Privacy Risks and Countermeasures in Publishing and Mining Social Network Data

Chiemi Watanabe
Department of Information Science,
Faculty of Science Ochanomizu University
Bunkyo-ku, Tokyo 112–8610
E-mail: chiemi@is.ocha.ac.jp

Toshiyuki Amagasa
Graduate School of Systems
and Information Engineering
University of Tsukuba
Tsukuba, Ibaraki, 305–8573
E-mail: amagasa@cs.tsukuba.ac.jp

Ling Liu
School of Computer Science
Georgia Institute of Technology
Atlanta, Georgia 30332–0250
Email: lingliu@cc.gatech.edu

***Abstract*—As interests in sharing and mining social network data continue to grow, we see a growing demand for privacy preserving social network data publishing. In this paper, we discuss privacy risks in publishing social network data and the design principles for developing countermeasures. The main contributions of this study are three folds. First, to the best of our knowledge, we make the first attempt to define the utility of released data in terms of exposure levels and query types, assuming queries are the most fundamental operations in social network analysis. We argue that using information exposure levels to characterize the utility of anonymized data can be used as a general and usage-neutral metric and query types can be used as the baseline usage driven utility metric. Second, we identify two types of background knowledge based inference attacks that can break some of most representative graph permutation based anonymization techniques in terms of anonymity violations. Third but not the least, we describe some design considerations for developing countermeasures in privacy preserving social network data publishing.**

***Index Terms*—social network,privacy,attack model**

I. Introduction

Social network analysis is gaining growing attraction as a tool of creating innovative marketing strategies, developing new social computing applications, carrying out sociological research and field studies by historians and genealogists. Generally speaking, a social network is modeled as a graph consisting of a set of entities and the connections between them. In addition to the graph structure, in many cases, social network data also contains descriptive and possible sensitive information about the entities, such as age, gender, address, professional and business affiliation, hobbies, and social clubs and activities. Such information is typically captured in the form of entity profiles, each corresponding to a node in the social network graph. Social network analysis is typically performed over either its graph structure or profile or both of them. It is widely recognized that social network data is generated through crowd sourcing. Typically each entity is the creator and owner of its own social network membership information and its social ties and determines which parts of its social network information can be accessible and to whom they can be made assessable and for how long.

With the continued revolution of social computing technologies, social network providers and many enterprises and government organizations are interested in privacy preserving publishing and mining of social network data. However, sharing and mining social network data should not intrude the personal privacy of individuals. Thus, data anonymization techniques are considered essential for safe and secure publishing and mining of social network data for a number of reasons. First, it is mutually beneficial to share social network data in order to allow third-parties to try new analysis and mining techniques not thought of by the data owner and to improve commercial and government services based on the need of end-users. Thus, it is critical to provide utility-driven data anonymization such that third party data mining service providers do not gain access to social network data that is unauthorized. Second, many enterprises and organizations need to anonymize user-generated data for data retention and usage purpose. For instance, many government privacy regulation such as HiPPA prevent companies from retaining customer information indefinitely. Data anonymization provides an unique opportunity for data retention. Google progressively anonymizes IP addresses in search logs.

Privacy is a complex concept. It is commonly acknowledged that data subjects have inherent right and expectation of privacy. Most companies have adopted a privacy policy and significant legal framework is established relating to privacy, such as UN Declaration of Human Rights, US Constitution, HIPAA, Video Privacy Protection, Data Protection Acts. US Census releases census data about every US household, who, where, age, gender, racial, income and educational data. This release enables study and research to determine representation and planning. US Census data is anonymized by aggregating to geographic areas (Zip code), broken down by various combinations of dimensions, and released in full after 72 years. In a similar spirit, NetFlix released 100M dated ratings from 480K users to 18K movies in order to draw talents to improve predicting ratings of unlabeled examples. Although no documentation on how exactly NetFlix anonymized their raw data prior to release, it is known that all identifier-based customer information are removed and only subset of full data is made available with dates modified and some ratings deleted and movie title and year published in full. Thus we argue that privacy in data publishing is sophisticated and domain specific

COLLABORATECOM 2011, October 15-18, Orlando, United States

DOI 10.4108/icst.collaboratecom.2011.247177

in nature.

In the context of social network data publishing, many researchers have proposed anonymization techniques for sanitizing social network data before releasing it for third party mining services [6], [7], [12], [18], [22]. It is widely recognized that the primary privacy risks in publishing social network data is centered around the inference attacks, namely high confidence inference of associations of published data with some background identity information of known individuals, thus leading to intrusion of privacy of such individuals. For example, one may infer salary for an individual in " census ", or infer individual' s viewing history in " video " or individual' s search history in " search ". All of these risks are inferred by linking sensitive information to some identity of an individual that is available as background knowledge or common sense or domain-specific knowledge. Example background knowledge includes facts about the released data set such as A dated B, A dined in restaurant B, or A has disease Y. Example common sense includes common knowledge about entities involved in the data set such as teen is children between 13 and 18. Example domain knowledge includes broad properties of data, such as breast cancer is rare in men. Another important factor that impacts on privacy risks of publishing social network data is the utility of released data. Anonymization is meaningless if the utility of anonymized data is close to zero or completely lost. If we consider the original data to have full utility (one end of the spectrum), then the empty data set should be considered to have perfect privacy (the other end of the spectrum). Thus, privacy preserving publishing of social network data should aim at preserving privacy required by users (social network entities) while maintaining the maximum utility of released data. To achieve this goal, we face a number of technical challenges. First, how should we define utility? Utility of data is heavily dependent on the usage model of the data. For instance, for data clustering services, distance preservation is the most important utility of the released dataset if we want to maintain high quality of clustering results. However, if the released social network data is used for community detection[5], then preserving the original graph structure is perhaps the most important utility. Second, one of the most honorable goals of data publishing is the potential to support new applications that are unknown at the time of data release or to retain data for future unknown business analysis. The third challenge is the need to quantify not only the privacy guarantee but also the utility of different anonymization methods.

In this paper, we discuss privacy risks in publishing social network data and the design principles for developing countermeasures. We broadly classify anonymization techniques for social network data publishing into two representative categories: relational table based anonymization and graph based anonymization. The former can be characterized by k-anonimity based data perturbation and the later can be characterized by structural permutation. The main contributions of this study are three-fold. First, to the best of our knowledge, we make the first attempt to define the utility of released data in terms of exposure levels and query types, assuming queries are the most fundamental operations in social network analysis. We argue that using information exposure levels to characterize the utility of anonymized data can be used as a general and usage-neutral metric and query types can be used as the baseline usage driven utility metric. Second, we identify two types of background knowledge based inference attacks that can break some of most representative graph permutation based anonymization techniques in terms of anonymity violations. Third but not the least, we describe some design considerations for developing countermeasures in privacy preserving social network data publishing. We conclude with a brief overview of related work and an outlook of future research directions and our research agenda.

II. Social Network Reference Model

Most of the social network services today model and maintain social ties among their members through two types of social relationship representations: Friendships and activity groups. By friend relationships, people get connected through the friend request and agreement protocol to establish the mutual confirmation of friendship. By activity group relationships, people are getting connected through engaging in the same type of activities or events. In fact, friendship can be viewed as one specific type of activity group. Recently Facebook has introduced the ability to establish geographical groups of friends for its members who are interested in such new activity-oriented social grouping features. One of the big advantages of promoting activity based classification of social ties is to facilitate finer granularity of exploration on social relationships among people to enhance personal and business experiences of social networks.

