Chapter 10

# PRIVACY IN SOCIAL NETWORKS: A SURVEY

Elena Zheleva
*Department of Computer Science*
*University of Maryland*
*College Park, MD 20742, USA*
elena@cs.umd.edu

Lise Getoor
*Department of Computer Science*
*University of Maryland*
*College Park, MD 20742, USA*
getoor@cs.umd.edu

**Abstract** In this chapter, we survey the literature on privacy in social networks. We focus both on online social networks and online affiliation networks. We formally define the possible privacy breaches and describe the privacy attacks that have been studied. We present definitions of privacy in the context of anonymization together with existing anonymization techniques.

**Keywords:** privacy, social networks, anonymization, disclosure

## 1. Introduction

With the proliferation of online social networks, there has been increasing concern about the privacy of individuals participating in them. While disclosing information on the web is a voluntary activity on the part of the users, users are often unaware of who is able to access their data and how their data can potentially be used. Data privacy is defined as "freedom from unauthorized intrusion" [28]. However, what constitutes an unauthorized intrusion in social networks is an open question. Because privacy in social networks is a young field, in this chapter, we mainly aim at identifying the space of problems in this emerging area rather than proposing solutions. We present existing work

C. C. Aggarwal (ed.), *Social Network Data Analytics,*
DOI 10.1007/978-1-4419-8462-3_10, 

when appropriate, but many of these problems have not yet been addressed in the research literature. One of the contributions of this chapter is in cataloging the different types of privacy disclosures in social networks. These are studied in the research literature but they are often not explicitly formalized.

We focus on two scenarios for privacy in social networks: privacy breaches and data anonymization. In the first scenario, an adversary is interested in learning the private information of an individual using publicly available social network data, possibly anonymized. In the second scenario, a data provider would like to release a social network dataset to researchers but preserve the privacy of its users. For this purpose, the data provider needs to provide a privacy mechanism, so that researchers can access the (possibly perturbed) data in a manner which does not compromise users' privacy. A common assumption in the data anonymization literature is that the data is described by a single table with attribute information for each of the entries. However, social network data can exhibit rich dependencies between entities which can be exploited for learning the private attributes of users, and we explore the consequences of this possibility.

In Section 2, we discuss the different types of privacy breaches: private information that can leak from a social network. We define the types of queries for each type of disclosure, and ways to measure the extent to which a disclosure has occurred in an online or anonymized social network. We are abstracting these definitions from the types of privacy breaches that have been studied in data anonymization. The definitions can be applied both in the anonymization scenario and in the scenario of an intrusion in an online social network. We also provide pointers to work which studies these privacy breaches in the context of anonymization. We present privacy definitions in Section 3 and privacy mechanisms for publishing social network data in Section 4.

In the context of this book chapter, when we refer to social networks, we generally mean online social networks. This includes online sites such as Facebook, Flickr, LinkedIn, etc., where individuals can link to, or "friend," each other, and which allow rich interactions such as joining communities or groups of interest, or participating in discussion forums. These sites often also include online services which allow users to create profiles and share their preferences and opinions about items, such as tagging articles and postings, commenting on photos, and rating movies, books or music. Thus, we view a social network as a multi-modal graph in which there are multiple kinds of entities, including people, groups, and items, but where at least one type is an individual and the links between individuals represent some sort of social tie. Each node of an individual has a profile, and profiles can have personal attributes, such as age, gender, political affiliation, etc. Our view is informed by the link mining and statistical relational learning research communities [22, 23], which study the mining of interconnected relational data.

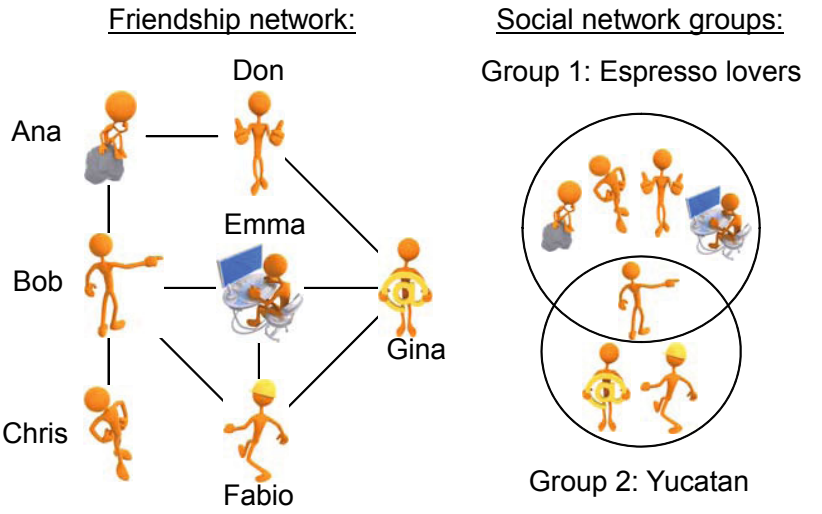


*Figure 10.1.* A toy social network.

We concentrate on networks which have two types of commonly occurring links - user-user links, and user-group links. More formally, we represent the social network as a graph $G = (V, E_v, H, E_h)$, where $V$ is a set of $n$ nodes which represent user profiles, such as the one in Figure 10.2. Each node can have a set of properties $v.A$. An edge $e_v(v_i, v_j) \in E_v$ is a *social link* and represents a relationship between the nodes $v_i$ and $v_j$ such as friendship. Relationships can be of different types (such as in a multiplex network), and there can be more than one relationship between a pair of users. We use $H$ to denote both formal online groups and other online content for which users have preference, such as photos, movies, fan pages, etc. We refer to $H$ as affiliation groups. An edge $e_h(v_i, h_j) \in E_h$ represents an *affiliation link* of the membership of node $v_i$ to affiliation group $h_j$. Social links, affiliation links and groups also can have attributes, $e_v.A$, $e_h.A$ and $h.A$, respectively. We also define $P$ to be a set of real-world entities which represent actual people.

As a running example, we consider the social network presented in Figure 10.1. It consists of seven profiles which describe a collection of individuals (Ana, Bob, Chris, Don, Emma, Fabio, and Gina), along with their friendship links and their affiliation groups of interest. Users are linked by a friendship link, and in this example they are reciprocal. There are two groups that users can participate in: the "Espresso lovers" affiliation group and the "Yucatan" affiliation group. These individuals also have personal attributes on their profiles: name, age, gender, zip code and political views (see Figure 10.5 on page 288). User-group affiliations can also be represented as a bipartite graph, such as the ones in Figure 10.6 (page 298) and Figure 10.7(a) (page 299).

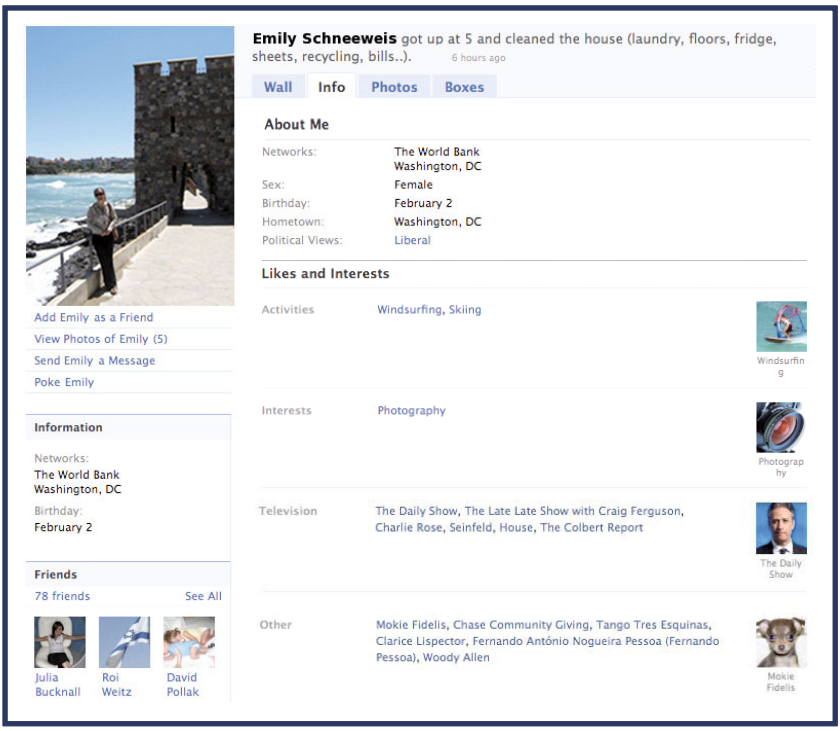


*Figure 10.2.* A hypothetical Facebook profile.

