A Brief Survey on Anonymization Techniques for Privacy Preserving Publishing of Social Network Data*

Bin Zhou
School of Computing Science
Simon Fraser University,
Canada
bzhou@cs.sfu.ca

Jian Pei
School of Computing Science
Simon Fraser University,
Canada
jpei@cs.sfu.ca

Wo-Shun Luk
School of Computing Science
Simon Fraser University,
Canada
woshun@cs.sfu.ca

ABSTRACT

Nowadays, partly driven by many Web 2.0 applications, more and more social network data has been made publicly available and analyzed in one way or another. Privacy preserving publishing of social network data becomes a more and more important concern. In this paper, we present a brief yet systematic review of the existing anonymization techniques for privacy preserving publishing of social network data. We identify the new challenges in privacy preserving publishing of social network data comparing to the extensively studied relational case, and examine the possible problem formulation in three important dimensions: privacy, background knowledge, and data utility. We survey the existing anonymization methods for privacy preservation in two categories: clustering-based approaches and graph modification approaches.

1. INTRODUCTION

Recently, social networks [24; 31] have received dramatic interest in research and development, partly due to more and more social networks are built online and the fast development of Web 2.0 applications. Social networks model social relationships by graph structures using vertices and edges. Vertices model individual social actors in a network, while edges model relationships between social actors. Many different kinds of social networks present in our lives such as friendship networks, telephone call networks, and academia co-authorship networks.

Due to the rapidly increasing popularity of social networking sites on the Web, more and more people participate in social networks. According to a poll by TNS Canadian Facts[1], a Canadian marketing and social research firm, teens and young adults are the heaviest users of social networking sites. Specifically, 83% of 13-17 years old people and 74% of 18-29 years old people visited at least one social networking site. 60% of people in their 30s and 45% of those in their 40s visited at least one social networking site.

Social networks connect social actors. The connections are often beneficial to entrepreneurs and commercial companies. For example, they can use the connections to expand their customer bases. In many cases, those social networks can serve as a customer relationship management tool for companies selling products and services. Companies can also use social networks to identify potential customers or recruit candidate employees. For example, according to the statistics published in Time Magazine[2], 12% of employers in the United States use popular social networking sites such as MySpace and Facebook to investigate potential employees.

With the rapid growth of social networks, social network analysis [8; 24; 29; 25] has emerged as a key technique in modern sociology, geography, economics, and information science. The goal of social network analysis is to uncover hidden social patterns. The power of social network analysis has been shown much stronger than that of traditional methods which focus on analyzing the attributes of individual social actors. In social network analysis, the relationships and ties between social actors in a network are often regarded more important and informative than the attributes of individual social actors. Social network analysis approaches have been shown very useful in capturing and explaining many real-world phenomena such as the well-known "small world phenomenon" [30].

1.1 Privacy Attacks Using Published Social Network Data

As more and more rich social media, popular online social networking sites, and various kinds of social network analyzing and mining techniques are available, privacy in social networks becomes a serious concern [14; 25; 3], particularly when social network data is published.

An adversary may intrude privacy of some victims using the published social network data and some background knowledge. Importantly, many of the richest emerging sources of social network data come from settings such as e-mails, instant messages or telephone communication. Users have strong expectations of privacy on such data. When social network data is made public in one way or another, it is far from sufficient to protect privacy by simply replacing the

*The research was supported in part by an NSERC Discovery grant and an NSERC Discovery Accelerator Supplements grant. All opinions, findings, conclusions and recommendations in this paper are those of the authors and do not necessarily reflect the views of the funding agencies.
We sincerely thank the anonymous reviewers for their constructive and informative advice which helps to improve the quality of this paper. Particularly, we thank them for their suggestions on some related work and the description of some critical properties of some methods.

[1] http://www.privcom.gc.ca/information/social/index_e.asp

[2] August 20, 2007, which is available at http://www.time.com/time/magazine/article/0,9171,1651513,00.html.

identifying attributes such as `name` and `SSN` of individuals by meaningless unique identifiers.

EXAMPLE 1 (BACKGROUND KNOWLEDGE-BASED ATTACKS). *Backstrom et al. [3] identified a family of attacks such that even from a single anonymized copy of a social network hiding the identifying attributes, it is possible for an adversary to learn whether edges exist or not between some specific target pairs of vertices. The attacks are based on the uniqueness of some small random subgraphs embedded in an arbitrary network, using ideas related to those found in arguments from Ramsey theory [2].*

Two categories of attacks are addressed in [3]. The first category is active attacks. Before releasing the anonymized network G of $(n-k)$ vertices, the attackers can choose a set of b target users, randomly create a subgraph H containing k vertices, and then attach H to the target vertices. After the anonymized network is released, if the attackers can find the subgraph H in the released graph G, then the attackers can follow edges from H to locate the b target vertices and their locations in G, and determine all edges among those b vertices.

To implement the attacks, the random graph H should satisfy the following requirements:

1. *H must be uniquely and efficiently identifiable regardless of G;*
2. *there is no other subgraph S in G such that S and H are isomorphic; and*
3. *H has no automorphism.*

Backstrom et al. [3] provide two methods to construct subgraphs satisfying the above requirements.

The second category is passive attacks, which are based on the fact that most vertices in social networks usually belong to a small uniquely identifiable subgraph. Thus, an attacker can collude with other $(k-1)$ friends to identify additional vertices connected to the distinct subset of the coalition. The attacks are possible under the following assumptions:

1. *all colluders should know edges among themselves, that is, the internal structure of H;*
2. *all colluders should know the name of their neighbors outside the coalition; and*
3. *there does not exist a Hamiltonian path linking $x_1, x_2, \ldots, x_n$, where x_i is a vertex in G.*

The experiments on a real social network with 4.4 million vertices and 77 million edges show that the creation of 7 vertices by an attacker can reveal on average 70 target vertices and compromise the privacy of approximately 2,400 edges between them. ∎

Example 1 indicates that privacy issues in social networks are real. Several initiative methods on privacy preservation in social network data publishing have been proposed, which will be surveyed in the rest of the paper. Generally, privacy preservation methods against background knowledge-based attacks often adopt data anonymization approaches, which are the focus of this survey.

1.2 Challenges in Anonymizing Social Network Data

Privacy preservation on relational data has been studied extensively. A major category of privacy attacks on relational data is to re-identify individuals by joining a published table containing sensitive information with some external tables modeling background knowledge of attackers.