Conceptually, a social network can be represented as a graph $G = (V, E)$, where V is a set of nodes and E is a set of edges, each representing a type of relationship between a pair of nodes in G. When we constrain the nodes of G denoting the members of a social network and the edges denoting the friend relationship among nodes, $G = (V, E)$ presents a social network of people connected through their friend relationships. When we model a social network of people by two types of nodes: member nodes and activity-based group nodes, the edges are representing the engagement or a participation of a member in a specific activity or group, $G = (V, E)$ now presents an activity-group based social network. Based on the types of edges, we refer to the former as a *user-user link* graph and the latter as a *user-group link* graph. A user-user link graph is specific and homogeneous as it is comprised of users as nodes and friend relationship as an edge between a pair of nodes. In comparison, a user-group link graph is more general and heterogeneous as it is comprised of both user nodes and group nodes. User nodes vary from user to user and group nodes are activity driven and vary from group to group. Users engage in multiple activities belong to multiple groups, such as tennis, football, swimming or debate. Also an edge can only be established between a user node and an activity node and it represents the group participation or engagement relationship.

In this paper we model a social network as a user-group graph for a number of reasons. First, a user-user link can be expressed using two user-group links with the friendship node as the common group node. Thus we can view a user-user link graph as a subgraph of a user-group graph. For example, we represent the fact that v_1 and v_2 are friends by using two user-group links, a group node "friend" h and connecting between h and v_1 as well as between h and v_2 respectively. In addition, by using user-group link, we are no longer constrained to only friend relationship and we can represent and explore different kinds of relationships between users. Furthermore, user-group links can represent relations among more than two users. In fact, user-group links are richer in capturing different types of user-user links and user-group links as well as group-group links. Most importantly, companies invested in social network data mining services are more interested in user-group links than simple user-user friendship links due to the amount and richness of the social information inherent in such user-group links. For example, targeted marketing is gaining increasing attractions in social networks due to activity-oriented and geo-location-based grouping data.

Formally, we model a user-group link graph using we the bipartite graph $G = (V, W, E)$, where V is a set of user-node, W is a set of group-nodes, and E is a set of user-group links which establish a connection between a user-node $v \in V$ and a group-node $w \in W$. Furthermore, social network data typically include information about users, such as user's age, gender, address, hobbies, education and professional experience. We refer to such user-specific information as a user profile, which is linked to the corresponding user-node. A profile about an user can be described by a set of attributes in terms of key-value representation. Similarly, group-nodes have their activity based group specific profile. In a social network data where many kinds of group-nodes are included, we can categorize them into static groups and dynamic groups. Static groups are used to model those user activities that are relatively stable and long term, such as interest or hobby based communities for leisure or professionals, past and current work affiliations, education and school experiences, past and current residential neighborhood relationships among users, and so forth. Dynamic groups are used to capture the relationships between users and the events and activities they engage and participate. Such groups can be short-lived in terms of group validity duration. For example, a project oriented group lasts for the duration of the project. An interactive game group may be valid only for the duration of game and members of this game group are dynamically changing depending on who are the players. Similar to user nodes, group nodes are described by a set of key-value attributes, referred to as activity group profile or group profile for short. Typical attributes of group nodes include group name, spatial attributes, such as meeting place, temporal attributes, such as starting and ending time, and context attributes such as activity name, description, and so on. The use of bipartite graph allows us to differentiate user nodes from group nodes in our analysis of user privacy protection against unauthorized disclosures and leakages in publishing social network data mining results.

III. Social Network Utility Model

Privacy preserving methods for publishing social network data are categorized into data perturbation techniques and data permutation techniques, as outlined earlier. By perturbing or permuting social network data, one may reduce certain risks of unauthorized privacy leakages at the price of data utility reduction. Thus, to truly understand the privacy risks and countermeasures in publishing social network data, it is important to understand and model utility of social network data being published through utility-aware metrics.

Given that most of the social network data mining and data publishing approaches perform rich content analysis, such as community detection and node classification, and different types of content analysis require different aggregations of social network data. Thus, in order to truly understand the risks of unauthorized disclosure of sensitive user data without enumerating all possible inference attacks over published datasets, we promote to model the utility of published social network data based on both graph structure and node profiles inherent in the published datasets as well as the types of data manipulation operations applied to the published datasets. In this section, we define the utility of social network data based on the exposure levels in terms of graph structure and node profiles as well as query types.

In general, utility measures how usable the data is for social network analysis. If a piece of data can be used for answering many types of queries with high accuracy, then its utility is high. It is widely accepted that high privacy guarantee may lead to low data utility in publishing social network data. In order to manage and control the privacy risks in social network data publishing, a fundamental challenge is to understand quantitatively the privacy risks for a given utility of published social network data. In-depth understanding of how to model utility will enable us to device more effective anonymization techniques that can preserve the privacy while maximize the utility in publishing and mining social network data.

Based on the fact that social network (SN) data consists of graph structure and node profiles and queries are the fundamental operations over SN data, we first introduce the utility levels based on whether and at what degree the graph structure and/or node profiles are exposed in the published datasets. Then we introduce the operational utility based on query types. We define the utility levels in terms of *exposure levels* by introducing the following three exposure levels.

Definition 1 (Exposure Level):

- **Level 1: Exposure of only graph structure**
  In this level, data owner deletes all profile data from every node prior to publishing the SN data. Thus, social network mining services can analyze the network structure patterns over the published datasets but they cannot obtain any concrete profile information of nodes.
- **Level 2: Exposure of only profiles of nodes**
  In this level, data owner only exposes the profile data of nodes in SN data publishing and hides the graph structure

among nodes. In this case the node profile data are stored in a table and each tuple corresponds to a profile.

- **Level 3: Exposure of graph structure and profiles of nodes**
  In this level, data owner expose the graph structure and the profile of nodes though some of the structural information is perturbed or permuted in the published datasets.

Based on these exposure levels, we discuss the types of data analysis that one may perform before we analyze the privacy risks associated with each in the next section. In general, enterprises may be interested in learning statistical trends by analyzing published SN datasets. Thus, data summarization and aggregation operations, such as sum, count and average, are frequently used. Based on this observation, we categorize query types based on standard SQL aggregation and use query types to measure utility of published SN data. If the published data can answers more types of queries with high accuracy, then the utility of the published data is high.

Definition 2 (Query Types):

- **Type 0 :** Queries using graph structure only.
  E.g., finding nodes with highest node degree (the most number of edges).
- **Type 1 :** Queries using only node profiles.
  E.g., compute the number of users for each of the age groups.
- **Type 2 :** Queries with property-wise aggregation over a specific graph structure.
  This type of queries first selects nodes by using features of graph structure, and they apply aggregate operations to the selected nodes.
  E.g., count a number of people in each age group, who has more than 10 friends or who wrote more than 5 reviews in the last 2 months.
- **Type 3 :** Queries with graph-wise aggregation over specific condition matched profiles.
  This type of queries first select those nodes that match the specified conditions in terms of attribute-value of profiles, and then apply aggregate operations to the selected nodes.
  E.g., calcurate the average number of friends for those people whose are in the age group between 20 and 30.