In this chapter, we focus on both privacy breaches in online social networks and privacy-preserving techniques for publishing social network data. In addition, there are other existing surveys on privacy preservation in social networks that focus on different aspects [12, 25, 34, 47, 52]. The surveys on privacy-preserving data publication for networks cover privacy attacks, edge modification, randomization and generalization privacy-preserving strategies for network structure [34, 47, 52] and richer graphs [47]. Clarkson et al. [12] discuss anonymization techniques which aim to prevent identity disclosure. The survey of Hay et al. [25] concentrates on privacy issues with network structure, and it covers attacks and their effectiveness, anonymization strategies, and differential privacy for private query answering.

## 2. Privacy breaches in social networks

When studying privacy, it is important to specify what defines a failure to preserve privacy. A *privacy breach* occurs when a piece of sensitive information about an individual is disclosed to an adversary, someone whose goal is to compromise privacy. Traditionally, two types of privacy breaches have

been studied: *identity disclosure* and *attribute disclosure*. We discuss these two types in the context of social networks. We also present two more disclosure types, specific to network data: *social link disclosure* and *affiliation link disclosure*.

## 2.1 Identity disclosure

Identity disclosure occurs when an adversary is able to determine the mapping from a profile $v$ in the social network to a specific real-world entity $p$. Before we are able to provide a formal definition of identity disclosure, let us consider three questions related to the identity of $p$ in which an adversary may be interested.

DEFINITION 10.1 **Mapping query**. *In a set of individual profiles $V$ in a social network $G$, find which profile $v$ maps to a particular individual $p$. Return $v$.*

DEFINITION 10.2 **Existence query**. *For a particular individual $p$, find if this individual has a profile $v$ in the network $G$. Return true or false.*

DEFINITION 10.3 **Co-reference resolution query**. *For two individual profiles $v_i$ and $v_j$, find if they refer to the same individual $p$. Return true or false.*

A simple way of defining *identity disclosure* is to say that the adversary can answer the *mapping query* correctly and with full certainty. However, unless the adversary knows unique attributes of individual $p$ that can be matched with the observed attributes of profiles in $V$, this is hard to achieve. One way of formalizing *identity disclosure* for an individual $p$ is to assoicate a random variable $\hat{v}_p$ which ranges over the profiles in the network. We assume that the adversary has a way of computing the probability of each profile $v_i$ belonging to individual $p$, $Pr(\hat{v}_p = v_i)$. In addition, we introduce a dummy profile $v_{dummy}$ in the network which serves the purpose of absorbing the probability of individual $p$ not having a profile in the network. We assume that $p$ has exactly one profile, and the true profile of $p$ in $V \cup \{v_{dummy}\}$ is $v_*$. We use the shorthand $Pr_p(v_i) = Pr(\hat{v}_p = v_i)$ to denote the probability that $v_i$ corresponds to $p$; $Pr_p$ provides a mapping $Pr_p : V \cup \{v_{dummy}\} \rightarrow \mathbb{R}$. We leave it open as to how the adversary constructs $Pr_p$. Then we can define *identity disclosure* as follows:

DEFINITION 10.4 **Identity disclosure with confidence** $t$. *In a set of individual profiles $V$ in a social network $G$, identity disclosure occurs with confidence $t$ when $Pr_p(v_*) \geq t$ and $v_* \neq v_{dummy}$.*

An alternative definition of *identity disclosure* considers that the possible values of $v_i$ can be ranked according to their probabilities.

DEFINITION 10.5 **Identity disclosure with *top-k* confidence**. *In a set of individual profiles $V$ in a social network $G$, identity disclosure occurs with* top-k *confidence when $v_*$ appears in the top $k$ profiles (or top $p\% = k * 100/|V|$), in the list of profiles ranked by $Pr_p$ from high to low.*

The majority of research in social network privacy has concentrated on identity disclosure [4, 8, 26, 27, 30, 35, 41, 45, 48, 51, 53]. We discuss it in more detail in Section 4.

## 2.2 Attribute disclosure

A common assumption in the privacy literature is that there are three types of possibly overlapping sets of personal attributes:

- Identifying attributes - attributes, such as social security number (SSN), which identify a person uniquely.
- Quasi-identifying attributes - a combination of attributes which can identify a person uniquely, such as name and address.
- Sensitive attributes - attributes that users may like to keep hidden from the public, such as politic affiliation and sexual orientation.

Attribute disclosure occurs when an adversary is able to determine the value of a sensitive user attribute, one that the user intended to stay private. This attribute can be an attribute of the node itself, the node's links or the node's affiliations. Without loss of generality, here we discuss the attributes of the node itself. Again, to make this definition more concrete, we assume that each sensitive attribute $v.a_s$ for profile $v$ has an associated random variable $v.\hat{a}_s$ which ranges over the possible values for $v.a_s$. Let the true value of $v.a_s$ be $v.a_*$. We also assume that the adversary can map the set of possible sensitive attribute values to probabilities, $Pr_a(v.\hat{a}_s = v.a) : v.a \rightarrow \mathbb{R}$, for each possible value $v.a$. Note that this mapping can be different for each node/profile. Now, we can define attribute disclosure as follows:

DEFINITION 10.6 **Attribute disclosure with confidence** $t$. *For a profile $v$ with a hidden attribute value $v.a_s = v.a_*$, attribute disclosure occurs with confidence $t$ when $Pr_a(v.\hat{a}_s = v.a_*) \geq t$.*

Similarly to *identity disclosure*, there is an alternative definition of *attribute disclosure* which considers that the possible values of $v.A_s$ can be ranked according to their probabilities.

DEFINITION 10.7 **Attribute disclosure with *top-k* confidence**. *For a profile $v$ with a hidden attribute value $v.a_s = v.a_*$, attribute disclosure occurs with*

top-k *confidence when $a_*$ appears in the top $k$ values of the list of possible values ranked by their probabilities $Pr_a$.*

Clearly, if an adversary can see the identifying attributes in a social network, then answering the identity *mapping query* becomes trivial, and identity disclosure with confidence 1 can occur. For example, if a profile contains a SSN, then identifying the real person behind the profile is trivial since there is a one-to-one mapping between individuals and their social security numbers. Therefore, in order to prevent identity disclosure, the identifying attributes have to be removed from the profiles.

Sometimes, a combination of attributes, referred to as *quasi-identifying attributes*, can lead to identity disclosure. What constitutes *quasi-identifying attributes* depends on the context. For example, it has been observed that 87% of individuals in the U.S. Census from 1990 can be uniquely identified based on their date of birth, gender and zip code [43]. Another example of quasi-identifiers is a combination of a person's name and address.

Similarly, matching records from different datasets with quasi-identifying attributes can lead to further privacy breaches. This is known as a *linking attack*. If the identities of users in one dataset are known and the second dataset does not have the identities but it contains sensitive attributes, then the sensitive attributes of the users from the first dataset can be revealed. For example, matching health insurance records, in which the identifying information is removed, with public voter registration records can reveal sensitive health information about voters. Using this attack, Sweeney was able to identify the medical record of the governor of Massachusetts [43].

In the context of social and affiliation networks, there has not been much work on sensitive attribute disclosure. Most studies look at how attributes can be predicted [40, 33, 50], and very few on how they can be protected [8]. We discuss this work in more detail in Section 4.

## 2.3 Social link disclosure

Social link disclosure occurs when an adversary is able to find out about the existence of a sensitive relationship between two users, a relationship that these users would like to remain hidden from the public. Similarly to the previous types of disclosures, we assume that there is a random variable $\hat{e}_{i,j}$ associated with the link existence between two nodes $v_i$ and $v_j$, and an adversary has a model for assigning a probability to $\hat{e}_{i,j}$, $Pr(\hat{e}_{i,j} = true) : e_{i,j} \rightarrow \mathbb{R}$.

DEFINITION 10.8 **Social link disclosure with confidence** $t$. *For two profiles $v_i$ and $v_j$, a social link disclosure occurs with confidence $t$ when $e_v(v_i, v_j) \in E_v$ and $Pr(\hat{e}_{i,j} = true) \geq t$.*

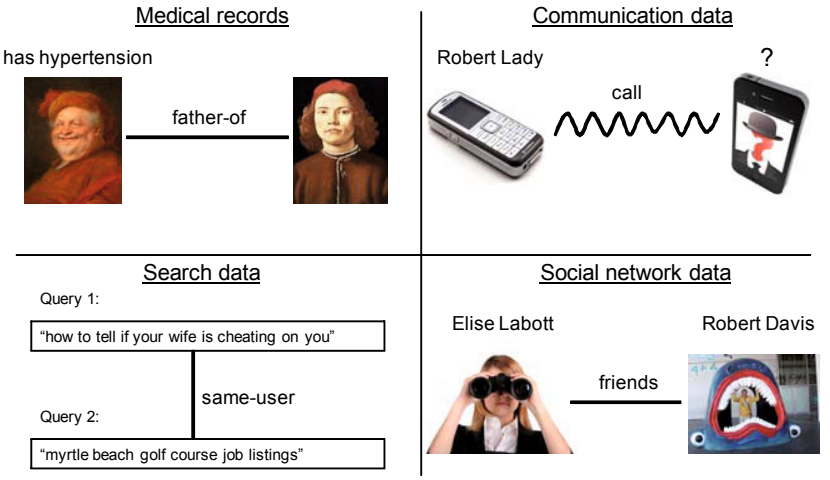


*Figure 10.3.* Sensitive link examples.

