THE NATIONAL UNIVERSITY
of SINGAPORE

School of Computing
Computing 1, 13 Computing Drive, Singapore 117417

TRA8/12

Discretionary Social Network Data Revelation with a User-Centric Utility Guarantee

Yi Song, Panagiotis Karras, Sadegh Nobari, Giorgos Cheliotis, Mingqiang Xue, and Stéphane Bressan

August 2012

T e c h n i c a l R e p o r t

Foreword

This technical report contains a research paper, development or tutorial article, which has been submitted for publication in a journal or for consideration by the commissioning organization. The report represents the ideas of its author, and should not be taken as the official views of the School or the University. Any discussion of the content of the report should be sent to the author, at the address shown on the cover.

OOI Beng Chin
Dean of School

Discretionary Social Network Data Revelation with a User-Centric Utility Guarantee

Yi Song[1] Panagiotis Karras[2] Sadegh Nobari[1]
Giorgos Cheliotis[3] Mingqiang Xue[1] Stéphane Bressan[1]
[1]School of Computing [3]Faculty of Arts and Social Sciences
National University of Singapore
{songyi, snobari, gcheliotis, xuemingq, steph}@nus.edu.sg
[2]Rutgers Business School
Rutgers University
karras@business.rutgers.edu

ABSTRACT

The proliferation of online social networks has created intense interest in studying the nature of such networks and revealing network information of interest to the end user. At the same time, the revelation of such data raises privacy concerns. Existing research addresses this problem following an approach popular in the database community: a model of data privacy is defined, and the data is rendered in a form that satisfies the constraints of that model while aiming to maximize some utility measure. Still, these is no consensus on what constitutes a clear and quantifiable utility measure over graph data. In this paper, we take a different approach: instead of starting out with a privacy objective, we define a *utility guarantee*, in terms of certain graph *connectivity* properties being preserved, that should be respected when releasing data, while otherwise distorting the graph to an extend desired for the sake of confidentiality. We propose a form of data release which builds on current practice in social network platforms: A user may want to see a subgraph of the whole network graph, in which that user as well as distant connections and affiliates participate. Such a snapshot should not allow malicious users to gain private information, yet provide useful information for benevolent users. We propose a mechanism to prepare data for user view under this setting. In an experimental study with real-world data, we demonstrate that our method preserves graph properties of interest (e.g., clustering coefficient, shortest path length, diameter, radius) more successfully than methods that randomly distort the graph to an *equal* extent, while it withstands structural attacks proposed in the literature.

1. INTRODUCTION

Online Social Network Sites (SNSs) allow users to discover and share information about themselves and their peers, while they provide researchers with a valuable tool for social, cultural, and media studies via data analysis and mining [22]. The capacity to exchange information in such networks rests on an assumed underlying trust among users [8]. While trust is thicker among people with strong interpersonal ties, it also affects one's ability to cultivate and mobilize weak social ties for the transfer of valuable information [18]. Trust is thus essential not only for *bonding* social capital, associated with strong ties, but also for *bridging* social capital, associated with weak social ties and information-seeking behavior [11]. SNSs are valuable for the development of both types of social capital, while the positive effects of their use may be stronger for bridging social capital [6]. In short, the technological affordances of SNSs provide leverage in building weak ties and bridging social capital, while the value of these ties for an individual is mediated by interpersonal trust [18].

In order to safeguard such trust, as well as institutional trust users place in the owners and administrators of the SNS, the privacy of users has to be guarded from malicious users, as well as from malicious data recipients when data is published to third parties. Still, the facility to ease the creation of social ties online is a central feature in any SNS [3]; such facility requires that *some* information about users is made available to both known others and to strangers. This tension between confidentiality and facility is especially pertinent in sites like LinkedIn or Xing, specializing in professional networking that eases the formation of weak ties.

Consequently, the need arises for a method that reveals network graph data in a *discretionary* manner, with the deterrence of malicious users in mind, while at the same time provides certain utility for benevolent users; thus a problem of *discretionary user-centric network data release* emerges. This problem is related to, but distinct from the problem of revealing whole-network data to third parties. We focus on the problem of revealing user data to end-users with the aim of helping them network better. The end-user derives utility from such data revelation, and may thus willingly choose to participate in such a scheme. We aim to guarantee such utility while releasing data in a discretionary manner.