We can observe some obvious and yet interesting connections between exposure levels and query types. When the SN data is published in exposure level 1 or 2, the published data can answer only queries in query type 0 or 1 respectively and thus provides relatively low utility. When the SN data is published in exposure level 2, the data can answer all types of queries and thus has high utility. In addition, by restricting SN data publishing only in exposure level 1 and 2, not only is the data utility low, but also relatively lower risks of privacy intrusion. For example, queries in type 0 can be leveraged for analyzing the characteristics of the graph structure of the entire social network [10]. However, it controls the privacy risks by constraining the exposure level and query type allowed for SN data publishing and mining. Even if one can select nodes with particular topological features of the graph from the published SN datasets in exposure level 1, it is hard to infer unauthorized information about selected nodes even with background knowledge. This is because neither user profile nor group profile information are made available in the published SN datasets. Unfortunately, many common and basic SN analysis, such as community detection [5], node classification [15], queries of type 2 and/or type 3 are required. Thus it is critical to understand the possible inference attacks based on the level of data exposure and query types in order to truly understand the privacy risks and device effective countermeasures in social network data publishing and mining.

IV. Social Network Data Publishing Models

A. Sensitive Attributes in Social Network

Social network data publishing typically addresses the privacy of users by removing user identity attributes in user profiles, such as user name or user ID which is used for uniquely identifying a user in a given social network. However, recent privacy research has pointed out numerous privacy attacks over perturbed social network datasets where user IDs and names are removed completely. Most of such privacy risks are due to the inference attacks over quasi-identifier attributes and sensitive attributes of user profiles. Thus we describe the social network data publishing model in terms of profile attributes and structured attributes and categorize all attributes into the following three types: (1) Identifier attributes (2) Quasi-identifier attributes, and (3) Sensitive attributes.

- **Identifier data**
  The most representative identifier attributes are user names and social security number (SSN) in user node profiles. In addition, nodeID can be an identifier attribute because nodeID can uniquely identify a profile. All data publishing services remove identifier attributes in the raw dataset prior to release. In some cases, only removing identifier attributes may not be sufficient, such as NetFlix dataset.
- **Quasi-identifier data**
  Quasi-identifier attributes refer to those attributes in node profile data which can be used in combination to uniquely identify a profile, though used in separation causes no disclosure danger. Examples of quasi-identifiers are birthdate, residence zipcode or city, gender of users. It is well known that by combining birthday, sex and zipcode, one can identify large population of people living in United state without knowing their identifier data [16]. For social network data, structural features of a node, such as degree and neighborhood sub-graph, can be used as quasi-identifiers to uniquely identify a node. This can lead to the privacy risk of linking a user ID with the sensitive information contained in the published SN dataset when certain background knowledge is made publically available. Quasi-identifier data often contain good utility of released data for both safe sharing and retention.

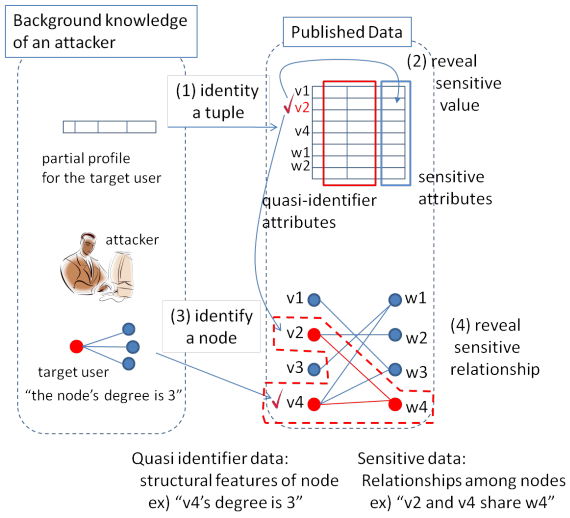

Fig. 1. Quasi-identifying data and Sensitive data in Social Network

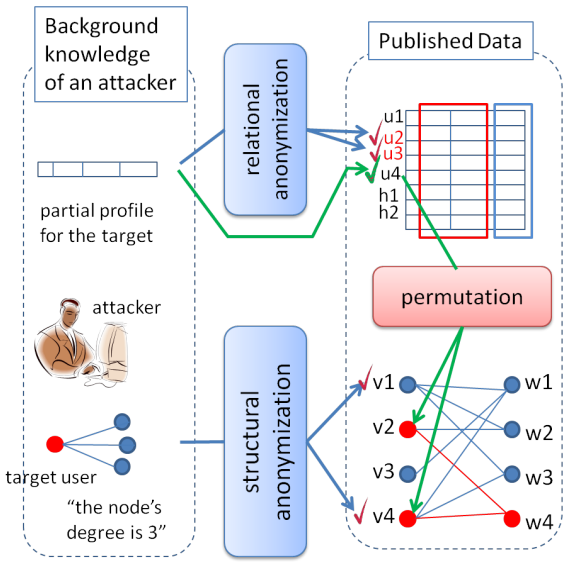

Fig. 2. Categories of Publishing Approaches

- **Sensitive data**
  Sensitive data in social network context refer to those user attributes in node profiles that are considered private and have controlled access, such as specific activities or where-about of a user. In addition, certain relationships between nodes can be sensitive data. For example, an activity that Alice and Bob had a date on 8/31/2011 in Figo can be sensitive information for both of them. Thus, publishing this activity simply by replacing user IDs of Alice and Bob with randomly generated numbers can be risky due to possible inference attack on the subgraph of two user nodes connected to the same group node which represents a dating activity on 8/31/2011 in Figo. Furthermore, the participation of certain event or specific interest group of a user can be highly sensitive to some users but not considered sensitive to others. Thus, data sensitivity is a very personal matter and may differ from user to user, event to event, location to location, and time to time. Sensitive attributes of entities, on one hand, are the part of data with highest utility and on the other hand, present the sensitive associations that we want to hide or prevent identity linking attacks.

Figure 1 shows the intimate relations among the three types of attributes for social network datasets and how an attacker reveals sensitive data of some target users. Basically, with certain background knowledge acquired by an attacker, such as partial profile of a target user or partial structure/relationship pattern, the attacker may identify the profile of the target user with high probability (Figure 1 (1)), then the attacker can infer sensitive attribute values of the target user in the corresponding profile (Figure 1 (2)). In addition, the attacker can also reveal the relationship of the target user in the graph structure (Figure 1 (3)), and reveal sensitive relationship of the target user (Figure 1 (4)).

B. Anonymization by Perturbation or Permutation

In order to prevent unauthorized disclosure of sensitive information from published social network data, many data anonymization techniques are proposed. We can classify existing anonymization techniques into two broad categories: perturbation based approaches and permutation based approaches. The former applies generalization and suppression techniques [16] to quasi-identifier attributes in node profiles and thus we refer to this category of techniques as relational anonymization. The latter applies permutation to graph structure and user-group relations and thus we refer to this category of techniques as structural anonymization. Figure 2 shows these two categories of anonymization techniques and how the sensitive data is protected.