Note that since the link existence $\hat{e}_{i,j}$ has only two possible values, true and false, the *top-k* definition does not apply to social link disclosure.

Examples of sensitive relationships can be found in social networks, communication data, disease data and others. In social network data, based on the friendship relationships of a person and the public preferences of the friends such as political affiliation, it may be possible to infer the personal preferences of the person in question as well. In cell phone communication data, finding that an unknown individual has made phone calls to a cell phone number of a known organization can compromise the identity of the unknown individual. In hereditary disease data, knowing the family relationships between individuals who have been diagnosed with hereditary diseases and ones that have not, can help infer the probability of the healthy individuals to develop these diseases. Figure 10.3 presents a summary of these examples.

Researchers have studied attacks that expose sensitive links in social networks [4, 6, 31, 49]. Sensitive edge properties, such as link strength (or weight), have also been the focus of recent work [16, 37].

## 2.4 Affiliation link disclosure

Another type of privacy breach in relational data is *affiliation link disclosure* whether a person belongs to a particular affiliation group. Whether two users are affiliated with the same group can also be of sensitive nature. Sometimes, affiliation link disclosure can lead to attribute disclosure, social link disclosure, or identity disclosure. Thus, hiding affiliations is a key to preserving the privacy of individuals.

As before, we assume that there is a random variable $\hat{e}_{v,h}$ associated with the existence of an affiliation link between a profile $v$ and a group $h$, and that an adversary has a way of computing the probability of $\hat{e}_{v,h}$, $Pr(\hat{e}_{v,h} = true)$ : $e_{v,h} \rightarrow \mathbb{R}$.

DEFINITION 10.9 **Affiliation link disclosure with confidence** $t$. *For a profile $v$ and an affiliation group $h$, an affiliation link disclosure occurs with confidence $t$ when $e_h(v,h) \in E_h$ and $Pr(\hat{e}_{v,h} = true) \geq t$.*

One type of disclosure can lead to another type. For example, Wondracek et al. [45] show a de-identification attack in which affiliation link disclosure can lead to the identity disclosure of a supposedly anonymous Internet user. An adversary starts the attack by crawling a social networking website and collecting information about the online social group memberships of its users. It is assumed that the identities of the social network users are known. According to the collected data, each user who participates in at least one group has a group signature, which is the set of groups he belongs to. Then, the adversary applies a *history stealing attack* (for more details on the attack, see [45]) which collects the web browsing history of the target Internet user. By finding the group signatures of social network users which match the browsing history of the Internet user, the adversary is able to find a subset of potential social network users who may be the Internet user. In the last step of the attack, the adversary looks for a match between the id's of the potential users and the browsing history of the target individual, which can lead to de-identification of the Internet user.

Another example of affiliation link disclosure leading to identity disclosure is in search data. If we assume that users posing queries to a search engine are the individuals in the social network, and the search queries they pose are the affiliation groups, then disclosing the links between users and queries can help an adversary identify people in the network. Users interact with search engines in an uninhibited way and reveal a lot of personal information in the text of their queries. There was a scandal in 2006 when AOL, an Internet Service provider, released an "anonymized" sample of over half a million users and their queries posed to the AOL search engine. The release was well-intentioned and meant to boost search ranking research by supplementing it with real-world data. Each user was specified by a unique identifier, and each query contained information about the user identifier, search query, the website the user clicked on, the ranking of that website in the search results, and the timestamp of the query.

One of the problems with the released data was that even though it was in a table format (Table 10.1), its entries were not independent of each other. Shortly after the data release, New York Times reporters linked 454 search queries made by the same individual which gave away enough personal infor-

*Table 10.1.* A snapshot of the data released by AOL. Here, we are omitting the timestamps included in the data.

| *User ID* | *Search query* | *Clicked website* | *Ranking* |
|---|---|---|---|
| 4417749 | clothes for age 60 | http://www.news.cornell.edu | 10 |
| 4417749 | dog who urinate on everything | http://www.dogdayusa.com | 6 |
| 4417749 | landscapers in lilburn ga. | | |
| 4417749 | pine straw in lilburn ga. | http://gwinnett-online.com | 9 |
| 4417749 | gwinnett county yellow pages | http://directory.respond.com | 1 |
| 4417749 | best retirement place in usa | http://www.amazon.com | 7 |
| 4417749 | mini strokes | http://www.ninds.nih.gov | 1 |

mation to identify that individual – Thelma Arnold, a 62-year old widow from Lilburn, Georgia [5]. Her queries included names of people with the same last name as hers, information about retirement, her location, etc.

Affiliation link disclosure can also lead to attribute disclosure, as illustrated in a *guilt-by-association attack* [14]. This attack assumes that there are groups of users whose sensitive attribute values are the same, thus recovering the sensitive value of one user and the affiliation of another user to the group can help recover the sensitive value of the second user. This attack was used in the BitTorrent file-sharing network to discover the downloading habits of users [11]. Communities were detected based on social links, and monitoring only one user in each community was enough to infer the interests of the other people in the community. In this case the sensitive attribute that users would like to keep private is whether they violate copyrights. This attack has also been applied to identifying fraudulent callers in a phone network [14]. Cormode et al. [13] study data anonymization to prevent affiliation link disclosure. They refer to affiliation links as associations (see Section 4.2).

## 3. Privacy definitions for publishing data

The goal of data mining is discovering new and useful knowledge from data. Sometimes, the data contains sensitive information, and it needs to be sanitized before it is published publicly in order to address privacy concerns. Data sanitization is a complex problem in which hiding private information trades off with utility reduction. The goal of sanitization is to remove or perturb the attributes of the data which help an adversary infer sensitive information. The solution depends on the properties of the data and the notions of privacy and utility in the data.

Privacy preservation in the context of social network data is a relatively new research field. Rather than assuming data which is described by a single table of independent records with attribute information for each, it takes into

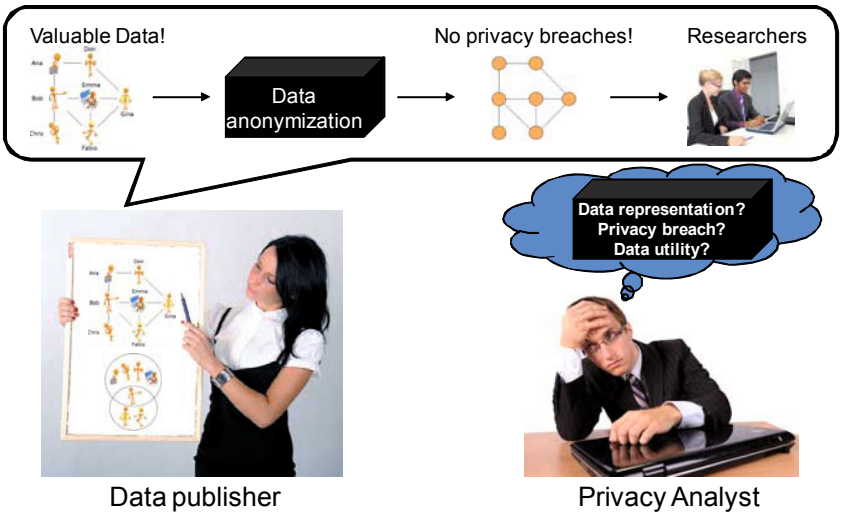


*Figure 10.4.* Anonymization scenario.

consideration more complex real-world datasets. As discussed earlier, relational data, often represented as a multi-graph, can exhibit rich dependencies between entities. The challenge of sanitizing graph data lies in understanding these dependencies and removing sensitive information which can be inferred by direct or indirect means.

One way in which data providers can sanitize data is by anonymization. Figure 10.4 shows a typical scenario in which a data owner is interested in providing researchers with valuable data and in order to meet privacy concerns, she consults a privacy analyst before publishing a perturbed version of the data. In the process of anonymizing the data, the identifying information is removed and other attributes are perturbed. Anonymizing techniques have been criticized as often being ad hoc and not providing a principled way of preserving privacy. There are no guarantees that an adversary would not be able to come up with an attack which uses background information and properties of the data, such as node attributes and observed links, to infer the private information of users. Another way of sanitizing data is by providing a private mechanism for accessing the data, such as allowing algorithms which are provably privacy-preserving to run on it. Next, we will discuss privacy preservation definitions. Some of these definitions were not developed specifically for network data but we provide examples from the social network domain.