To battle the re-identification attacks, the mechanism of k-anonymity was proposed [23; 26]. Specifically, a data set is said to be k-anonymous ($k \geq 1$) if, on the quasi-identifier attributes (that is, the maximal set of join attributes to re-identify individual records), each record is indistinguishable from at least $(k-1)$ other records. The larger the value of k, the better the privacy is protected.

Although k-anonymity has been well adopted, Machanavajjhala *et al.* [20] showed that a k-anonymous table may still have some subtle but severe privacy problems due to the lack of diversity in the sensitive attributes. In particular, they showed that, the degree of privacy protection does not really depend on the size of the equivalence classes on quasi-identifier attributes which contain tuples that are identical on those attributes. Instead, it is determined by the number and distribution of distinct sensitive values associated with each equivalence class. To overcome the weakness in k-anonymity, they propose the notion of l-diversity [20]. Xiao and Tao [33] prove that l-diversity always guarantees stronger privacy preservation than k-anonymity.

Though several important models and many efficient algorithms have been proposed to preserve privacy in relational data, most of the existing studies can deal with relational data only. Those methods cannot be applied to social network data straightforwardly. Anonymizing social network data is much more challenging than anonymizing relational data [39].

First, it is much more challenging to model background knowledge of adversaries and attacks about social network data than that about relational data. On relational data, it is often assumed that a set of attributes serving as a quasi-identifier is used to associate data from multiple tables, and attacks mainly come from identifying individuals from the quasi-identifier. However, in a social network, many pieces of information can be used to identify individuals, such as labels of vertices and edges, neighborhood graphs, induced subgraphs, and their combinations. It is much more complicated and much more difficult than the relational case.

Second, it is much more challenging to measure the information loss in anonymizing social network data than that in anonymizing relational data. Typically, the information loss in an anonymized table can be measured using the sum of information loss in individual tuples. Given one tuple in the original table and the corresponding anonymized tuple in the released table, we can calculate the distance between the two tuples to measure the information loss at the tuple level. However, a social network consists of a set of vertices and a set of edges. It is hard to compare two social networks by comparing the vertices and edges individually. Two social networks having the same number of vertices and the same number of edges may have very different network-wise properties such as connectivity, betweenness, and diameter. Thus, there can be many different ways to assess information loss and anonymization quality.

Last but not least, it is much more challenging to devise

anonymization methods for social network data than for relational data. Divide-and-conquer methods are extensively applied to anonymization of relational data due to the fact that tuples in a relational table are separable in anonymization. In other words, anonymizing a group of tuples does not affect other tuples in the table. However, anonymizing a social network is much more difficult since changing labels of vertices and edges may affect the neighborhoods of other vertices, and removing or adding vertices and edges may affect other vertices and edges as well as the properties of the network.

1.3 Contributions and Paper Organization

Privacy preserving data publishing and analyzing techniques on relational data have been well developed. Recently, there have been a few studies on privacy preservation in social network data. However, the research and development of privacy preservation techniques on social network data are still in their infancy. This survey provides a timely overview of the recent studies on this direction. Particularly, we make the following two contributions.

- We analyze the privacy models in social networks. Based on the experience gained in the previous studies in privacy preservation in relational data, several important factors should be modeled, including privacy of individuals subject to attacks and background knowledge of adversaries. We categorize the information in social networks, and model privacy, attacks, and background knowledge in social network data. Moreover, we investigate the utility of social networks, which is the major optimization goal of anonymization.

- We categorize the recent anonymization techniques in privacy preserving publishing of social network data. Several anonymization methods have been proposed to prevent specific privacy attacks. We classify them into clustering-based approaches and graph modification approaches. As far as we know, this is the first work to systematically categorize recent privacy preservation techniques in social networks.

The rest of the paper is organized as follows. In Section 2, we analyze the privacy models in social networks. We categorize the existing anonymization methods in Section 3. We survey the clustering-based approaches and the graph modification approaches in Sections 4 and 5, respectively. We conclude the paper in Section 6.

2. MODELING PRIVACY PRESERVATION IN SOCIAL NETWORKS

To battle privacy attacks and develop protection techniques in social networks, we need to model three aspects. First, we need to identify the *privacy information* which may be under attack. Second, we need to model the *background knowledge* that an adversary may use to attack the privacy of target individuals. Last, we need to specify the *usage* of the published social network data so that an anonymization method can try to retain the utility of the data as much as possible while the privacy information is fully preserved. In this section, we systematically analyze the privacy models in social networks.

Generally, we model a social network as a simple graph $G = (V, E, L, \mathcal{L}_V, \mathcal{L}_E)$, where V is a set of vertices, $E \subseteq V \times V$ is a set of edges, L is a set of labels, and a labeling function $\mathcal{L}_V : V \rightarrow L$ assigns each vertex a label and a labeling function $\mathcal{L}_E : E \rightarrow L$ assigns each edge a label. For a graph G, $V(G)$, $E(G)$, $L(G)$, $\mathcal{L}_V(G)$, and $\mathcal{L}_E(G)$ are the set of vertices, the set of edges, the set of labels, the vertex labeling function in G, and the edge labeling function in G, respectively.

2.1 Privacy in Social Networks

In privacy preservation on relational data, the attributes in a table are divided into two groups: non-sensitive attributes and sensitive attributes. The values in sensitive attributes are considered to be private for individuals. However, in social network data, much more pieces of information can be considered as privacy of individuals. We list some of them below as examples.

- **Vertex existence**. In social network data, whether a target individual appears in the network or not can be considered as the privacy of the individual. For example, suppose a social network of millionaires is released where each vertex in the network represents a millionaire. If a target individual can be determined appearing in the network, an attacker knows that the target must be a millionaire. As another example, a disease infection network is valuable in public health research. However, if an adversary can determine that a target individual appears in the network, then the target's privacy of having the infection is breached.

- **Vertex properties** [3; 12; 13; 18; 36]. In social network data, some properties of a vertex such as degree can be considered as privacy of the individual. For example, if an adversary knows the degree of a target individual in a financial support network, the adversary knows how many support sources the target has. As another example, if an adversary knows the distance between a target individual to the center of a community in a social network, whether the victim is a community leader can be derived.

- **Sensitive vertex labels** [38; 4; 37]. In social network data, vertices may carry labels, which can be divided into two categories: non-sensitive vertex labels and sensitive vertex labels. Similar to the case of relational data, the values of sensitive vertex labels are considered to be privacy of individuals. For example, in a disease infection network, each individual may be associated with a sensitive label `disease`. The disease of a target individual can be identified by adversaries once the target can be uniquely linked to a vertex in the graph or a group of vertices having the same sensitive label in the graph.