Relational Anaonymization. Social network data are represented as relational tables. Most relational anaonymization methods typically use k-anonymity[16] as safety condition to group the users into k-anonymized groups by generalizing quasi-identifier attributes while preserving sensitive attributes (utility). For instance, zipcode of five digits such as 30329 and 30310 will be generalized into 30*** in order to group k users with the same quasi-identifiers into a k-anonymized group. Unfortunately, k-anonymity is not sufficient as a privacy protection measure when all k users in one k-anonymized group share the same sensitive value (participated in the same sensitive activity). Thus l-diversity[13] is proposed to enhance the privacy strength of k-anonymity. Unfortunately, even with l diverse sensitive values, one may still leak sensitive information when all l diverse values are referring to the same fact (e.g., different glucose values higher than 180 all indicate the same disease – Diabetes regardless of their value diversity). Several new metrics are proposed to further enhance k-anonymity and l-diversity, including proximity privacy [14], t-closeness [11] and so forth. Relational anonymizaiton tech-

niques remain to be unsafe in the presence of background knowledge of attackers.

Structural Ananonymization.
Structural anonymization prevents an attacker from uniquely identifying structure or profile data of a target user by utilizing structural features of users in the graph. This type of anonymization techniques modify edges/nodes or add fake edges/nodes, aiming at making it harder for any attacker to link some nodes and structures to a target user by inference attacks. For example, K-candidate anonymity [6] defines that 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 likelifood of every candidate node in this set is relevant to Q is less than or equal to $\frac{1}{K}$. Other similar approaches include K-automorphism anonymity [22], general definition of structural anonymization [7], edge modification [12], and Randomization [18].

Permutation techniques prevent from revealing sensitive values by breaking links between a node in the graph and its corresponding profile and structure. It is based on the table data permutation approach [17], which breaks the linkage between quasi-identifier attributes and the sensitive attributes. Cormode et al [4] applied and extended the table permutation approach to graph data. However, the table permutation based technique suffers from loss of utility in terms of link based analysis. Bhagat et al [1] improved the Cormode's approach for anonymizing the social network data by improving the link utility with (k,l)-grouping safe condition. The (k,l)- grouping of bipartite graph $G(V,W,E)$ uses an edge augmentation based approach to partition V (W) into overlapping subsets of size = k (l) and the publish edges E' is isomorphic to E, where mapping from E to E' is anonymized based on augmented partitions of V, W such that spurious semantic associations between user node and group node are added to satisfy (k,l) grouping. Safe (k,l) groupings ensure that nodes in same group of V have no common neighbors in W, which is essentially a diversity condition to ensure node and edge sparsity in the bipartite graph. [1] claims that safe (k,l) groupings are safe against static attacks and safe against attackers who know a limited number of edges. Figure 3 shows the permutated social network data by using (k,l)-grouping permutation [1] ($k=2, l=2$). The bipartite graph consists of seven (k,l)-grouped user nodes and six activity group nodes. k and l denote the minimum sizes of (k,l) groups for user nodes V and group nodes W respectively ($k>1$, $l\geq 1$). By (k,l) partitioning of V and W, all the user-group links are preserved but the user node and group node in each original user-group link is now permuted with a set of at least k user nodes and a set of at least l group nodes. The main idea of (k,l)-grouping is to transform each user-group link in the original social network graph into a permuted (k,l) link that links a set of at least k users to a set of at least l groups in the anonymized bipartite graph. Thus, every user is not uniquely associated with a group since each user node can only link to a group with at least k other users present, though among the k users, at least one user is the true user node associated to the

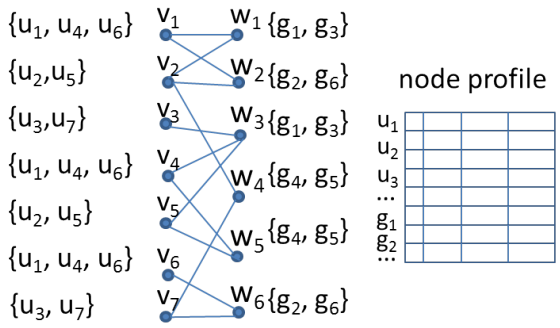

Fig. 3. Applying permutation technique for user-group affiliation network

give group node. By applying the *safe-grouping property* such that any two nodes in the same group of V have no common neighbors in W, the (k,l)-grouping based graph permutation technique outlined in [1] aims at ensuring that an attacker can guess the existence of an edge (u,g) in the original social network graph G_0 with the probability at most $1/max(k,l)$ based on any given (k,l)-grouping transformed link (v,w) in G_1 where $u\in v$ and $g\in w$.

The graph permutation by (k,l)-grouping safe condition is the most representative graph permutation technique for privacy-preserving publishing of social network data in the recent years as it preserves the statistical semantics of user-group links. Unfortunately, we argue that such permutation approach may not always deliver the safe condition that no attackers can infer from the (k,l) anonymized SN graph the existence of an edge (v,w) in the original social network graph with the probability at most $1/max(k,l)$. In the subsequent sections, we will describe the vulnerabilities of social network data publishing using graph permutation like (k,l) grouping and cauterize the privacy risks using threat models.

V. Vulnerabilities and Risk Analysis

One way to view anonymization is to consider it as an obfuscation method of adding uncertainty to certain data such that an attacker cannot be sure about the presence or the associations of released data to some known identity. A key quality measure of privacy is the level of confidence that all possible interpretations of released data have equal probability of unauthorized disclosure under association/linking attacks. A key quality measure of utility is the range of services that can be performed over the released data with high confidence and accuracy.

In this section we describe the vulnerabilities found in existing social network data anonymizaiton mechanisms and provide better understanding of privacy risks involved in publishing social network data as well as the importance of realistic assumptions in designing effective countermeasures. We first describe the problem statement through an illustrative example. Then we introduce the reference model and the types of background knowledge we use to model and quantify uncertainty of an attacker and to understand the impact of

different background knowledge. Finally we describe two types of inference attacks that an adversary may use to launch an inference attack, which can help an adversary to effectively reduce the space of possible worlds that may match to the real world of social network data and uncover sensitive information of targeted users with high confidence.

A. Motivating Example and Problem Statement

In this section we first describe an common problem shared by existing models for social network data publishing, as outlined in the previous section. All these anonymization approaches assume that user identity attributes are removed and replaced with pseudo identifiers. Most of them also assume that quasi-identifier attributes and sensitive attributes are preserved but a selection of quasi-identifier attributes may be anonymized through data perturbation or data permutation processing prior to release in order to meet the safety conditions such as those described in Section III. For presentation convenience, we will use the (k, l)-grouping permutation approach as a representative example technique to illustrate the problem and the possible inference attacks.