To formalize privacy preservation, Chawla et al. [9] proposed a framework based on the intuitive definition that "our privacy is protected to the extent we

| Identifier | Quasi-identifiers | | | Sensitive |
|---|---|---|---|---|
| Name | Age | Sex | Zip | Pol. views |
| Ana | 21 | F | 20740 | liberal |
| Bob | 25 | M | 83222 | liberal |
| Chris | 24 | M | 20742 | liberal |
| Don | 29 | M | 83209 | conservative |
| Emma | 24 | F | 20640 | liberal |
| Fabio | 24 | M | 20760 | liberal |
| Gina | 28 | F | 83230 | liberal |
| Halle | 29 | F | 83201 | conservative |
| Ian | 31 | M | 83220 | conservative |
| John | 24 | M | 20740 | liberal |

*5-anonymity applied to data* →

| Equiv. class | Quasi-identifiers | | | Sensitive |
|---|---|---|---|---|
| | Age | Sex | Zip | Pol. views |
| C1 | [21,24] | * | 20*** | liberal |
| C2 | [25,31] | * | 832** | liberal |
| C1 | [21,24] | * | 20*** | liberal |
| C2 | [25,31] | * | 832** | conservative |
| C1 | [21,24] | * | 20*** | liberal |
| C1 | [21,24] | * | 20*** | liberal |
| C2 | [25,31] | * | 832** | liberal |
| C2 | [25,31] | * | 832** | conservative |
| C2 | [25,31] | * | 832** | conservative |
| C1 | [21,24] | * | 20*** | liberal |

*Figure 10.5.* 5-anonymity applied to data with 10 records.

blend in the crowd." Obviously, with the richness of information in online social network profiles, this is hard to achieve and users are easily identifiable. We will look at a simpler case when a data provider is interested in releasing a dataset with online social network profiles. To give a flavor of existing work, we present four existing privacy preservation approaches which make the definition of "blending in the crowd" more concrete.

## 3.1 *k*-anonymity

$k$-anonymity protection of data is met if the information for each person contained in the data cannot be distinguished from at least $k-1$ other individuals in the data. $k$-anonymity can be achieved by suppressing and generalizing the attributes of individuals in the data. Suppressing an attribute value means deleting it from the perturbed data. Generalizing an attribute means replacing it with a less specific but semantically consistent value. One can see that suppression is a special case of generalization, and that suppressing all attributes would guarantee $k$-anonymity. This is why a notion of utility in the data has to be incorporated whenever sanitizing data. The actual objective is to maximize utility by minimizing the amount of generalization and suppression. Achieving $k$-anonymity by generalization with this objective as a constraint is an NP-hard problem [3]. $k$-anonymity has been studied mostly for table data, so we begin by presenting its definition using only the nodes $V$ and their attributes $V.A$ i.e., disregarding links and affiliation groups.

DEFINITION 10.10 **$k$-anonymity**. *A set of records $V$ satisfies k-anonymity if for every tuple $v \in V$ there exist at least $k-1$ other tuples $v_{i_1}, v_{i_2}, ..., v_{i_{k-1}}$ $V$ such that $v_{i_1}.A_q = v_{i_2}.A_q = ... = v_{i_{k-1}}.A_q$ where $A_q \in A$ are the quasi-identifying attributes of the profile.*

Figure 10.5 shows an example of applying 5-anonymity to the data of 10 individuals. The data includes their names, ages, genders and zip codes. The perturbed data meets a 5-anonymity constraint because each individual is indistinguishable from at least 4 other individuals. Here, the assumption is that name is an identifying attribute, therefore it has been suppressed. Three of the attributes, *Age*, *Sex* and *Zip code*, are quasi-identifiers, therefore, they have been generalized. The sensitive attributes remain the same.

$k$-anonymity provides a clustering of the nodes into equivalence classes such that each node is indistinguishable in its quasi-identifying attributes from some minimum number of other nodes. In the previous example, there were two equivalence classes: class $C1$ of individuals whose age is in the range $[21, 24]$ years and have a zip code $20 * **$, and class $C2$ of individuals whose age is in the range $[25, 31]$ years and have a zip code $832 * *$. Note, however, that these equivalent classes are based on node attributes only, and inside each equivalence class, there may be nodes with different identifying structural properties and edges. This makes it hard to define k-anonymity for nodes in social networks. We discuss some approaches later in Section 4.

$k$-anonymity ensures that individuals cannot be uniquely identified by a linking attack. However, it does not necessarily prevent sensitive attribute disclosure. Here, we present two possible attacks on $k$-anonymized data [38]. The first one can occur when there is little diversity in the sensitive attributes inside an equivalence class. In this case, the sensitive attribute of everyone in the equivalence class becomes known with high certainty. For example, if an adversary wants to figure out Ana's political views knowing that her age is 21 and her zip code is 20740, then he can figure out that her record is in equivalence class $C1$. There is no diversity in the sensitive attribute value of equivalence class $C1$, i.e., everyone in $C1$ has liberal political views, therefore, the adversary is able to infer Ana's political views even though he does not know which row corresponds to her. This is known as the *homogeneity attack* [38].

The second problem with $k$-anonymity is that in the presence of background knowledge, attribute and identity disclosure can still occur. For example, knowing that someone's friends are liberal, makes it highly likely that this person is liberal as well. In our toy example, the knowledge that Gina's friends, Emma and Fabio, belong to equivalence class $C1$ where everyone is liberal, can help an adversary infer with high certainty that Gina is liberal as well. This is known as the *background attack* [38].

There are a number of definitions derived from $k$-anonymity tailored to structural properties of network data. Some examples of such defintions include *k-degree anonymity* [35], *K-Candidate anonymity* [27], *k-automorphism anonymity* [53], *k-neighborhood anonymity* [51, 47], and *(k,l)-grouping* [13].

We introduce the intuition behind them, together with their definitions in Section 4.1.1 and Section 4.2, privacy mechanisms for networks.

## 3.2 $l$-diversity and $t$-closeness

A privacy definition which alleviates the problem of sensitive attribute disclosure inherent to k-anonymity is $l$-diversity [38]. As its name suggests, $l$-diversity ensures that the sensitive attribute values in each equivalence class are diverse.

DEFINITION 10.11 **$l$-diversity**. *A set of records in an equivalence class $C$ is $l$-diverse if it contains at least $l$ "well-represented" values for each sensitive attribute. A set of nodes $V$ satisfy $l$-diversity if every equivalence class $C' \subseteq V$ is $l$-diverse.*

There are a number of ways to define "well-represented." Some examples include using frequency counts and measuring entropy. However, even in the case of $l$-diverse data, it is possible to infer sensitive attributes when the sensitive distribution in a class is very different from the overall distribution for the same attribute. If the overall distribution is skewed, then the belief of someone's value may change drastically in the anonymized data (*skewness attack*) [32]. For example, only $30\%$ of the records in Figure 10.5 have conservative political views. However, in equivalence class $C2$ this number becomes $60\%$, thus the belief that a user is conservative increases for users in $C2$. Another possible attack, known as the *similarity attack* [32], works by looking at equivalent classes which contain very similar sensitive attribute values. For example, if $Age$ is a sensitive attribute and an adversary wants to figure out Ana's age knowing that she is in equivalence class $C1$ (based on her *Zip code*), then he would learn that she is between 21 and 24 years old which is a much tighter age range than the range in the whole dataset.

This leads to another privacy definition, $t$-closeness, which considers the sensitive attribute distribution in each class, and its distance to the overall attribute distribution. The distance can be measured with any similarity score for distributions.

DEFINITION 10.12 **$t$-closeness**. *A set of records in an equivalence class $C$ is $t$-close if the distance between the distribution of a sensitive attribute $A_s$ in $C$ and its distribution in $V$ is no more than a threshold $t$. A set of nodes $V$ satisfy $t$-closeness if every equivalence class $C' \subseteq V$ is $t$-close.*

Just like with $k$-anonymity, sanitizing data to meet either $l$-diversity or $t$-closeness comes with a computational complexity burden. There are other privacy definitions of this flavor but they have all been criticized for being ad hoc. While they guarantee syntactic properties of the released data, they come with no privacy semantics [18].