- **Link relationship** [5]. In social network data, an edge between two vertices indicates that there is a relationship between the two corresponding individuals. The link relationship among vertices can be considered as privacy of individuals. For example, in a finance transaction network, two vertices are connected by an edge if there is a finance transaction happens between them. An adversary may detect whether two target individuals have finance transactions if whether an edge

between the individuals exists in such a network can be determined.

- **Link weight** [19]. Some social networks may be weighted. The weights of edges can reflect affinity between two vertices or record the communication cost between two individuals. For example, a social network about communication between friends may be weighted such that the weight of an edge is the communication frequency between two individuals, which may be considered privacy for some people.

- **Sensitive edge labels** [4; 37]. In social network data, edges may carry several labels as well. Similar to the above case of sensitive vertex labels, the edge labels may be divided into non-sensitive edge labels and sensitive edge labels. The values of sensitive edge labels are considered as privacy for the corresponding two individuals.

- **Graph metrics**. In social network analysis, many graph metrics have been proposed to analyze graph structures, such as betweenness (that is, the degree an individual lies between other individuals in the network), closeness (that is, the degree an individual is near to all other individuals in the network directly or indirectly), centrality (that is, the count of the number of relationships to other individuals in the network), path length (that is, the distances between pairs of vertices in the network), reachability (that is, the degree any member of a network can reach other members of the network), and so on. All of the above metrics may be considered as privacy for some individuals.

Modeling privacy is important which sets up the goal of privacy preservation in social networks. Different privacy concerns may lead to different problem definitions and accordingly different privacy preservation methods.

Liu *et al.* [17] and Zheleva and Getoor [37] proposed a categorization schema different from ours in this paper. They classified privacy in social networks into identity disclosure (that is, the identity of an individual who is associated with a vertex is revealed), link disclosure (that is, the sensitive relationship between two individuals is disclosed), and content disclosure (that is, the sensitive data associated with each vertex is compromised, for example, the email messages sent and/or received by the individuals in an email communication network). The categorization presented here is more extensive than the three categories in [17; 37].

2.2 Background Knowledge of Adversaries

In relational data, a major type of privacy attacks is to re-identify individuals by joining the published table with some external tables modeling the background knowledge of users. Specifically, the adversaries are assumed knowing the values on the quasi-identifier attributes of the target victims.

In privacy preservation in publishing social networks, due to the complex structures of graph data, the background knowledge of adversaries may be modeled in various ways.

- **Identifying attributes of vertices**. A vertex may be linked uniquely to an individual by a set of attributes, where the set of identifying attributes play a role similar to a quasi-identifier in the re-identification attacks on relational data. Vertex attributes are often modeled as labels in a social network. An adversary may know some attribute values of some victims. Such background knowledge may be abused for privacy attacks [4].

- **Vertex degrees**. The degree of a vertex in the network captures how many edges the corresponding individual is connected to others in the network. Such information is often easy to collect by adversaries. For example, the neighbor of a target individual may easily estimate the number of friends the victim has. An adversary equipped with the knowledge about the victim's degree can re-identify the target individual in the network by examining the vertex degrees in the network [12; 13; 18; 36].

- **Link relationship**. An adversary may know that there are some specific link relationships between some target individuals. For example, in a social network about friendship among people, edges may carry labels recording the channels people use to communicate with each other such as phone, email, and/or messaging. An adversary may try to use the background knowledge that a victim uses only emails to contact her friends in the network to link the victim to vertices in the network. Privacy attacks using link relationship as the background knowledge are studied in [4; 5; 37].

- **Neighborhoods**. An adversary may have the background knowledge about the neighborhood of some target individuals. For example, an adversary may know that a victim has four good friends who also know each other. Using this background knowledge, the adversary may re-identify the victim by searching vertices in the social graph whose neighborhoods contain a clique of size at least 4. Generally, we can consider the d-neighbor of a target vertex, that is, the vertices within a distance d to the target vertex in the network, where d is a positive integer [12; 13; 36; 38; 39].

- **Embedded subgraphs**. An adversary may embed some specific subgraphs into a social network before the network is released. After collecting the released network, it is possible for the adversary to re-identify the embedded subgraph if the subgraph is unique. As shown in [3], the creation of 7 vertices by an attacker can reveal an average of 70 target vertices.

- **Graph metrics**. Graphs have many metrics, such as betweenness, closeness, centrality, path length, reachability, and so on. Graph metrics can be used as background knowledge for the adversaries to breach the privacy of target individuals [12; 36].

2.3 Utility in Social Networks

An important aspect of anonymizing social network data is how the anonymized networks are expected to be used. Different applications may use anonymized data in different ways. For example, in some situations, anonymized networks may be used to analyze the global network structures. In some other situations, anonymized networks may be used to analyze the micro-structures. Clearly, different usage intentions may lead to different anonymization schemes.

Privacy	Utility	Background knowledge					
		Identifying attributes of vertices	Vertex degrees	Link relationship	Neighborhoods	Embedded subgraphs	Graph metrics
Vertex existence	General graph properties						
	Aggregate network queries						
Vertex properties	General graph properties		[12; 13; 18; 36]		[12; 13; 36]	[3]	[12]
	Aggregate network queries				[39; 38]		
Sensitive vertex labels	General graph properties	[4]		[4; 37]			
	Aggregate network queries				[38]		
Link relationship	General graph properties						
	Aggregate network queries			[5]			
Link weight	General graph properties						[19]
	Aggregate network queries						
Sensitive edge labels	General graph properties			[4; 37]			
	Aggregate network queries						
Graph metrics	General graph properties						
	Aggregate network queries						

Table 1: Summary of privacy models in social network data.

So far, two types of utility as follows have been considered.

- **General graph properties**. One of the most important applications of social network data is analyzing graph properties. For example, researchers may be interested in the distribution of vertex degrees in a network. Some other graph properties that are often used include diameter and clustering co-efficient of networks. Some of them are addressed in [12; 13; 18; 36; 19; 37; 4].

- **Aggregate network queries**. An aggregate network query [39; 38; 5] computes the aggregate on some paths or subgraphs satisfying some given conditions. As an example, suppose a user is interested in the average distance from a medical doctor vertex to a teacher vertex in a social network. For each doctor vertex, we can find the nearest neighbor vertex that is a teacher. Then, the aggregate network query returns the average of the distance between a doctor vertex to its nearest teacher neighbor. Aggregate network queries are useful in many applications, such as customer relationship management.