Recall the example in Figure 3 where a (k, l)-grouping based anonymization is performed with $k = 2$ and $l = 2$. In the published user-group links related to group vertex w_1, two anonymized user vertices v_1 and v_2 are associated with w_1, where v_1 corresponds to three original user nodes: u_1, u_4, u_6 and v_2 corresponds to two original user nodes: u_2, u_5 respectively. Also the anonymized group node w_1 corresponds to two distinct activity group nodes: g_1 and g_3 in the original SN graph. This anonymized graph shows that the anonymized user-group link (v_1, w_1) can be matched to only one of the six possible worlds in the original SN graph: (u_1, g_1), (u_4, g_1), (u_6, g_1), (u_1, g_3), (u_4, g_3), (u_6, g_3). Similarly (v_2, w_1) has six possible worlds as well. Thus, there are a total of 12 possible worlds for the subgraph of (v_1, w_1) and (v_2, w_1). Figure 4 shows six possible worlds or interpretations of the piece of published data related to activity group g_1 in w_1, denoted by $pw_1, ..., pw_6$. If the exposure level is 1, then the utility of the anonymized SN data is very limited and only queries of type 0 can be served due to the fact that only graph structure is released after permutation and all profile information are removed in the released dataset. Not surprisingly, we can be assured that no attackers can infer from the (k, l) anonymized SN graph the existence of an association of u with g (i.e., a user-group link (u, g)) with the probability higher than $1/max(k, l)$. It is obvious that restricting the data sharing and publishing to only graph structure with no node profiles is neither realistic nor meaningful for data sharing and data retention purpose.

On the other hand, when the exposure level is 2 or 3, we can gain much higher utility with (k, l) anonymized SN graph and enjoy all four types of queries with high accuracy. This is because all structural links are preserved although some spurious associations are added as the results of structural permutation. Also all quasi-identifier attributes and sensitive attributes of nodes in the original SN graph are available.

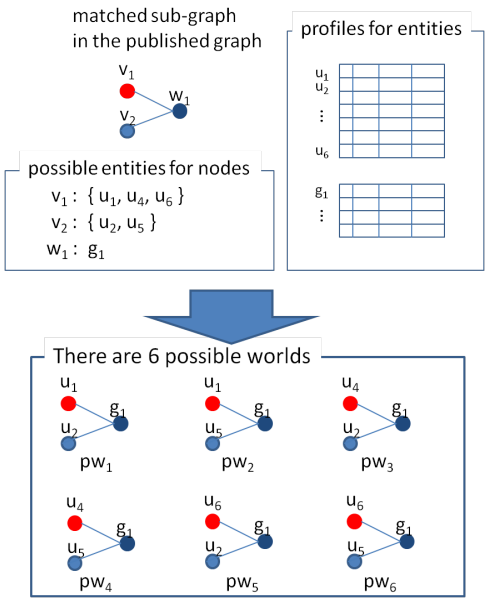

Fig. 4. Six Possible Worlds from a subgraph of the anonymized SN graph

However, we can no longer enjoy the safe condition guarantee claimed by [4], [1]: attackers now can easily infer from the (k, l) anonymized SN graph the existence of an association of u with g (i.e., a user-group link (u, g) in the original social network graph) with the probability higher than $1/max(k, l)$.

Concretely, in Figure 4, user profiles u_1, u_4 and u_6 are attached to v_1 by (k, l)-anonymization algorithm. By utilizing common sense background knowledge, one can dramatically reduce the search space for matching possible worlds. For instance, if g_1 refers to the research meeting of database laboratory in a university and g_3 refers to a swim meet of Atlanta teens, u_1 is a Ph.D student, u_4 is a lawyer and u_6 is an Italian chef. The attacker can utilize the profile data as such and common sense knowledge to infer with high confidence that g_3 has close to zero probability to be the true group with which u_1, u_4 or u_6 is associated. Also in comparison, certain interpretations such as the lawyer or Italian chef attended a meeting on Database research in a university have close to zero probability to be true, and can easily be eliminated in the attack analysis, thus defeating the safe condition and the claimed privacy protection of (k, l)-grouping anonymization algorithms. Thus, the adversary can infer that u_1 has the highest probability of true entity among the three possible entities to associate with g_1. Thus all six possible worlds involving g_3 can be eliminated. Similarly, pw_3, pw_4, pw_5, pw_6 have close to zero probability, much lower than $1/max(k, l)$, to be the true world. By using such inference attacks, an adversary can guess with high confidence that u_1 is associated to g_1 in the original SN graph. This example also illustrates the fact that the probability of matching a possible world to the true world is not the same for all possible worlds when the exposure level of anonymized dataset is higher than 1 or when anonymized dataset contains node profile information

in addition to graph structure. An adversary can easily launch inference attacks by analyzing and comparing the user profile data using his background knowledge, such as common sense, domain knowledge and so forth.

By carefully examining the problems illustrated above, we can see that this type of vulnerabilities is introduced primarily due to the lack of careful consideration in the design of graph permutation methods, especially when adding uncertainty into certain data. For example, the (k, l)-grouping based anonymization approach fails to incorporate possible background knowledge of an attacker when designing the method of adding uncertainty into the original SN graph during (k, l) permutation of user nodes and group nodes.

B. Reference Model of Uncertainty in Data Publishing

Uncertainty in anonymized data can be modeled typically in terms of multiple possible worlds or interpretations of original data. In the context of social network (SN) data, each possible world corresponds to a SN graph. The uncertainty model may attach a probability to each world and queries conceptually range over all possible worlds. We distinguish possible interpretations from probabilistic interpretations. The former define an interpretation or a fact possible if a possible world W where it is true exists. The later defines the probability of an interpretation or a fact being true.

Let $G = (V, W, E)$ denote a social network graph and $G' = (V', W', E')$ denote the anonymized graph of G using (k, l)-grouping permutation, where V' is a set of anonymized user nodes and $v \in V'$ is a subset of V, denoted by $\{u_1, ..., u_k | u_i \in V, 1 \leq i \leq k\}$, and W' is a set of anonymized group nodes and $w \in W'$ is a subset of W, denoted by $\{g_1, ..., g_l | g_j \in W, 1 \leq j \leq l\}$. We call u_i and g_j possible entities of v and w respectively. E' is isomorphic to E. For any user-group link $(u_i, g_j) \in E$ of original SN graph G, we have a corresponding edge (v, w) in E' such that $u_i \in v$ and $g_j \in w$. Each user node and group node in $G = (V, W, E)$ is associated with a profile that describes the entity by a set of attributes. Although the identity attributes are removed as initial step of anonymization, quasi-identifier attributes and sensitive attributes are kept after anonymization. Typical information included in a node profile ranges from (i) spatial information such as birth-place, home address, office address and so forth, (ii) temporal information such as starting and ending of an activity, birthdate, educational histories and career histories, (iii) categorical information such as age, gender, affiliation, title.

Background Knowledge.

Background knowledge refers to information that is essential to understanding a situation or problem. In our context, the background knowledge refers to the common sense and the domain knowledge that an adversary may use to launch inference attacks over published social network datasets. Recall the example in Figure 3, the adversary breaks the privacy of anonymized SN data by utilizing his background knowledge, such as a PhD student is more likely than an Italian chef or a civil lawyer to participate in a research meeting held in a Database lab at a university and teens refer to children between age 13 and age 18. Background knowledge can be categorized into common sense knowledge and domain knowledge.