## 3.3 Differential privacy

The notion of differential privacy was developed as a principled way of defining privacy, so that "the risk to one's privacy [...] should not substantially increase as a result of participating in a database" [17]. This shifts the view on privacy from comparing the prior and posterior beliefs about individuals before and after publishing a database to evaluating the risk incurred by joining a database. It also imposes a guarantee on the data release mechanism rather than on the data itself. Here, the goal is to provide statistical information about the data while preserving the privacy of users in the data. This privacy definition gives guarantees that are independent of the background information and the computational power of the adversary.

Returning to our running example, if the social network data set is released using a differentially private mechanism, this would guarantee that Ana's participation in the social network does not pose a threat to her privacy because the statistics would not look very different without her participation. It *does not* guarantee that one cannot learn sensitive information about Ana using background information but such guarantee is impossible to achieve for any kind of dataset [17].

DEFINITION 10.13 **$\epsilon$-differential privacy**. *A randomized function $K$ satisfies $\epsilon$-differential privacy if for all data sets $D_1$ and $D_2$ differing in at most one element, and any subset $S$ of possible outcomes in $Range(K)$,*

$$P(K(D_1) \in S) \leq exp(\epsilon) \times P(K(D_2) \in S). \tag{10.1}$$

Here, one can think of a profile in the social network as being an element, and $V$ being the data set, thus $D_1 \subseteq V$ and $D_2 \subseteq V$. The randomized function $K$ can be thought of as an algorithm which returns a random variable, possibly with some noise. When developing a differentially private algorithm, one has to keep in mind the utility of the data and incorporate the desired knowledge in the algorithm. $Range(K)$ is the output range of algorithm $K$. A common way of achieving $\epsilon$-differential privacy is by adding random noise to the query answer.

One type of algorithm that has been proven to be differentially private is a $count$ query to which one adds Laplacian noise [20]. For example, if the count query is $K =$*"How many people are younger than 22?"*, then the output range of the query is $Range(K) = \{1, ..., n\}$ where $n$ is the size of the social network. The count query is considered a low-sensitivity query because it has a sensitivity of $\Delta K = 1$ for any $D_1$ and $D_2$ differing in one element. Sensitivity is defined as

$$\Delta K = \max_{D_1, D_2} ||K(D_1) - K(D_2)|| \tag{10.2}$$

for any $D_1$ and $D_2$ which differ in at most one element. Note that this query has the same sensitivity not only for our specific data but for any data in this

format. The Laplacian noise, which is added to the answer, is related to the sensitivity of the query.

A $mean$ query, such as $K$ ="*What is the average age of people in the social network?*", has an even lower sensitivity for large data sets because removing any profile from the social network would change the output of the query by at most $\Delta K = max(age)/n$. There are also queries, such as $median$ queries, which have high sensitivity and require different techniques for generating noise.

A similar and somewhat weaker definition of differential privacy is the one of $(\epsilon, \delta)$-differential privacy which was developed to deal with very unlikely outputs of $K$ [19].

DEFINITION 10.14 **($\epsilon$,$\delta$)-differential privacy**. *A randomized function $K$ satisfies $\epsilon$-differential privacy if for all data sets $D_1$ and $D_2$ differing in at most one element, and any subset $S$ of possible outcomes in $Range(K)$,*

$$P(K(D_1) \in S) \leq exp(\epsilon) \times P(K(D_2) \in S) + \delta. \tag{10.3}$$

Generally, $\epsilon$ and $\delta$ are considered to be very small numbers and are picked according to different considerations, such as the size of the database.

## 4. Privacy-preserving mechanisms

So far, we have discussed existing notions of privacy preservation related to the user profiles, mostly ignoring the structural properties of the social network. Next, we discuss how privacy preservation can be achieved considering the network structure: the links between users $E_v$, and affiliation links $E_h$ to affiliation groups of users $H$. First, we present existing privacy mechanisms for social networks in Section 4.1. Section 4.2 includes overview of the mechanisms for affiliation networks. Finally, we describe research which considers both types of networks in Section 4.3. Except for the privacy mechanisms based on differential privacy, each mechanism was developed to counterattack a specific adversarial attack and background knowledge which we also present.

### 4.1 Privacy mechanisms for social networks

The majority of research in this area considers anonymization which strips off all the personal attributes from user profiles but keeps some of the structure coming from the social links between users [4, 26, 27, 35, 51, 53]. We describe this research in Section 4.1.1. In Section 4.1.2, we mention approaches to anonymizing data which consider that there is utility in keeping both attribute and structural information [8, 49].

**4.1.1 Anonymizing network structure.** One naive way of anonymizing a social network is by removing all the attributes of the profiles, and leaving

only the social link structure. This creates an anonymized graph which is isomorphic to the original graph. The intuition behind this approach is that if there are no identifying profile attributes, then attribute and identity disclosures cannot occur, and thus the privacy of users is preserved. Contrary to the intuition, this not only removes a lot of important information but it also does not guarantee the privacy of users. Two types of attacks have been proposed which show that identity and social link disclosures would occur when it is possible to identify a subgraph in the released graph in which all the node identities are known [4]. The *active attack* assumes that an adversary can insert accounts in the network before the data release, and the *passive attack* assumes that a number of friends can collude and share their linking patterns after the data release.

In the active attack an adversary creates $k$ accounts and links them randomly, then he creates a particular pattern of links to a set of $m$ other users that he is interested to monitor. The goal is to learn whether any two of the monitored nodes have links between them. When the data is released, the adversary can efficiently identify the subgraph of nodes corresponding to his $k$ accounts with provably high probability. Then he can recover the identity of the monitored $m$ nodes and the links between them which leads to social link disclosure for all $\binom{m}{2}$ pairs of nodes. With as few as $k = \Theta(\log n))$ accounts, an adversary can recover the links between as many as $m = \Theta(\log^2 n)$ nodes in an arbitrary graph of size $n$. The passive attack works in a similar manner. It assumes that the exact time point of the released data snapshot is known and that there are $k$ colluding users who have a record of what their links were at that time point.

Another type of structural background information that has been explored is similar in spirit to the linking attack mentioned in Section 2. The existence of an auxiliary social network in which the identity of users is known can help an adversary identify nodes in a target social network [41]. Starting from a set of users which form a clique both in the target and the auxiliary networks, an adversary expands the matching by finding the most likely nodes that correspond to each other in the two networks by using structural information, such as number of user friends (node degree), and number of common neighbors. To validate this attack, it has been shown that the discovered matches sometimes correspond to matches found using descriptive user attributes such as username and location in the social networks of Twitter and Flickr [41].

**Structural privacy**. Starting from the idea that certain subgraphs in the social network are unique, researchers have studied the mechanism of protecting individuals from identity disclosure when an adversary has *background information about the graph structure* around a node of interest [26, 27, 35, 48, 51, 53]. Each node has structural properties (subgraph signature) that are the same as the ones of a small set of other nodes in the graph, called a candidate set for this node [27]. Knowing the true structural properties of a node, an adversary

may be able to discover the identity of that node in the anonymized network. Structure queries can be posed to the network to discover nodes with specific subgraph signatures.

Looking at the immediate one-hop neighbors, each node has a star-shaped subgraph in which the size of the subgraph is equal to the degree of the node plus one. With the assumption that identity disclosure can occur based on a node's degree, the degree becomes an identifying attribute that a data provider would like to hide. In our toy network (Figure 10.1), Ana and Don would be in each other's candidate sets because they both have degree 2; Emma, Gina and Fabio appear in the same candidate set for either of the three nodes; Bob and Chris are uniquely identifiable because they are the only ones in the network with degrees four and one, respectively. The notion of *k-degree anonymity* [35] was formulated to protect individuals from an adversary who has background information of user's node degrees. It states that each node should have the same degree as at least $k-1$ other nodes in the anonymized network.

Adding the links between the one-hop neighbors of a node, sometimes referred to as the 1.5-hop neighborhood, creates a richer structural signature. Based on this, Ana and Don still have the same subgraph signature, and so do Emma and Fabio. However, Gina has a unique signature and is easily identifiable by an adversary who has knowledge of her true 1.5-hop neighborhood structure. Zhou and Pei [51] formalize the desired property to protect individuals from this type of attack. A graph satisfies *k-neighborhood anonymity* if every node in the network has a 1.5-hop neighborhood graph isomorphic to the 1.5-hop neighborhood graph of at least $k-1$ other nodes. The name of this property was given by Wu et al. [47].

In our example, Ana and Don become uniquely identifiable once we look at their 2-hop neighborhoods. Emma and Fabio have isomorphic signatures regardless of the size of the neighborhood for which the adversary has background information. This leads to the most general privacy preservation definitions of *k-candidate anonymity* [27] and k-automorphism anonymity [53].