2.4 Summary

In summary, complex network structures introduce more dimensions and consequently more challenges in modeling privacy preservation problems in social network data. Generally, many pieces of information in social network data can be used to model privacy, background knowledge, and utility of anonymized data. Different combinations of those factors may lead to different problem settings. Accordingly, different anonymization methods should be developed. Table 1 shows the existing studies in a space of 3 dimensions: the privacy concerns, the background knowledge, and the data utility. As can be seen, the research and development of privacy preserving methods in social network data is still in its infancy. Many problems still have not been touched.

3. CATEGORIES OF ANONYMIZATION METHODS

In privacy preserving data publishing, in order to prevent privacy attacks, data should be anonymized properly before it is released. Anonymization methods should take into account the privacy models of the data and the utility of the data.

Generalization and perturbation are the two popular anonymization approaches for relational data. Although privacy preservation in social network data is a relatively new problem, several privacy preserving methods have been developed. Similar to privacy preservation methods in relational data, specific anonymization methods are developed for specific privacy models of social networks and specific utility goals of anonymized data.

We categorize the state-of-the-art anonymization methods on social network data into two categorizes as follows.

- **Clustering-based approaches**. A clustering-based method clusters vertices and edges into groups and anonymizes a subgraph into a super-vertex. In such a way, the details about individuals can be hidden properly. The methods in this category can be further divided into vertex clustering methods, edge clustering methods, vertex and edge clustering methods, and vertex-attribute mapping clustering methods.

- **Graph modification approaches**. A graph modification method anonymizes a graph by modifying (that is, inserting and/or deleting) edges and vertices in a graph. The modification can be conducted in three ways and correspondingly there are three sub-categories of the methods. The optimization approaches try to make up an optimal configuration and the modify the graph accordingly. The randomized graph modification approaches conduct perturbation. Last, the greedy graph modification approaches greedily introduce modification to meet the privacy preservation requirement and optimize the data utility objectives.

In the rest of this paper, we will focus on the clustering-based approaches in Section 4. Section 5 is dedicated to the graph modification approaches.

4. CLUSTERING-BASED APPROACHES

Generalization is a popular way to anonymize relation data. Essentially, generalization can be regarded as clustering ver-

tices and edges into groups and generalize all members in a group to the same.

Depending on the subjects of clustering, the clustering-based approaches can be further divided into four subcategories: vertex clustering methods, edge clustering methods, vertex and edge clustering methods, and vertex-attribute mapping clustering methods.

4.1 Vertex Clustering Methods

Hay *et al.* [12] considered a simple graph model, in which vertices and edges are unlabeled. They addressed vertex identifier attacks, and proposed a vertex clustering approach. Three models of external information were considered as the possible background knowledge of an adversary. These models represent a range of structural information that may be available to an adversary, including complete and partial descriptions of vertex neighborhoods, and connections to hubs in the network. The authors formalized the structural indistinguishability of a vertex with respect to an adversary with external information about the local neighborhood of the vertex. Specifically, background knowledge of adversaries are modeled using the following types of queries.

- **Vertex refinement queries**. This class of queries, with increasing attack power, model the local neighborhood structure of a vertex in the network. Consider a vertex v, the weakest knowledge query, denoted as $\mathcal{H}_0(v)$, returns the label of v. Since unlabeled graph is considered, $\mathcal{H}_0(v)$ returns $\emptyset$ only in such a case. The queries are successively more descriptive. $\mathcal{H}_1(v)$ returns the degree of v, and $\mathcal{H}_2(v)$ returns the multiset of degrees of v's neighbors. Generally, query $\mathcal{H}_i(v)$ returns the multiset of values which are the results of evaluating $\mathcal{H}_{i-1}$ on the set of vertices adjacent to v, that is,

  $$\mathcal{H}_i(v) = \{\mathcal{H}_{i-1}(u_1), \mathcal{H}_{i-1}(u_2), \ldots, \mathcal{H}_{i-1}(u_m)\},$$

  where $u_1, \ldots, u_m$ are the vertices adjacent to v.

- **Subgraph queries**. This class of queries assert the existence of a subgraph around the target vertex. The descriptive power of a subgraph query is measured by the number of edges in the subgraph. An adversary is assumed to be able to gather a fixed number of edges in a subgraph around a target vertex v. By exploring the neighborhood of v, the adversary is capable of identifying whether a subgraph exists around v. The existence of this subgraph can be expressed as a query, and the adversary's knowledge can be modeled by granting the answer to such a query.

- **Hub fingerprint queries**. This class of queries model the connections of a vertex to a set of selected hubs in the network. A hub is defined as a vertex in a network with high a degree and a high betweenness centrality. Hubs are important components of the topology of networks. A hub fingerprint for a target vertex v is defined as the vector of distances between v and a set of hubs.

Several graph properties are considered to be the utility of the network, including the degree distribution of vertices, the distribution of shortest path lengths of 500 randomly sampled pairs of vertices in the network, the distribution of clustering coefficients which are the proportion of all possible neighbor pairs of a vertex that are connected, network resilience which is the number of vertices in the largest connected component of the graph when vertices are removed in degree decreasing order, and infectiousness which is the proportion of vertices infected by a hypothetical disease and simulated by first infecting a randomly chosen vertex and then transmitting the disease to each neighbor with the specified infection rate.

Hay *et al.* [12] proposed a scheme of anonymity through structural similarity. Vertices that look structurally similar may be indistinguishable to an adversary. A strong form of structural similarity between vertices is automorphism equivalence.

The anonymization technique proposed in [12] is a vertex clustering approach. It generalizes an input network by grouping vertices into partitions and publishing the number of vertices in each partition along with the densities of edges within and across partitions. Data analysts can still use the anonymized graphs to study macro-properties of the original graph.

The partitioning of vertices is chosen such that the generalized graph satisfies the privacy preservation goals and maximizes the data utility. To ensure anonymity, we need to make sure that any adversary has at least a minimum level of uncertainty about the re-identification of any target vertex. Hay *et al.* [12] proposed to use the size of a partition to provide a basic guarantee against re-identification attacks, which mimics k-anonymity in relational data. Specifically, for any partition of vertices, the size should be at least k.

To retain utility as much as possible, the partitions should best fit the input graph. The proposed method estimates fitness via a maximum likelihood approach. A local search is adopted to explore the exponential number of possible partitionings. To find the partitioning that maximizes the likelihood function, the algorithm uses simulated annealing [22].