- Common Sense Knowledge
  Attackers can leverage the common-sense knowledge from their daily life about the relations between users and activity groups. For example, men rarely belong to a knitting club. Thus, if an anonymized user node containing k users is linked to a knitting activity group, then only females are likely to be the true entity that joins a knitting club. Similarly, we consider all the facts such as "students who commute on foot to school must live in the same city where his school is located " are also characterized as common-sense type of background knowledge.
- Domain Knowledge
  Domain knowledge refers to the concepts or abstractions or facts that domain experts tend to use in their reasoning and analysis. Attackers can leverage the domain knowledge of entities and groups to infer which user nodes in the anonymized set are totally irrelevant. Recall the anonymized user-group edge (v_1, w_1) in Figure 4 where $w_1 = \{g_1, g_3\}$ and $v_1 = \{u_1, u_4, u_6\}$. Given that g_3 refers to a teen swim meet, and u_1, u_4, u_6 are PhD student, civil lawyer, Italian chef, thus an attacker can easily combine the common sense knowledge and domain knowledge to infer that all possible worlds involving g_3 should be eliminated.

We argue that any attack-resilient social network publishing mechanism should take into account the possible background knowledge an adversary may have about the user nodes and activity group nodes when designing data anonymizaiton methods by inserting uncertainty into certain data.

C. Attack Model

In this section we discuss two types of background knowledge attacks: user-group constraint attack and skewed distribution attack. Both types of attacks utilize background knowledge about user nodes and group nodes in the published SN graph to eliminate the number of possible worlds that are clearly irrelevant.

- User-group constraint violation attack
  An adversary makes use of his background knowledge to define a set of constraints between user nodes and group nodes and between user nodes that participate in the same group activity. Such graph structure based constraints can be a powerful tool for launching inference attacks. For example, a woman who is less than 15 years old cannot get marriage in Japan is a well known custom constraint on marriage relationship. Vegetarian does not eat meat is another common-sense constraint on user-group such that it is unlikely for a vegetarian to join a French cooking club. An adversary can utilize this type of user-group constraint violation attacks to identify and eliminate those possible worlds that clearly do not make sense or impossible to be true.

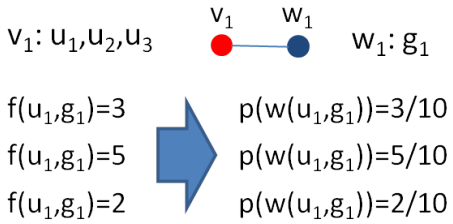

Fig. 5. Skewed probability distribution attack and an example scoring function

- Skew Probability Distribution attack
  This attack deals with the situations in which an adversary may not be able to determine with high confidence which possible worlds to eliminate. Often in such situations, if an adversary uncovers the skewed probability distribution over the set of possible worlds for an anonymized SN graph, the adversary may leverage the skewed distribution of the probabilities to launch a successful inference attack. For example, an adversary may define a scoring function $f(u, g)$ between a possible user node u and a possible activity group node g based on his background knowledge. This function calculates the probability of this association to be true in the original SN graph. Figure 5 shows an example of such scoring function. We have an anonymized user-group association (v_1, w_1) where $v_1 = \{u_1, u_2, u_3\}$ and $w_1 = \{g_1\}$. Thus we have 3 possible worlds: (u_1, g_1), (u_2, g_1), (u_3, g_1). Assume that the adversary uses his background knowledge to obtain the scoring function and the scores for (u_1, g_1), (u_2, g_1) and (u_3, g_1) are 3, 5 and 2 respectively. We can easily compute the probability for each possible world as shown in Figure 5. By using value of function, the attacker can infer the probability of each possible world.

D. Attack Analysis

In this section, we describe the detail definitions of User-Group Constraint Violation Attack and Probability Skew Attack.

Running Example

Figure 6 and Figure 7 shows the anonymized graph data and user profile and group profile data. This profile data includesthe log of events such as who attended what event. Group-nodes represent events and an edge between an user-node v_i and a group-node w_j represents user v_i attended the event w_j.

The graph in Figure 6 is anonymized by using (k, l)-grouping permutation with $k = 2$ and $l = 1$ [4]. Also all identity attributes are removed before anonymization is performed. In Figure 6, v_1 connected w_1, v_1 maps to a list of user ids $\{u_1, u_4, u_6\}$ and w_3 maps to group id g_1. g_1 denotes the event of sharing email between v_1 and v_2 at 18:23 on 2011/06/23.

We assume that an attacker wants to know who attends

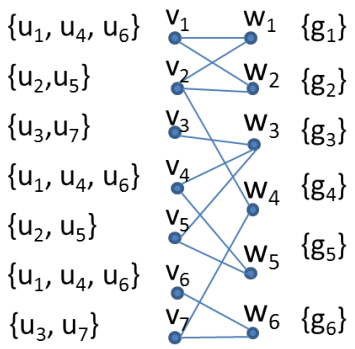

Fig. 6. Graph Structure of an example dataset

Uid	Age	Sex	Job	City	Country
u_1	29	M	Florist	Ibaraki	JP
u_2	20	M	Architect	Tokyo	JP
u_3	24	F	Ph.D stud.	Atlanta	US
u_4	31	M	Ph.D stud.	Atlanta	US
u_5	44	M	Prof.	Atlanta	US
u_6	34	F	Lower	Munich	DE
u_7	44	M	Architect	Munich	DE

Eid	Event	Startdate	Starttime	Enddate	Endtime
g_1	email	20110603	18:23	20110603	
g_2	meeting	20110610	11:00 EDT	20110610	12:30 EDT
g_3	meeting	20110506	14:00 EDT	20110506	15:30 EDT
g_4	email	20110530	21:04	20110530	
g_5	game	20110702	0:12	20110702	2:57
g_6	chat	20110613	14:00 EDT	20110613	15:45 EDT

Fig. 7. Profile Data of the example dataset in Figure 6

the meeting at 14:00 EDT on 2011/05/06, namely which user nodes have true association to g_3. From the anonymized graph, the attacker can obtain the information that the target event is g_3 and v_3,v_4,v_5 are connected to g_3. Then the attacker tries to find the profiles for v_3,v_4 and v_5 by using two kinds of attacks defined in the previous section.

Let G=(V,W,E) denote a social network graph and $G' = (V', W', E')$ denote an (k, l) anonymized graph of G as defined in Section III-B. Let $PW(G, G')$ denote a set of possible worlds of G' . In this example, the attacker can find 12 possible worlds from the sub-graph with three user-nodes v_3,v_4,v_5 and one group-node w_3 as shown in Figure 6. Given a possible world pw_i of the anonymized graph $G' = (V', W', E')$ where $V' = \{v_3, v_4, v_5\}$, $W' = \{g_3\}$ and $E' = \{(v_3, g_3), (v_4, g_3), (v_5, g_3)\}$, a mapping of this possible world to the real world in G (the original SN graph), denoted by M_i, is defined as $M_i = \{(v_3, u_3), (v_4, u_1), (v_5, u_2), (w_3, g_3)\}$. For presentation convenience, we describe a possible world with mapping $M_i = \{(v_1, u_1), (v_2, u_2), (v_3, u_3), (w_3, g_3)\}$ as $pw(u_1, u_2, u_3, g_3)$ when no confusion occurs. By this expres-

sion, the 12 possible worlds are described as follows;

$$PW(G,G') = \{pw(u_3,u_1,u_2,g_3), pw(u_3,u_1,u_5,g_3), pw(u_3,u_4,u_2,g_3), pw(u_3,u_4,u_5,g_3), pw(u_3,u_6,u_2,g_3), pw(u_3,u_6,u_5,g_3), pw(u_7,u_1,u_2,g_3), pw(u_7,u_1,u_5,g_3), pw(u_7,u_4,u_2,g_3), pw(u_7,u_4,u_5,g_3), pw(u_7,u_6,u_2,g_3), pw(u_7,u_6,u_5,g_3)\}$$