DEFINITION 10.15 **$K$-Candidate anonymity**. *An anonymized graph satisfies $K$-Candidate Anonymity with respect to a structural query $Q$ if there is a set of at least $K$ nodes which match $Q$, and the likelihood of every candidate for a node in this set with respect to $Q$ is less than or equal to $1/k$.*

$K$-Candidate anonymity [27], considers the structural anonymity of users given a particular structural query, i.e., a subgraph signature. Hay et al. define three types of structural queries, vertex refinement queries, subgraph queries and hub fingerprint queries [27, 26]. Zou et al. [53] assume a much more powerful adversary who has knowledge of any subgraph signature of a target individual. They propose the notion of *k-automorphism anonymity* to fend off such an adversary.

DEFINITION 10.16 **$k$-automorphism anonymity**. *An anonymized graph is $k$-automorphic if every node in the graph has the same subgraph signature (of arbitrary size) as at least $k-1$ other graph nodes, and the likelihood of every candidate for that node is less than or equal to $1/k$.*

**Anonymization**. The anonymization strategies for social network structure fall into four main categories:

- **Edge modification**. Since complete removal of the links to keep structural properties private would yield a disconnected graph, edge modification techniques propose edge addition and deletion to meet desired constraints. Liu and Terzi anonymize the network degree sequence to meet $k$-degree anonymity [35]. This is easy to achieve for low-degree nodes because the degree distribution in social networks often follows a power law. For each distinguishable higher-degree node, where distinguishable is defined as a degree for which there are less than $k$ nodes with that degree, the anonymization algorithm increases its degree artificially so that it becomes indistinguishable from at least $k-1$ other nodes. The objective function of the algorithm is to minimize the number of edge additions and deletions. We discuss another edge modification algorithm [51] with a similar objective but a stronger privacy guarantee in Section 4.1.2. Zou et al. [53] propose an edge modification algorithm that achieves $k$-automorphism anonymity.

- **Randomization**. Anonymization by randomization can be seen as a special case of anonymization by edge modification. It refers to a mechanism which alters the graph structure by removing and adding edges at random, and preserves the total number of edges. Hay et al. [27] show that if this is performed uniformly at random, then it fails to keep important graph metrics of real-world networks. Ying and Wu [48] propose *spectrum-preserving randomization* to address this loss of utility. The graph's spectral properties are the set of eigenvalues of the graph's adjacency matrix to which important graph properties are related. Preserving this spectrum guides the choice of random edges to be added and deleted. However, the impact of this approach on privacy is unclear.

  Two recent studies have presented algorithms for reconstructing randomized networks [44, 46]. Wu et al. [46] take a low rank approximation approach and apply it to a randomized network structure, such that accurate topological features can be recovered. They show that in practice reconstruction may not pose a larger threat to privacy than randomization because the original network is more similar to the randomized network than to the reconstructed network. Vuokko and Terzi [44] consider reconstruction mechanisms for networks where randomization has been

applied both to the structure and attributes of the nodes. They identify cases in which reconstruction can be achieved in polynomial time. The effect of both reconstruction strategies on privacy has not been assessed.

- **Network generalization**. One way to alleviate an attack based on structural background information is by publishing the aggregate information about the structural properties of the nodes [26]. In particular, one can partition the nodes and keep the density information inside and between parts of the partition. Nodes in each partition have the same structural properties, so that an adversary coming with a background knowledge is not able to distinguish between these nodes. In practice, sampling from the anonymized network model creates networks which keep many of the structural properties of the original network, such as degree distribution, path length distribution and transitivity. Network generalization strategies for other network types are discussed in Section 4.1.2 [8, 49] and Section 4.3 [6].

- **Differentially private mechanisms**. Differentially private mechanisms refer to algorithms which guarantee that individuals are protected under the definition of differential privacy (see Section 3.3). Hay et al. [24] propose an efficient algorithm which allows the public release of one of the most commonly studied network properties, degree distribution, while guaranteeing differential privacy. The algorithm involves a post-processing step on the differentially private output, which ensures a more accurate result. The empirical analysis on real-world and synthetic networks shows that the resulting degree-distribution estimate exhibits low bias and variance, and can be used for accurate analysis of power-law distributions, commonly occurring in networks.

**4.1.2 Anonymizing user attributes and network structure.** So far, we have discussed anonymization techniques which perturb the structure of the network but do not consider attributes of the nodes, such as gender, age, nationality, etc. However, providing the (perturbed) structure of social networks is often not sufficient for the purposes of the researchers who study them. In another line of privacy research, the assumption is that anonymized data will have utility only if it contains both structural properties and node attributes.

**Anonymization**. Zhou and Pei [51] assume that each node has one attribute which they call a label. They show that achieving $k$-neighborhood anonymity is $NP$-hard and propose a greedy *edge modification* and *label generalization* algorithm. The algorithm extracts the 1.5-neighborhood signatures for all nodes in the graph and represents them concisely using *DFS trees*. Then it clusters the signatures and anonymizes the ones in each cluster to achieve $k$-neighborhood anonymity. The objective function of the algorithm is simi-

lar to the one of Liu and Terzi [35], the minimization of the number of edge additions.

Zheleva and Getoor [49] study the problem of social link disclosure in graphs with multiplex relations. The assumption is that an adversary has an accurate statistical model for predicting sensitive relationships if given the attributes of nodes and edges in the original data, therefore attributes have to be perturbed in the released data. They propose anonymization by generalization of the data as a two-step process. In the first step, nodes are treated as a table of records, and their attributes are anonymized to guarantee the privacy of users, for example, to meet one of the privacy definitions described earlier. Using $k$-anonymity, this creates a partition of the network nodes into equivalence classes. In the second step, the structure of the network is partially preserved by keeping aggregate structural information inside and between the equivalence classes.

Campan and Truta [8] also take a network generalization approach to the process of anonymizing a social network. Their greedy algorithm optimizes a utility function using the attribute and structural information simultaneously rather than as a two-step process. They introduce a structural information loss measure, and adopt an existing measure of attribute information loss. The anonymization algorithm can be adjusted to preserve more of the structural information of the network or the nodes' attribute values.

## 4.2 Privacy mechanisms for affiliation networks

Next, we concentrate on affiliation networks and discuss privacy-preserving techniques developed specifically for this type of network. The affiliation network is represented as a bipartite graph with two types of nodes $V$ and $H$, and the affiliation links between them $E_h$. Figure 10.6 shows an illustration of this graph where on the left-hand side there are users, and on the right-hand side there are movies that the users rated. The affiliation links have a weight corresponding to the movie ratings for each user, on a scale from $1$ to $5$.

Netflix, an online movie rental company, set up a competition aimed at improving their movie recommendation systems. They released a dataset with around $100$ million dated ratings from $480$ thousand randomly-chosen Netflix customers. To protect customer privacy, each customer id has been replaced with a randomly-assigned id. However, this naive anonymization was found to be vulnerable under a linking attack [40]. Using the dates of user ratings and matching the records released by $Netflix$ and user profiles in $IMDB$, an online movie database, Narayanan and Shmatikov [40] were able to achieve identity and sensitive attribute disclosure for some of the users in the Netflix dataset.

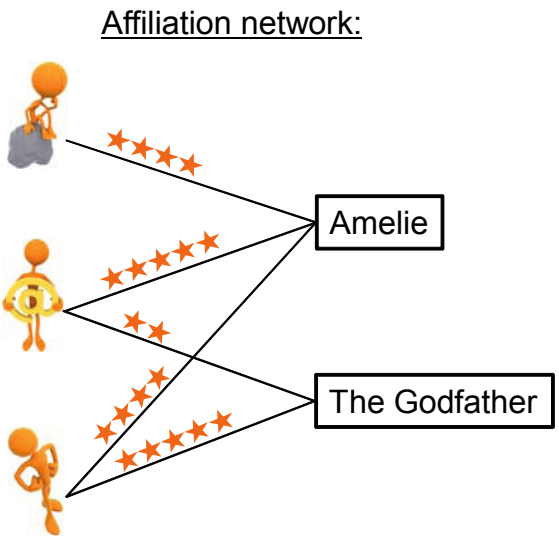


*Figure 10.6.* An affiliation network as a bipartite graph between three users and two movies. The affiliation links show the ratings that users gave to the movies on a scale from 1 to 5.

A related problem is the problem of releasing a search query graph in which user information is contained in the affiliation links between search engine queries and clicked website URLs [30]. In particular, there is a bipartite graph of (query,URL) pairs.Here, the links have a weight corresponding to the number of users who posed a particular query and clicked on the particular URL. In addition, there are links between queries with a weight equal to the number of users who posed the first query and then reformulated it into the second query. Each query also has counts of the number of times the query was posed to the search engine. The utility in such data is in using it for learning better search ranking algorithms. Figure 10.7(a) shows an example a user-query graph. Figure 10.7(b) shows its reformulation into a search query graph where individual users are not represented explicitly but only as aggregate numbers.