4.2 Edge Clustering Methods

In general, a social network can have different types of vertices and different types of edges. Zheleva and Getoor [37] focused on the case where there are multiple types of edges but only one type of vertices. Among all types of edges, one type is assumed sensitive and should be protected against link re-identification attacks. The privacy breach is measured by counting the number of sensitive edges that can be inferred from the anonymized data.

To model the background knowledge of adversaries, the authors considered predicting sensitive edges based on the other observed non-sensitive edges. To address the worst case, the authors assumed that an adversary has an accurate probabilistic model which can predict the existence of a sensitive edge e^s_{ij} (that is, an edge between two vertices v_i and v_j carrying a sensitive label s) based on a set of observations $\mathbf{O} : P(e^s_{ij}|\mathbf{O})$, where each observation is an edge. A simple noisy-or model [21] for the existence of the sensitive edge is adopted. The noisy-or model can capture the scenario where each observed edge contributes to the probability of the existence of a sensitive edge.

The authors assumed that each observed edge e_k has a noise parameter λ_k, which models the independent influence of e_k on the existence of a sensitive edge. In addition, they assumed that there exists a leak parameter λ_0 which models

the probability of the existence of a sensitive edge due to some other hidden factors. According to the noisy-or model, the probability of the existence of a sensitive edge is calculated as

$$P(e^s_{ij} = 1) = P(e^s_{ij} = 1|e_1, \ldots, e_n) = 1 - \prod_{k=0}^{n}(1 - \lambda_k).$$

An adversary succeeds when she/he can correctly figure out whether a sensitive edge exists between two vertices.

In order to model the data utility, the authors proposed to count the number of observations which have to be deleted during the anonymization process. The smaller the number of removed observations, the higher the utility.

In order to protect sensitive relationships, several graph anonymization strategies are proposed. The first edge anonymization strategy is to only remove the sensitive edges, leaving all other observed edges intact.

Another anonymization strategy is to remove some observed edges. Generally, a particular type of observations which significantly contributes to the overall likelihood of a sensitive relationship, or a certain percentage of observations that meet some pre-specified criteria (for example, at random, connecting high-degree vertices, etc.) can be removed. The most conservative anonymization strategy is to remove all edges in the network. Obviously, in the above approaches, the utility of an anonymized network is low.

The authors assumed that the vertices are divided into equivalence classes and each class is anonymized properly using some existing relational data anonymization method. Then, a more effective approach to anonymize the social network is to collapse all vertices in an equivalence class into a single vertex, and consider which edges to be included in the collapsed graph. One feasible way is to publish for each edge type the number of edges of the type between two equivalence class vertices. This approach is called cluster-edge anonymization.

The difference between the cluster-edge anonymization approach and the approach in [12] is that the cluster-edge anonymization method aggregates edges on type to prevent the disclosure of sensitive relationships, while [12] clusters vertices to protect vertex identities.

4.3 Vertex and Edge Clustering Methods

Campan and Truta [4] modeled a social network as a simple undirected graph. Moreover, vertices in the network are associated with some attributes. Following the previous models in relation data, the attributes associated with vertices can be classified into three categories, identifier attributes such as `name` and `SSN` which should be removed in publishing, quasi-identifier attributes such as `zipcode` and `sex` which may be used by an adversary in re-identification attacks, and sensitive attributes such as `diagnosis` and `income` which are assumed to be privacy information. Furthermore, in [4], edges are not labeled.

To model data utility, Campan and Truta [4] consider the information loss due to generalization and the changes of structural properties. Information loss occurs when vertex labels are generalized. The changes of structural properties quantify the probability of error when one tries to reconstruct the structure of the original social network from the masked version.

To protect privacy in social network data, Campan and Truta [4] advocates the k-anonymity model. Every vertex should be indistinguishable with at least other $(k-1)$ vertices in terms of both the attributes and the associated structural information such as neighborhood of vertices. The anonymization method disturbs as little as possible the social network data, both the attribute data associated to the vertices and the structural information.

The method for anonymizing vertex attribute data uses generalization, which has been well studied in relational data. For structure anonymization, the proposed method is called edge generalization, which is similar to the one described in [37] to some extent. The critical difference is that the method in [4] takes into account both the generalization information loss and the structural information loss during the clustering procedure. This process can be tuned by users to achieve a desirable tradeoff between preserving more structural information of the network and preserving more vertex attribute information.

Similar to [37], in [4], vertices are partitioned into clusters in anonymization. To anonymize edges, vertices in the same cluster are collapsed into one single vertex, labeled with the number of vertices and edges in the cluster. The edges between two clusters are collapsed into a single edge, labeled by the number of edges between them.

4.4 Vertex-attribute Mapping Clustering Methods

In some applications, entities and their relationships can be modeled as a bipartite graph, such as customers and medical products used. The edges in such a bipartite graph may be considered as privacy. Cormode *et al.* [5] focused on the problem of anonymizing bipartite graphs.

Generally, a bipartite graph $G = (V, W, E)$ consists of $|V|$ vertices of one type and $|W|$ vertices of the other type, and a set of $|E|$ edges $E \subseteq V \times W$. When a bipartite graph is published, the graph structure is retained. The vertices are clustered into groups and the mapping between groups in the original graph and groups in the published graph is released. For example, the mapping table may state that vertices $\{x_1, x_2, x_3\}$ in the original graph are mapped to $\{a_{20}, a_{31}, a_{206}\}$ in the published graph. By devising the mapping properly, privacy of entities such as whether a customer consumes a specific product can be preserved.

To model the background knowledge of adversaries, Cormode *et al.* [5] consider both static attacks and learned link attacks. If a group of vertices $X \subset V$ only connect to a group of vertices $Y \subset W$, a static attack can immediately obtain the vertices that those in X connect to. Generally, if very few edges exist between vertices in X and vertices not in Y, then a learned link attack can obtain the vertices that those in X connect to with a high confidence.

The data utility is measured by the accuracy of answering aggregate queries, such as the average number of products purchased per user. Attributes of vertices in V (or W), or both can be used to compose predicates in aggregate queries, such as the average number of products purchased by customers in California, and the average number of vitamin products by customers in California.

Cormode *et al.* [5] proposed the safe grouping mechanism to protect privacy. A safe grouping of a bipartite graph partitions vertices into groups such that two vertices in the same group of V have no common neighbors in W and vice

versa. To control the anonymization granularity, a (k, l)-safe grouping ensures that each group on V contains at least k vertices and each group on W contains at least l vertices.