Existing research in the area follows a *privacy-driven* paradigm: it formulates a certain *privacy principle*, and develops techniques that bring the network data to a form that abides thereby, while keeping the associated loss of utility low [1, 13, 19, 2, 5]. The transformed data is then ready to be released. However, the extent to which such techniques maintain the information utility of the network and structure thereof is vague. These studies suffice themselves to measuring ad hoc *utility metrics*; unfortunately, such metrics do not capture the extent to which an object as complex as a graph maintains its original properties. Nevertheless,

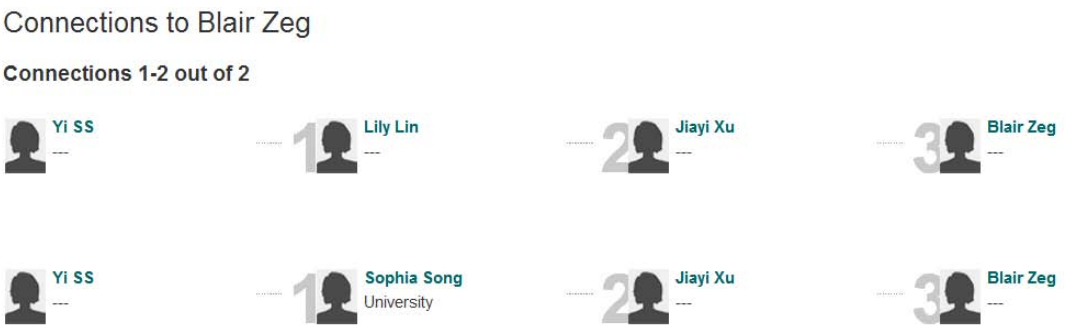


Figure 1: Visualization of connections in Xing

in the case where the information recipients are end-users of the social network site, aiming to utilize it for networking purposes, they would like to have a guarantee precisely on the *utility* of the released data, in terms of certain *graph properties* that may be valuable for networking, no less than they would desire a certain *privacy guarantee* about their own information being revealed to others.

A network is modeled as an undirected graph $G = (V, E)$, where V is a set of vertices (nodes) representing entities and E is a set of edges representing relations between entities. Nodes and edges may be annotated with attributes (e.g, occupation or interests for nodes, type or weight for edges), yet in this work we consider the most basic graph model.

A *naive anonymization* of G would substitute all entity identifiers in G by synthetic identifiers. However, such an anonymization does not suffice to conceal the identities behind the published graph, as the structural information in the network can itself serve to identify nodes [1, 29, 13, 19]. Thus, a *structural anonymization* is called for. Besides, a privacy threat is not posed by the identification of nodes in the network per se, but rather by the disclosure of the positions of such identified nodes with respect to each other. We contend that, when the data recipient is an end-user, a structural anonymization would suffice to provide the confidentiality users require, while other identifying information can still be published, as it may be valuable for purposes such as professional networking.

1.1 A Practical Example

We envisage a scenario in which an SNS user requests to see the network subgraph involving one's connections up to a certain number of hops. Such a subgraph would provide the user with an overview of her position in the broader network neighborhood of her contacts and their contacts. Thus, it could provide ideas as to whom she might be able to connect to next. To be truly useful, this subgraph should correctly reveal the identities of individuals within its scope and also provide some indication as to the relative positions of such individuals. However, for the sake of confidentiality, the subgraph should not reveal the *precise* relationships of such individuals among each other.

Currently, many SNS platforms, such as LinkedIn[1] and Xing,[2] provide a functionality by which users can see information about a path connecting them to other persons; in some cases, one can also see individuals along that path. This service offers valuable information to networkers. Yet, this practice poses problems, both from a privacy and a utility point of view: From a privacy perspective, the revelation of individuals along the path poses a risk to them, as the relationships among distant connections to the querying user may be sensitive and can be exploited by a malicious user. On the other hand, from a utility viewpoint, the published information is limited; a user may wish to view her position relative to a whole neighborhood, so as to identify nodes of interest; single paths do not provide such information.

Figure 1 shows a screen shot of the information provided by Xing in an example we have created using fictional names. Likewise, Figure 2 shows an example of the type of information provided by LinkedIn, again with fictional names. While the provided information indicates the existence of a connection, it is limited to a single path, and does not reveal other graph neighborhood information that may be of legitimate interest to the user.

Figure 2: Visualization of connections in LinkedIn

Noticeably, Xing shows all intermediate connections, and even provides names along a single path, in contrast to LinkedIn. If taken further, i.e., to longer paths, as it stands, this practice would arguably compromise the privacy of users involved. Nevertheless, inspired by this practice, we envisage that a user could ask for a presentation of a fuller view of the network's neighborhood structure around the presented path, or, more generally, for the presentation of any network subgraph of interest. Such a service should be *discretionary*, not revealing too much information about the network's microstructure that would compromise individual users' confidentiality, yet at the same time it should be informative.

1.2 The potential for structural attacks

Nevertheless, revealing a network's structural information can render users vulnerable to attacks. A malicious user may create a set of fake accounts and attempt to forge direct links between those accounts and to one or more targets, so as to directly elicit private information from them, or to create a unique structure that can be later identified in a revealed graph. This observation is the basis of the structural attack

[1]http://www.linkedin.com/
[2]http://www.xing.com/

introduced in [1]. We aim to design a utility-driven data revelation scheme that can foil such attacks.