**4.2.1 Anonymization.** Two types of privacy mechanisms for affiliation networks have been studied in the research literature:

- **Network generalization**. Cormode et al. [13] propose a privacy definition for affiliation networks, *(k,l)-grouping*, tailored to prevent sensitive affiliation link disclosure. The authors make the assumption that affiliation links can be predicted based on node attributes and the structure of the network. They show why existing table anonymization techniques fail to preserve the structural properties of the network, and propose a greedy anonymization algorithm which keeps the structure intact but

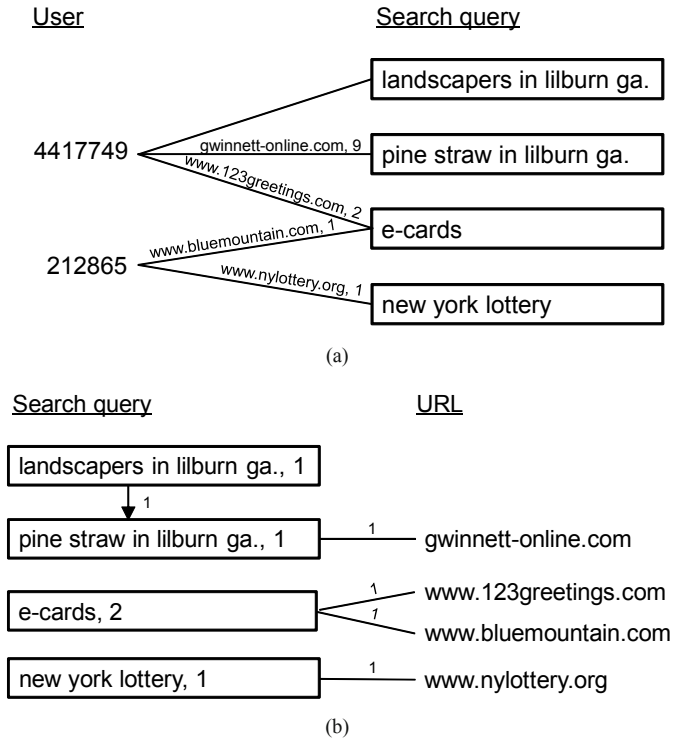


*Figure 10.7.* a) *User-query graph* representing the users, their queries, the websites they clicked on and the ranking of each website, and b) its reformulation into a *search query graph*.

generalizes node attributes. The algorithm requires that each node is indistinguishable from at least $k-1$ other nodes in terms of node properties, and each affiliation group is indistinguishable from at least $l-1$ other affiliation groups, the basis of $(k,l)$-grouping. The utility is in being able to answer accurately aggregate queries about users and affiliation groups.

- **Differentially private mechanisms**. A private mechanism for a recommender system has been developed specifically for the movie recommendation setting [39]. The system works by providing differentially private mechanisms for computing counts, rating averages per movie and per user, and the movie-movie covariances in the data. These statistics are sufficient for computing distances based on k-nearest neighbor for predicting the ratings associated with new affiliation links. Using the statistics released by the mechanism, the algorithm performs with an accuracy comparable to the one in the original data.

  Korolova et al. [30] have proposed an ($\epsilon$,$\delta$)-differentially private algorithm which allows the publication of a search query graph for this purpose. Here the search logs are the database, and pairs of databases $D_1$ and $D_2$ are considered to differ in one element when one database excludes the search logs of exactly one user. The algorithm keeps only a limited number of queries and clicks for each user and allows for two types of functions on the graph which are sufficient for evaluating ranking algorithms. The first function gives a search query and its noisy count if it exceeds a pre-specified threshold. The second function publishes the noisy weight of the (query,URL) link for the top URLs for each query which was safe to publish according to the first function.

## 4.3 Privacy mechanisms for social and affiliation networks

There has not been much research on the privacy implications of the interplay between social and affiliation networks. It is obvious that they inherit all the privacy issues discussed so far for either type of network. What is not so obvious is that the complex dependencies these networks create can allow an adversary to learn private information in intricate ways. In particular, one can use the social environment of users to learn private information about them. One type of attack, which we call an *attribute inference attack*, assumes that an attribute is sensitive only for a subset of the users in the network and that the other users in the network are willing to publish it publicly [50]. The analogy in real-world social networks is the existence of private and public profiles. The attack works by creating a statistical model for predicting the sensitive attribute using the publicly available information and applying that model to

predict the users with private profiles. In its basic form, the attack assumes that besides the network structure, the only user attributes that are available are the sensitive attribute value for the public profiles. Naturally, using other profile attributes can create even more powerful statistical models, as Lindamood et al. show [33]. An adversary succeeds when he can recover the sensitive attribute values for a subset of the nodes with high probability.

By taking into account all social and affiliation links, often declared publicly in online social networks, the model can use link-based classification techniques. Link-based classification breaks the assumption that data comprises of independent and identically distributed (iid) nodes and it can take advantage of autocorrelation, the property that attributes of linked objects often correlated with each other. For example, political affiliations of friends tend to be similar, students tend to be friends with other students, etc. A comprehensive survey of models for link-based classification can be found in the work by Sen et al. [42]. The results of Zheleva and Getoor [50] suggest that link-based classification can predict sensitive attributes with high accuracy using information about online social groups, and that social groups have a higher potential for leaking personal information than friendship links.

**4.3.1 Anonymization.** Bhagat et al. [6] consider attacks for sensitive social link disclosure in social and affiliation networks, to which they refer as *rich interaction graphs*. Two nodes participating in the same group is also considered as a sensitive social link between the two users. Bhagat et al. represent the social and affiliation networks as a bipartite graph, in which one type of nodes are the users and the other type of nodes are affiliation groups. Social links are represented as affiliation groups of size two.

They propose two types of network generalization techniques to prevent social link disclosure. The first technique, a *uniform list approach*, keeps the structure intact, in a manner similar to $(k, l)$-groupings [13]. It divides nodes into classes of size $m$ ensuring that each node's interactions fall on nodes of different classes. Each class is split into label lists of size $k$, thus ensuring that the probability of a link between two users (through a social link or a common affiliation group) is at most $1/k$. If the adversary has a background knowledge of the identities of $r$ of the nodes and $k$ is equal to $m$, then this probability becomes $1/(k-r)$. The second technique, a *partitioning approach*, also divides the nodes into classes of size $m$ so that each node's interactions fall on nodes of different classes. However, it does not keep the original network structure, and publishes only the number of edges between partitions. The probability of a link between two users is guaranteed to be at most $1/m$ with or without background knowledge. The utility of the anonymized graph is in allowing accurate structural and attribute analysis of the graph.

## 5. Related literature

Research on privacy in online social networks is a very young field which discovers and addresses some of the challenges of preserving the privacy of individuals in an interconnected world [4, 6, 8, 26, 27, 31, 30, 33, 35, 41, 48, 50, 49, 51, 53]. However, privacy research has a longer history in the data mining, database and security communities. For example, privacy-preserving data mining aims at creating data mining algorithms and systems which take into consideration the sensitive content of the data [28, 2]. Chen et al. [10] provide a comprehensive, recent survey of the field of privacy-preserving data publishing. The database and security communities have studied interactive and non-interactive mechanisms for sharing potentially sensitive data [17]. Most of this research assumes that there are one or more data owners who would like to provide data access to third parties while meeting privacy constraints. In contrast, access to data in online social networks is often freely available, and users can specify their personal privacy preferences. Addressing the new privacy challenges in this area is an active area of research [29]. The unexpected threats of freely publishing personal data online is exemplified by a number of researchers [1, 33, 41, 50]. boyd points out many privacy concerns and ethical issues, related to the analysis of large online social network data [15]. Measuring the privacy of social network users and enabling them to personalize their online privacy preferences has also been the focus of recent work [36, 21]. Privacy in dynamic social networks has also received recent attention [7, 53].

## 6. Conclusion

Here, we presented the possible privacy breaches in online social networks, together with existing privacy definitions and mechanisms for preserving user privacy. While initial steps have been taken in understanding and overcoming some of the challenges of preserving privacy online, many open problems remain. In particular, some exciting new directions include studying the effect of different types of privacy disclosures on each other, privacy-preserving techniques that prevent sensitive attribute disclosure in networks, a comparison between existing anonymization techniques in terms of utility, and privacy-preserving techniques that meet the individual privacy expectations of online social network users rather than privacy definitions imposed by a data publisher or an online service provider.

## Acknowledgments

The authors would like to thank Liliana Mihalkova, Daozheng Chen and the anonymous reviewers for the useful feedback, and Arkady Yerukhimovich for a fruitful discussion on differential privacy. Some of the images included

in Figure 10.3 and Figure 10.4 were taken from Wikimedia Commons and FreeDigitalPhotos.net. The cartoon characters in Figure 10.1 and Figure 10.6 are from www.lumaxart.com. This book chapter was supported in part by NSF Grant #0746930.