A greedy algorithm is developed, which may or may not find a safe grouping. The vertices are processed one by one. For each vertex, the algorithm checks whether it can be put into an existing group without breaking the safety. If yes, it is added into a group. Otherwise, a new group is created. After all vertices are processed, there may be some groups with fewer than k vertices. Those vertices are collected and the algorithm continues to run on the collection with a larger group size threshold, say $(k+1)$. The iteration continues until either a safe grouping is found or the group size threshold exceeds the number of vertices in the collection of vertices to be partitioned. In the latter case, the algorithm fails.

5. GRAPH MODIFICATION APPROACH

The clustering-based approaches reduce a cluster of vertices and edges into a super-vertex. Thus, the graph may be shrunk considerably after anonymization, which may not be desirable for analyzing local structures. To preserve the scale and the local structures of the original graph, graph modification approaches try to locally modify the graph structure to achieve the privacy preservation requirement.

5.1 Optimization Graph Construction Methods

Liu and Terzi [18] studied the k-degree anonymization problem on social networks without any vertex and edge labels. They considered the identity disclosure scenario where the identities of individuals associated with vertices are revealed.

To model the background knowledge of an adversary, the authors considered possible re-identification attacks against individuals by an adversary using the prior knowledge of the degree of a target vertex. An adversary is assumed to know the degree of a target victim. By searching the degrees of vertices in the published network, the adversary may be able to identify the individual, even when the identities of the vertices are removed before the network data is published.

Several graph properties are considered as utility of the networks, including clustering coefficient, average path length, and edge intersection (i.e., the percentage of edges in the degree-anonymous graphs that are also in the original graph).

In order to battle degree attacks, Liu and Terzi [18] proposed the notion of graph k-degree anonymity, which mimics k-anonymity in relational data. Specifically, a graph is said to be k-degree anonymous if for every vertex v in the graph, there exist at least $(k-1)$ other vertices in the graph with the same degree as v. An adversary with degree background knowledge can identify some target individuals with the probability at most $\frac{1}{k}$.

For a graph $G(V, E)$, the degree sequence of G, denoted by $\mathbf{d}$, is a sequence of vertices in the degree descending order. A degree sequence is k-degree-anonymous if, for each vertex, there are at least other $(k-1)$ vertices carrying the same degree. By providing a privacy parameter k, the anonymization method proceeds in two steps.

In the first step, starting from the original degree sequence $\mathbf{d}$, Liu and Terzi [18] developed a dynamic programming method to construct a new degree sequence $\hat{\mathbf{d}}$ that is k-degree-anonymous and minimizes the degree anonymization cost $D_A(\hat{\mathbf{d}} - \mathbf{d}) = L_1(\hat{\mathbf{d}} - \mathbf{d})$. Therefore, their method is optimal in terms of the resulting degree sequence.

In the second step, they constructed a graph $\hat{G}(V, \hat{E})$ such that $\hat{\mathbf{d}}$ is the degree sequence of $\hat{G}$ and $\hat{E} \cap E = E$ (or $\hat{E} \cap E \approx E$ in a relaxed version). The graph construction problem is related to the problem of realizing degree sequence with constraints, which has been studied extensively in graph theory [6; 11]. Generally speaking, the method of graph construction follows a randomized scheme. To achieve the desired degree sequence, a randomized edge swap transformation strategy was adopted in [18].

5.2 Randomized Graph Modification Approaches

Randomized approaches have been widely used in privacy preservation in relation data [7; 1; 27]. The approach has also been adopted for anonymizing social network data.

5.2.1 *Randomized Edge Construction Methods*

Hay *et al.* [13] tackled the same problem as [12] (reviewed in Section 4.1) except that hub fingerprint queries are not considered. They developed a randomized edge construction method.

The method constructs an anonymized graph G' from the original graph G through a sequence of m edge deletions followed by m edge insertions. Edges deleted are chosen uniformly at random from the set of all existent edges in G, while edges inserted are chosen uniformly at random from the set of all non-existent edges of the interim graph. The vertices are not changed. The process of perturbation and the perturbation parameter m are assumed to be publicly known.

An adversary may attempt to re-identify individuals using external information, such as vertex refinement queries and subgraph queries as discussed in Section 4.1. The perturbation of the graph ensures that the adversary cannot simply exclude from the candidate set of vertices (that is, a set of vertices matching the adversary's background knowledge about the target vertex) those do not match the structural properties of the target. The adversary must consider the set of possible worlds implied by the anonymized graph G' and m random edge insertions and m deletions. The set of possible worlds consist all graphs that can result in G' under m edge perturbations.

The candidate set of a target vertex v includes all vertices $u \in G'$ such that u is a candidate in some possible world. Any vertex that may be a candidate for v will also be a candidate under graph perturbation, since G is a possible world of G'. The candidate set may become very large with an increased number of perturbation operations. Consequently, the privacy of target individuals can be well protected.

5.2.2 *Randomized Spectrum Preserving Methods*

Ying and Wu [36] tackled the same problem as [13] by considering randomly adding/deleting edges or randomly switching edges. Instead of designing specific randomization algorithms, Ying and Wu [36] analyzed the effect of randomization in protecting attacks.

The spectrum of a graph is defined as the set of eigenvalues of the adjacency matrix of the graph. The eigenvalues of a network are connected to important topological properties such as diameter, presence of cohesive clusters, long paths

and bottlenecks, and randomness of the graph. Ying and Wu [36] showed that the spectrum property has close relation with many graph characteristics and can provide global measures for some network properties. Furthermore, Ying and Wu [36] investigated the spectrum of networks.

A natural idea for graph anonymization is to consider whether a graph can be perturbed without significantly changing one or some particular eigenvalues. If so, the approach is probable to better preserve structural characteristics. By considering the change of spectrum in the randomization process, the proposed spectrum preserving approach [36] can outperform the simple edge randomization methods. The algorithm can determine which edges should be added, removed or switched so that the change of the eigenvalues can be under control.

5.2.3 *Randomized Weight Perturbation Methods*

In some applications, the social networks may be weighted. The weights of edges can reflect affinity between two vertices or record the communication cost between two individuals. For example, a social network about communication between friends may be weighted such that the weight of an edge is the communication frequency between two individuals, which may be considered the privacy for some people. Liu *et al.* [19] studied anonymization of graphs where edge weights are considered sensitive. Two kinds of data utility were considered. First, the authors considered how to approximate the lengths of shortest paths between vertices within an error bound. Second, they considered how to retain the exact shortest paths between vertices in a selected subset.