User-Group Constraint Violation Attack

An adversary may exercise the user-group constraint violation attack to identify and eliminate those possible worlds that are clearly false. Most importantly, the adversary will select the right set of background knowledge in order to isolate those possible worlds that have low probability from those high probability ones. For example, event g_3 refers to a meeting which started 14:00 EDT. By time difference between Atlanta and Japan, we know that 14:00EDT is 3:00 JST in Japan. Thus the adversary can introduce a time-difference constraint between user and group such that for any activity group that has short time window associated with, any user whose time zone is 12 hour difference will not be possible to be associated with this group. Using this constraint, we can easily detect that (u_1, g_3) and (u_2, g_3) are violating this constraint since u_1, u_2 has Japan as its current residence country in their profiles, and thus are very difficult if not impossible for u_1, u_2 to attend this meeting. Based on this analysis, the adversary can easily identify those user-group relationships which violate the constraint. In this example, (u_1, g_3) and (u_2, g_3) violate the constraint $meetingtime\_const$, thus the attacker can eliminate those possible worlds which include these pairs of nodes. After removing the inappropriate possible worlds, there remain 4 possible worlds, and they are shown as follows:

$$PW(G,G') = \{pw(u_3,u_4,u_5,g_3), pw(u_3,u_6,u_5,g_3), pw(u_7,u_4,u_5,g_3), pw(u_7,u_6,u_5,g_3)\}$$

By eliminating those obvious false possible worlds, the attacker can detect that (u_5, g_3) has higher probability to be the true user-group link.

Probability Skew Attack

Anonymization introduces uncertainty into certain data. A desired property of anonymization is to ensure that all possible worlds have equal or very similar probability to be the true world. However, by exposing information in level 2 and 3, such ideal condition is no longer valid because different possible worlds may have different probabilities for being the true world.

In this attack, an adversary tries to select a world which is the closest to the true world based on his background knowledge. Concretely, one approach to conducting such inference is to define a score function $f(pw(G,G')$ that can produce a ranking of possible worlds in terms of how closely each matches with the true world. Such scoring function should take into account of as much background knowledge as possible to improve the attack-resilience of the published SN graph.

For example, g_3 in Figure 6 refers tos a meeting, then an attacker may use his background knowledge to assume that people who attend the meeting have the same or similar professional profile with each other. Based on this assumption, the attacker define a score function so that the possible world that closely matches with one another will have higher probability to be mapped to the true world.

An example score function can be introduced to define the background knowledge. For example, the adversary can select a set U of attributes that are representative of users' professional attributes. For each selected attribute, we count the maximun number of people who have the same value and we regard the max value as the similarity of the attribute sim_{attr}. For example, in a possible world $pw(u_3,u_6,u_5,g_3)$, u_3 and u_5 have Atlanta as their residence city, but u_6's city is "Munich" then sim_{city} is 2. We define the score function $f(pw(G,G'))$ for each possible world by calculating the sum of the values for all attributes as follows;

$$f(pw(G,G'),U)) = \sum_{a \in ATTR(U)} sim_a$$

The score for $pw(u_3,u_6,u_5,g_3)$ is calculated as follows;

$$\begin{aligned} f(pw(u_3,u_6,u_5,g_3)) &= sim_{age} + sim_{job} + sim_{city} \\ &\quad + sim_{country} \\ &= 1+2+1+2+2 \\ &= 8 \end{aligned}$$

Scores of all other possible worlds are as follows;

$$\begin{aligned} f(pw(u_3,u_4,u_5,g_3)) &= 1+2+2+3+3 = 11 \\ f(pw(u_7,u_4,u_5,g_3)) &= 2+3+1+2+2 = 10 \\ f(pw(u_7,u_6,u_5,g_3)) &= 2+2+1+2+2 = 9 \end{aligned}$$

Based on the scoring function and the results, the attacker identifies the possible world with the highest similarity score as the most probable matching to the true world. From the above example, given that $pw(u_3,u_4,u_5)$ has the highest similarity score of $11/(11+10+9+8) = 11/38$, thus it is identified by the attacker as most likely the true world.

Observation 1: When an attacker calculates the possibility of the true entity based on the scoring function $f(pw(G,G'),U)$, the highest possibility calculated by the following formula

$$\frac{max_{pw \in PW(G,G')} f(pw)}{\sum_{pw \in PW(G,G')} f(pw)}$$

is greater than $\frac{1}{|PW(G,G')|}$.

VI. Discussions and Countermeasures

We have described the privacy risks in publishing anonymized social network data and two types of background knowledge based attacks: constraint violation attack and probability skew attack. One of the fundamental vulnerabilities

in the design of graph permutation techniques is the lack of consideration of background knowledge and the risks of combining background knowledge with published profile data and graph structure data. Concretely, take (k, l) grouping permutation approach as an example, the algorithm for composing k user groups and l activity groups from input social network $G = (V, W, E)$ focuses on meeting the safe condition that nodes in same group of V have no common neighbors in W, which is a condition for higher utility but it does not guarantee background knowledge attack resilience. A straightforward approach to the (k, l) grouping algorithm is to revise the (k, l) grouping algorithm in terms of how to add uncertainty through inserting spurious user-group links. For example, the (k, l)-grouping algorithm first sorts the nodes by groups in the sense that user nodes that connect to the same group node are queued together. To simplify the discussion, let us set $l = 1$. Then we can initially consider each group as a class c and for each node in the sorted list, it checks whether the node and each class c satisfy the safety condition and if yes, this node is added into class c. Obviously, we can revise the sorting condition. Instead of sorting nodes according to groups, we sort nodes in terms of both groups and attribute similarity. Also we revise the safety condition such that nodes in same group of V have no common neighbors in W and also their aggregated attribute similarity should be higher than a system-defined threshold to ensure user nodes that are too dissimilar should not be placed in the same cluster. The intuition behind the design of a new countermeasure is two folds: First, for those user nodes that are very similar with respect to the group nodes they are associated with, then putting them into one cluster will make the anonymized graph safer and more resilient to background knowledge attacks. Second, putting those user nodes who are more likely to join a specific group but have not yet as the most eligible candidates to be added into the group cluster. This will immediately increase the resilience to both user-group constraint violation attack and probability skew attack.

Another countermeasure that is potentially interesting is to combine k-anonymity based algorithm with (k, l)-grouping permutation. For instance, we can apply k-anonymization over the set of nodes to construct k-anonymized groups. Such group can then be tuned and revised to obey the safety conditions. Due to space limit, we omit these issues for now.