## References

[1] A. Acquisti and R. Gross. Predicting social security numbers from public data. In *PNAS*, 2009.

[2] C. C. Aggarwal and P. S. Yu. *Privacy-Preserving Data Mining: Models and Algorithms*. Springer, 2008.

[3] G. Aggarwal, T. Feder, K. Kenthapadi, R. Motwani, R. Panigrahy, D. Thomas, and A. Zhu. Approximation algorithms for k-anonimity. *JPT*, Nov. 2005.

[4] L. Backstrom, C. Dwork, and J. Kleinberg. Wherefore art thou r3579x: anonymized social networks, hidden patterns, and struct. steganography. In *WWW*, 2007.

[5] M. Barbaro and T. Zeller. A face is exposed for aol searcher no. 4417749. *New York Times*, August 2006.

[6] S. Bhagat, G. Cormode, B. Krishnamurthy, and D. Srivastava. Class-based graph anonymization for social network data. In *VLDB*, 2009.

[7] S. Bhagat, G. Cormode, B. Krishnamurthy, and D. Srivastava. Privacy in dynamic social networks. In *WWW Poster*, 2010.

[8] A. Campan and T. M. Truta. A clustering approach for data and structural anonymity in social networks. *PinKDD*, 2008.

[9] S. Chawla, C. Dwork, F. Mcsherry, A. Smith, and H. Wee. Toward privacy in public databases. In *TCC*, 2005.

[10] B.-C. Chen, D. Kifer, K. LeFevre, and A. Machanavajjhala. Privacy-preserving data publishing. *Foundations and trends in databases*, 2(1–2):1–167, 2009.

[11] D. R. Choffnes, J. Duch, D. Malmgren, R. Guimera, F. E. Bustamante, and L. Amaral. Swarmscreen: Privacy through plausible deniability in p2p systems tech. Technical Report NWU-EECS-09-04, Department of EECS, Northwestern University, June 2009.

[12] K. Clarkson, K. Liu, and E. Terzi. Towards identity anonymization in social networks. In P.Yu, J. Han, and C. Faloutsos, editors, *Link Mining: Models Algorithms and Applications*. Springer, 2010, in press.

[13] G. Cormode, D. Srivastava, T. Yu, and Q. Zhang. Anonymizing bipartite graph data using safe groupings. In *VLDB*, 2008.

[14] C. Cortes, D. Pregibon, and C. Volinsky. Communities of interest. In *IDA*, 2001.

[15] danah boyd. Privacy and publicity in the context of big data. In *WWW Invited Talk*, 2010. Available at `http://www.danah.org/papers/talks/2010/WWW2010.html`.

[16] S. Das, Ĕmer Egecioglu, and A. E. Abbadi. Anonymizing weighted social network graphs. In *ICDE*, 2010.

[17] C. Dwork. Differential privacy. In *ICALP*, 2006.

[18] C. Dwork. An ad omnia approach to defining and achieving private data analysis. *PinKDD*, 4890:1–13, 2007.

[19] C. Dwork, K. Kenthapadi, F. McSherry, I. Mironov, and M. Naor. Our data, ourselves: privacy via distributed noise generation. In *EUROCRYPT*, 2006.

[20] C. Dwork, F. McSherry, K. Nissim, and A. Smith. Calibrating noise to sensitivity in private data analysis. In *TCC*, 2005.

[21] L. Fang and K. LeFevre. Privacy wizards for social networking sites. In *WWW*, 2010.

[22] L. Getoor and C. P. Diehl. Link mining: a survey. *SIGKDD Explorations*, 7(2):3–12, December 2005.

[23] L. Getoor and B. Taskar, editors. *Introduction to statistical relational learning*. MIT Press, 2007.

[24] M. Hay, C. Li, G. Miklau, and D. Jensen. Accurate estimation of the degree distribution of private networks. In *ICDM*, 2009.

[25] M. Hay, G. Miklau, and D. Jensen. Enabling accurate analysis of private network data. In F. Bonchi and E. Ferrari, editors, *Privacy-Aware Knowledge Discovery: Novel Applications and New Techniques*. Chapman & Hall/CRC Press, 2010, in press.

[26] M. Hay, G. Miklau, D. Jensen, and D. Towsley. Resisting structural identification in anonymized social networks. In *VLDB*, August 2008.

[27] M. Hay, G. Miklau, D. Jensen, P. Weis, and S. Srivastava. Anonymizing social networks. Technical report, University of Massachusetts, Amherst, March 2007.

[28] Y. Z. J. Vaidya, C. Clifton. *Privacy Preserving Data Mining*. Springer, 2006.

[29] J. M. Kleinberg. Challenges in mining social network data: processes, privacy, and paradoxes. In *KDD*, pages 4–5, 2007.

[30] A. Korolova, K. Kenthapadi, N. Mishra, and A. Ntoulas. Releasing search queries and clicks privately. In *WWW*, 2009.

[31] A. Korolova, R. Motwani, S. U. Nabar, and Y. Xu. Link privacy in social networks. In *CIKM*, 2008.

[32] N. Li, T. Li, and S. Venkatasubramanian. t-closeness: Privacy beyond k-anonymity and l-diversity. In *ICDE*, 2007.

[33] J. Lindamood, R. Heatherly, M. Kantarcioglu, and B. Thuraisingham. Inferring private information using social network data. In *WWW Poster*, 2009.

[34] K. Liu, K. Das, T. Grandison, and H. Kargupta. Privacy-preserving data analysis on graphs and social networks. In H. Kargupta, J. Han, P. Yu, R. Motwani, and V. Kumar, editors, *Next Generation of Data Mining*, chapter 21, pages 419–437. Chapman & Hall/CRC, 2008.

[35] K. Liu and E. Terzi. Towards identity anonymization on graphs. In *SIGMOD*, 2008.

[36] K. Liu and E. Terzi. A framework for computing the privacy scores of users in online social networks. In *ICDM*, 2009.

[37] L. Liu, J. Wang, J. Liu, and J. Zhang. Privacy preservation in social networks with sensitive edge weights. In *SDM*, 2009.

[38] A. Machanavajjhala, J. Gehrke, D. Kifer, and M. Venkitasubramaniam. l-diversity: Privacy beyond k-anonymity. In *ICDE*, 2006.

[39] F. McSherry and I. Mironov. Differentially private recommender systems: building privacy into the netflix prize contenders. In *KDD*, 2009.

[40] A. Narayanan and V. Shmatikov. Robust de-anonymization of large sparse datasets. *Security and Privacy*, 2008.

[41] A. Narayanan and V. Shmatikov. De-anonymizing social networks. In *Security and Privacy*, 2009.

[42] P. Sen, G. M. Namata, M. Bilgic, L. Getoor, B. Gallagher, and T. Eliassi-Rad. Collective classification in network data. *AI Magazine*, 29(3):93–106, 2008.

[43] L. Sweeney. Achieving k-anonymity privacy protection using generalization and suppression. *International Journal of Uncertainty*, 10(5):571–588, 2002.

[44] N. Vuokko and E. Terzi. Reconstructing randomized social networks. In *SDM*, 2010.

[45] G. Wondracek, T. Holz, E. Kirda, and C. Kruegel. A practical attack to de-anonymize social network users. In *Security and Privacy*, 2010.

[46] L. Wu, X. Ying, and X. Wu. Reconstruction of randomized graph via low rank approximation. In *SDM*, 2010.

[47] X. Wu, X. Ying, K. Liu, and L. Chen. A survey of algorithms for privacy-preserving social network analysis. In C. Aggarwal and H. Wang, editors, *Managing and Mining Graph Data*. Kluwer Academic Publishers, 2009.

[48] X. Ying and X. Wu. Randomizing social networks: a spectrum preserving approach. In *SDM*, 2008.

[49] E. Zheleva and L. Getoor. Preserving the privacy of sensitive relationships in graph data. *PinKDD*, 2007.

[50] E. Zheleva and L. Getoor. To join or not to join: the illusion of privacy in social networks with mixed public and private user profiles. In *WWW*, 2009.

[51] B. Zhou and J. Pei. Preserving privacy in social networks against neighborhood attacks. In *ICDE*, 2008.

[52] B. Zhou, J. Pei, and W.-S. Luk. A brief survey on anonymization techniques for privacy preserving publishing of social network data. *SIGKDD Explorations*, 10(2), 2009.

[53] L. Zou, L. Chen, and M. T. Ozsu. K-automorphism: A general framework for privacy preserving network publication. In *VLDB*, 2008.