Two methods were proposed. The first method uses Gaussian randomization multiplication. The authors showed that there does not exist a perturbation schema such that every edge weight is perturbed but the length of the shortest paths between every pair of vertices is preserved. Thus, they used a Gaussian noise matrix with mean 0 and standard deviation σ to perturb the weights of edges so that the shortest path lengths are approximated with a quality guarantee.

The second method is a greedy perturbation algorithm, which not only can keep exactly the same shortest paths for certain selected paths and approximate the shortest path lengths for the others, but also can maximize the weight privacy preservation. Generally, any edge $e_{i,j}$ can be in one of the three categories: a non-betweenness edge (that is, it is not on any shortest path in the graph), an all-betweenness edge (that is, all shortest paths in a set of pre-selected vertices pass through $e_{i,j}$), and a partial-betweenness edge (that is, only some of the shortest paths pass through this edge). On different categorizes of edges special weight modifications can be applied. The anonymization method perturbs the weights greedily until the privacy preservation requirement is achieved.

5.3 Greedy Graph Modification Approaches

Greedy approaches have been widely used in privacy preservation in relational data [34; 15; 10], and have been shown effective. They also can be used in anonymizing social network data.

Zhou and Pei [39] considered anonymization in social networks where each vertex is associated with non-sensitive attributes. An attacker may have background knowledge about the neighborhoods of victims. The privacy preservation goal is to protect neighborhood attacks which use neighborhood matching to re-identify vertices.

Consider a social network $G = (V, E, L, \mathcal{L})$ and the anonymization $G' = (V', E', L', \mathcal{L}')$ for publishing. Zhou and Pei [39] assumed that no fake vertices should be added to the anonymized graph. This assumption is often desirable in applications since introducing fake vertices may often change the global structure of a social network. Moreover, they assumed that the original connections between vertices in G are retained in the anonymization. To model the utility, they focused on using anonymized social networks to answer aggregate network queries.

To battle neighborhood attacks, Zhou and Pei [39] extended the k-anonymity model in relational data to social network data. For a social network G, suppose an adversary knows the neighborhood structure of a vertex $u \in V(G)$, denoted by $Neighbor_G(u)$. If $Neighbor_G(u)$ has at least k isomorphic copies in G' where G' is an anonymization of G, then u can be re-identified in G' with a confidence of at most $\frac{1}{k}$. Zhou and Pei [39] introduced a practical greedy method to anonymize a social network to satisfy the k-anonymity requirement in two steps.

In the first step, the algorithm extracts the neighborhoods of all vertices in the network. To facilitate the isomorphism tests among neighborhoods of different vertices which will be conducted frequently in anonymization, a simple yet effective neighborhood component coding technique based on minimal DFS code [35] was proposed which represents neighborhoods in a concise way.

In the second step, the algorithm greedily organizes vertices into groups and anonymizes the neighborhoods of vertices in a group to the same, until the graph satisfies k-anonymity. Due to the well recognized power law distribution of the degrees of vertices in large social networks, a heuristic of starting with those vertices of high degrees is adopted. The intuition is that in real social networks, those vertices with large degrees are the ones vulnerable to neighborhood attacks.

Zhou and Pei [38] extended [39] and introduced l-diversity into social network anonymization. In this case, each vertex is associated with some non-sensitive attributes and some sensitive attributes. If an adversary can re-identify the sensitive attribute values of one target individual with a high confidence, the privacy of that individual is breached. An l-diverse graph makes sure that the adversary cannot infer the sensitive attribute values with a confidence over $\frac{1}{l}$.

Zhou and Pei [38] extend the k-anonymity method developed in [39] to tackle the l-diversity problem. The diversity partitioning strategy is similar to that in [32].

6. CONCLUSIONS

In this paper, we surveyed a few recent studies on anonymization techniques for privacy preserving publishing of social network data. Although privacy preserving data publishing and analysis techniques in relational data have been well explored, the research and development of anonymization techniques on social network data is still in its infancy, as illustrated in Table 1.

We discussed the new challenges in privacy preservation in social network data comparing to the extensively studied relational case, and examined the possible problem formulation in three important dimensions: privacy, back-

ground knowledge, and data utility. We reviewed the anonymization methods for privacy preservation in two categories: clustering-based approaches and graph modification approaches.

More extensively, there are also some other studies related to privacy preservation in social network data. For example, Frikken and Golle [9] studied the problem of constructing a graph from individuals who are vertices in the graph without intruding the privacy of the individuals. Wang *et al.* [28] proposed using description logic as a knowledge representation in social network data publishing. Leskovec and Faloutsos [16] proposed a method to generate a graph fitting the graph properties of a give graph. The graph generated can be used as a (perturbed) anonymization of the original graph.

As social network data is much more complicated than relational data, privacy preserving in social networks is much more challenging and needs many serious efforts in the near future. Particularly, modeling adversarial attacks and developing corresponding privacy preservation strategies are critical.

7. REFERENCES

[1] R. Agrawal, R. Srikant, and D. Thomas. Privacy preserving olap. In *Proceedings of the 2005 ACM SIGMOD international conference on Management of data (SIGMOD'05)*, pages 251–262, New York, NY, USA, 2005. ACM.

[2] N. Alon and J. Spencer. *The Probabilistic Method*. John Wiley, 1992.

[3] L. Backstrom, C. Dwork, and J. Kleinberg. Wherefore art thou r3579x?: anonymized social networks, hidden patterns, and structural steganography. In *Proceedings of the 16th international conference on World Wide Web (WWW'07)*, pages 181–190, New York, NY, USA, 2007. ACM Press.

[4] A. Campan and T. M. Truta. A clustering approach for data and structural anonymity in social networks. In *Proceedings of the 2nd ACM SIGKDD International Workshop on Privacy, Security, and Trust in KDD (PinKDD'08), in Conjunction with KDD'08*, Las Vegas, Nevada, USA, 2008.

[5] G. Cormode, D. Srivastava, T. Yu, and Q. Zhang. Anonymizing bipartite graph data using safe groupings. In *Proceedings of the 34th International Conference on Very Large Databases (VLDB'08)*. ACM, 2008.

[6] R. Diestel. *Graph Theory (3rd Edition)*, volume 173. Springer-Verlag, Heidelberg, 2005.

[7] A. Evfimievski, J. Gehrke, and R. Srikant. Limiting privacy breaches in privacy preserving data mining. In *Proceedings of the twenty-second ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems (PODS'03)*, pages 211–222, New York, NY, USA, 2003. ACM.

[8] L. C. Freeman, D. R. White, and A. K. Romney. *Research Methods in Social Network Analysis*. George Mason University Press, Fairfax, VA, 1989.

