp. 1-6, 2009, DOI: 10.1109/CSA.2009.5404306.

- **Korra Sathya Babu** and Sanjay Kumar Jena, *Prevention of Malicious Transaction in DBMS*, Proceedings of Emerging Trends in Statistical Methods and Optimization Techniques, University of Jammu, Jammu Feb. 22-23, pp. 85-90, 2008.