VII. Related Works

A fair amount of work has been dedicated to privacy preserving social network data publishing and mining. Most of the techniques focus on graph data anonymizaiton using techniques from data privacy research. Concretely, Sweeny et. al proposed $k-$anonymity[16] as privacy measure for anonymizing table data. Subsequent anonymity measures are proposed such as l-diversity [13], $t-$closeness[11], proximity privacy [**?**]. Most of these anonymity measures improve the anonymity strength in terms of protection of sensitive attributes. Furthermore, data anonymizaiton implementation methods have been developed [8][9] to provide optimal or near optimal efficiency. Although these studies are popular in tabular data anaonymization, their applications to anonymizing social network data have been somewhat limited, especially in terms of utility and personalization.

Research on anaonymizing social network data has been primarily focused on the graph structure. Liu et al.[12] proposed anonymization methods which modify edges and nodes so that there are at least k nodes with the same degree. Zhou et al.[21] proposed k-neighborhood anonymity which can modify original graph structure so that every node has at least k other nodes whose 1.5 hop neighborhood subgraph have the same topology. Hay et al.[7] generalizes the multi-hop $k-$neighbourhood anonymity which can apply to multi-hop neighborhood subgraphs. These anonymization techniques are categorized as structural anonymization. Most of these structural anonymization methods tend to , ignore the existence of profile data for simplifying their privacy problem. k-candidate anonymity[6], k-automorphism[22], k-isomorphism[3] can be categorized in the same category.

In addition to anonymizing social network graph structure, there are some research efforts on anonymizing the attributes values of vertices in a social network graph. Zhou et al.[12] assumes that every node in social network has label data. It extracts 1.5-neighbourhood signatures as one of node attributes and it clusters these nodes. By using these clusters, it applies k-neighbourhood anonymity. Zheleva et al. [19] assumes that social network includes multi-kinds of public edges and sensitive edges and they propose anonymization method for protecting sensitive edges. This method performs two-steps anonymization andis categorized as structural/relational anonymization approach which is described in SectionIV-B. Campan and Truta[2] also assume that social network has profile data. It is also categorized structural/relational anonymization approach. It introduce measurements for structural information loss and data information loss, and users can adjust utilities of graph structure and its profile. However these works do not consider the attack using semantics of user nodes and group node which are proposed in this paper.

VIII. Conclusion

We present an analysis of privacy risks in anonymized social network data. We show that on one hand, social network publishing needs to be anonymized while preserving utility, and on the other hand, privacy protection in social network data publishing is also an arm race problem. We show that an attacker can leverages semantics of profile data and background knowledge related to the published data for narrowing down the options among the candidate answers the attacker can infer with high confidence. The main contributions of this studyA can be summarized into three highlights. First, to the best of our knowledge, we make the first attempt to define the utility of released data in terms of exposure levels and query types, assuming queries are the most fundamental operations in social network analysis. We argue that using information exposure levels to characterize the utility of anonymized data can be used as a general and usage-neutral metric and query types can be used as the baseline usage driven utility metric.

Second, we identify two types of background knowledge based inference attacks that can break some of most representative graph permutation based anonymization techniques in terms of anonymity violations. Third but not the least, we describe some design considerations for developing countermeasures in privacy preserving social network data publishing.

Acknowledgement

The third author is partially sponsored by grants from NSF CISE NetSE program, CrossCutting program, CyberTrust program and grants from industry such as IBM SUR grant, IBM faculty award, and Intel ISTC grant.

REFERENCES

[1] S. Bhagat, G. Cormode, B. Krishnamurthy and D. Srivastava: Class-based graph anonymization for social network data, In Proc. of the VLDB Endowment, Vol.2, Issue 1, pp.766-777, 2009.

[2] A. Campan and T.M.Truta: A Clustering Approach for Data and Structural Anonymity in Social Networks, PinKDD, 2008.

[3] J.Cheng, A.W.Fu and J. Liu: K-isomorphism: privacy preserving network publication against structural attacks, Proceedings of the 2010 international conference on Management of data (SIGMOD'10), pp.459–470, 2010.

[4] G. Cormode, D. Sirvastava, T. Yu and Q. Zhang: Anonymizing Bipartite Graph Data using Safe Groupings, In Proc. of the VLDB Endowment, Vol.1, Issue 1, pp.833-844, 2008.

[5] G. Flake, S. Lawrence, C.L.Giles and F. Coetzee: Self-organization and identification of web communities, SIGCOMM, pp.251-262, 2002.

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

[7] M. Hay, G. Miklau, D. Jensen, D. Towsley and C. Li: Resisting structural re-identification in anonymized social networks, The VLDB Journal, Vol.2010, No.19, pp.797 – 823, 2010.

[8] K. LeFevre, D.J. DeWitt and R. Ramakrithnan: Incognito: Efficient Full-Domain K-Anonymity, In Proceedings of the 2005 ACM SIGMOD international conference on Management of data(SIGMOD2005), 2005.

[9] K. LeFevre, D.J. DeWitt and R. Ramakrithnan: Mondrian: Multidimensional k-Anonymity, In Proceedings of ICDE 2006.

[10] J. Leskovec, J. Kleinberg, and C. Faloutsos: Graph over time: densification laws, shrinking diameters and possible explanations, In Proc. of ACM SIGKDD, pp.177 – 187, 2005.

[11] N.Li, T.Li, and S. Venkatasubramaniam: t-closeness: Privacy beyond k-anonymity and l-diversity, In IEEE ICDE, 2007.

[12] K.Liu and E. Terzi: Towards identity anonymization on graphs, Proceedings of the 2008 ACM SIGMOD international conference on Management of data(SIGMOD2008), pp.93–106, 2008.

[13] A. Machanavajjhara, J. Gehrke, D. Kifer and M. Venkitasubramaniam:l-diversity: Privacy beyond k-anonymity, In IEEE ICDE, 2006.

[14] Ting Wang and Ling Liu. "XColor: Protecting General Proximity Privacy", 26th IEEE International Conference on Data Engineering, March 1-6, 2010, Long Beach, California, USA.

[15] J.Neville and D. Jensen: Iterative Classification in Relational Data. In Workshop on Learning Statistical Models from Relational Data, AAAI, 2000.

[16] L. Sweeney: k-anonymity: A Model for Protecting Privacy, Uncertainty, Fuzziness and Knowledge-based Systems, Vol.10, No.5, pp.557–570, 2002.

[17] X. Xiao and Y. Tao: Anatomy: Privacy and Correlation Preserving Publication, Proceedings of the 32nd international conference on Very large data base(VLDB2006), pp.139-150, 2006.

[18] X. Ying and X. Wu: Randomizing social networks: a spectrum preserving approach, In Proceedings of SIAM International Conference on Data Mining, pp.60–71, 2008.

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

[20] E. Zheleva and L. Getoor: Privacy in Social Network: A Survey, In Social Network Data Analytics, Chapter 10, pp.277 – 306, 2011.

[21] B.Zhou and J. Pei: Preserving privacy in social networks against neighborhood attacks, In proceedings of the 24th International Conference on Data Engineering (ICDE), pp.506–515, 2008.

[22] L. Zou, L. Chen and M.T.Ozsu: K-automorphism: A general framework for privacy preserving network publication, In proceedings of the VLDB Endowment, Vol.2, Issue 1, pp.946–957, 2009.