[9] K. B. Frikken and P. Golle. Private social network analysis: how to assemble pieces of a graph privately. In *Proceedings of the 2006 ACM Workshop on Privacy in the Electronic Society (WPES'06)*, pages 89–98. ACM, 2006.

[10] B. C. M. Fung, K. Wang, and P. S. Yu. Anonymizing classification data for privacy preservation. *IEEE Transactions on Knowledge and Data Engineering*, 19(5):711–725, 2007.

[11] J. Gross and J. Yellen. *Graph theory and its applications*. CRC Press, Inc., Boca Raton, FL, USA, 1999.

[12] M. Hay, G. Miklau, D. Jensen, and D. Towsley. Resisting structural identification in anonymized social networks. In *Proceedings of the 34th International Conference on Very Large Databases (VLDB'08)*. ACM, 2008.

[13] M. Hay, G. Miklau, D. Jensen, P. Weis, and S. Srivastava. Anonymizing social networks. Technical Report 07-19, University of Massachusetts Amherst, 2007.

[14] J. M. Kleinberg. Challenges in mining social network data: processes, privacy, and paradoxes. In P. Berkhin, R. Caruana, and X. Wu, editors, *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'07)*, pages 4–5, San Jose, California, USA,, August 12-15, 2007 2007. ACM.

[15] K. LeFevre, D. J. DeWitt, and R. Ramakrishnan. Mondrian multidimensional k-anonymity. In *Proceedings of the 22nd International Conference on Data Engineering (ICDE'06)*, page 25, Washington, DC, USA, 2006. IEEE Computer Society.

[16] J. Leskovec and C. Faloutsos. Scalable modeling of real graphs using kronecker multiplication. In *ICML '07: Proceedings of the 24th international conference on Machine learning*, pages 497–504, New York, NY, USA, 2007. ACM.

[17] 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 Data Mining*. CRC Press, 2008.

[18] K. Liu and E. Terzi. Towards identity anonymization on graphs. In *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data (SIGMOD'08)*, pages 93–106, New York, NY, USA, 2008. ACM Press.

[19] L. Liu, J. Wang, J. Liu, and J. Zhang. 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, 2008.

[20] A. Machanavajjhala, J. Gehrke, D. Kifer, and M. Venkitasubramaniam. L-diversity: Privacy beyond k-anonymity. In *Proceedings of the 22nd IEEE International Conference on Data Engineering (ICDE'06)*, Washington, DC, USA, 2006. IEEE Computer Society.

[21] J. Pearl. *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1988.

[22] S. J. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach*. Pearson Education, 2003.

[23] P. Samarati and L. Sweeney. Generalizing data to provide anonymity when disclosing information. In *Proceedings of the 7th ACM SIGACT-SIGMOD-SIGART symposium on Principles of database systems (PODS'98)*, page 188, New York, NY, USA, 1998. ACM Press.

[24] J. Scott. *Social Network Analysis Handbook*. Sage Publications Inc., 2000.

[25] J. Srivastava, M. A. Ahmad, N. Pathak, and D. K.-W. Hsu. Data mining based social network analysis from online behavior. Tutorial at the 8th SIAM International Conference on Data Mining (SDM'08), 2008.

[26] L. Sweeney. K-anonymity: a model for protecting privacy. *International Journal on uncertainty, Fuzziness and Knowledge-based System*, 10(5):557–570, 2002.

[27] Y. Tao, X. Xiao, J. Li, and D. Zhang. On anti-corruption privacy preserving publication. In *Proceedings of the 24th International Conference on Data Engineering (ICDE'08)*, pages 725–734, 2008.

[28] D.-W. Wang, C.-J. Liau, and T. sheng Hsu. Privacy protection in social network data disclosure based on granular computing. In *Proceedings of the 2006 IEEE International Conference on Fuzzy Systems*, pages 997–1003, Vancouver, BC, Canada, July 16-21, 2006 2006.

[29] S. Wasserman and K. Faust. *Social network analysis: Methods and applications*. Cambridge University Press, 1994.

[30] D. J. Watts and S. H. Strogatz. Collective dynamics of "small-world" networks. *Nature*, 393(6684):440–442, June 1998.

[31] B. Wellman. For a social network analysis of computer networks: a sociological perspective on collaborative work and virtual community. In *Proceedings of the 1996 ACM SIGCPR/SIGMIS Conference on Computer Personnel Research (SIGCPR'96)*, pages 1–11, New York, NY, USA, 1996. ACM Press.

[32] X. Xiao and Y. Tao. Anatomy: Simple and effective privacy preservation. In U. Dayal, K.-Y. Whang, D. B. Lomet, G. Alonso, G. M. Lohman, M. L. Kersten, S. K. Cha, and Y.-K. Kim, editors, *Proceedings of the 32nd International Conference on Very Large Data Bases (VLDB'06)*, pages 139–150. ACM, 2006.

[33] X. Xiao and Y. Tao. Personalized privacy preservation. In *Proceedings of the 2006 ACM SIGMOD international conference on Management of data (SIGMOD'06)*, pages 229–240, New York, NY, USA, 2006. ACM Press.

[34] J. Xu, W. Wang, J. Pei, X. Wang, B. Shi, and A. W.-C. Fu. Utility-based anonymization using local recoding. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD'06)*, pages 785–790, New York, NY, USA, 2006. ACM Press.

[35] X. Yan and J. Han. gspan: Graph-based substructure pattern mining. In *Proceedings of the 2002 IEEE International Conference on Data Mining (ICDM'02)*, page 721, Washington, DC, USA, 2002. IEEE Computer Society.

[36] X. Ying and X. Wu. Randomizing social networks: a spectrum preserving approach. In *Proceedings of the 2008 SIAM International Conference on Data Mining (SDM'08)*, pages 739–750. SIAM, 2008.

[37] E. Zheleva and L. Getoor. Preserving the privacy of sensitive relationships in graph data. In *Proceedings of the 1st ACM SIGKDD Workshop on Privacy, Security, and Trust in KDD (PinKDD'07)*, 2007.

[38] B. Zhou and J. Pei. The k-anonymity and l-diversity approaches for privacy preservation in social networks against neighborhood attacks. *Submitted for publication.*

[39] B. Zhou and J. Pei. Preserving privacy in social networks against neighborhood attacks. In *Proceedings of the 24th IEEE International Conference on Data Engineering (ICDE'08)*, pages 506–515, Cancun, Mexico, 2008. IEEE Computer Society.