In concrete terms, an attacker who naturally knows the identity of her targets could contact those targets directly and try to gain their trust. The chances of success at securing such targets' trust will increase if she can present herself as sharing a mutual friend, implying an endorsement of her request to connect to the target. When the path to the target is published, it becomes easier for the attacker to exploit such "friend-of-a-friend" trust. Platforms like LinkedIn and Xing appear to be vulnerable to such exploits as they publish partial or full path information. However, our approach will obstruct the attacker, as she will not know with certainty who is connected to whom. A guess at the exact chain that leads to her target will then be risky; if she mistakenly presents herself as a friend of a friend to any node in the chain, the chances of gaining that node's trust will be diminished. On the other hand, a benevolent networker, truthful about her intentions, will be able to solicit the assistance of users along the path to the target; as long as those users assess that she has a legitimate reason to reach her target, they will forward her request to the next hop.

1.3 Our proposal

Motivated by the above discussion, in this paper, we suggest a methodology for revealing social network data to relevant users following a *utility-driven* paradigm. By our scheme, network data are manipulated under certain constraints, aiming to preserve structural properties of the underlying graph, while otherwise distorting the graph's microstructure to the farthest extent allowed by those constraints. In this manner, the trade-off between data utility and data privacy is addressed in a novel manner, adhering to a utility guarantee. We define the structural constraints in terms of distance properties between pairs of nodes, and demonstrate that the resulting graphs can withstand attacks by adversaries possessing prior structural background knowledge, as suggested in [1]. Specifically, in the experimental section we measure the success rate for *any* attack based on the identification of an embedded subgraph in the distorted graphs, as a function of the amount of distortion incurred on it; as we discussed, such an embedded graph may consists of fake accounts created before graph releasing and connected among themselves and to other, victim nodes, so as to follow a unique and identifiable pattern.

In our approach, we publish a subgraph of the network graph, containing *nodes of interest* with respect to the querying user (possibly along with identifying information, depending on the application at hand). This subgraph is constructed so as to faithfully preserve the *reachability* information in the true subgraph: if a node is reachable from another node by a path of length lower than a threshold k, then it should also be similarly reachable in the released graph. However, the subgraph is otherwise distorted, so as to conceal exact node-to-node relationships, to the extent allowed by the reachability constraint. Thus, a querying user cannot confidently infer the potentially sensitive relationships among distant connections. Yet the same querying user obtains a wide view of her own and her peers' position in the overall network. Thereby, a benevolent user obtains valid information that is relevant in determining how to expand her network, while a malicious user is prevented from drawing accurate inferences about the relationships among people she is not closely related to, and is consequently deterred from attempting to utilize such information in order to gain their trust towards malicious ends (see also the discussion in section 3.4). We contend that such reachability-preserving graph transformation maintains crucial information with regard to graph structure that is valuable to the SNS user (as well as the a researcher or social network analyst), while distorting the graph in a way that renders it proof against structural attacks. Thus, the data release model we propose provides both higher utility and higher security than the naive path revelation model discussed in Section 1.1.

2. REACHABILITY PRESERVATION

Real-world social networks of certain size are usually *connected*; any two individuals in them are bound to be linked by a sufficiently large path. The *distance* between two individuals, i.e., the length of a shortest path connecting them, is usually rather small, not exceeding *six* steps. Milgram's small world experiment [20] suggested that social networks of people in the United States are characterized by such short distances, of approximately three friendship links, on average, without considering global linkages; Watts [24] recreated Milgram's experiment on the internet and found that the average number of intermediaries via which an e-mail message can be delivered to a target was around six; Leskovec and Horvitz [17] found the average distance among users of an instant messaging system to be 6.6; Goel et al. [10] tested the extent to which pairs of individuals in a large social network can actually *find* paths connecting them; they introduced a rigorous way of estimating true chain (i.e., *search distance*) lengths in a messaging network, and found that roughly half of all chains can be completed in 6-7 steps.

In view of this *connectedness* of real-world social networks, we deduce that no previously unknown information is disclosed when the mere *existence* of a path among two entities in a network is revealed. Thus, an objective of thwarting the inference of any linkage *whatsoever*, as in [5], would set an unnecessarily high goal and irretrievably alter the nature of the network. Besides, a bona fide SNS user can reasonably expect to be able to learn whether other individuals in the same network are *reachable* at up to a certain distance threshold and also gain a glimpse of the nature of the network that stands between them. Such information is vital to SNS users, e.g., job seekers in a professional network, newcomers in a city, or professionals looking for new partners. On the other hand, a *discretionary* revelation of such reachability information should *not* reveal the exact relationships among people in the exposed neighborhood, as malicious users can may take advantage thereof to launch attacks and gain access to potentially sensitive information.

As we discussed, professional networking platforms provide a function that concerns us: when users search for someone, they can see the path that leads from their node to the searched-for person, possibly under the condition that the path is not longer than 3 hops. Thus, Alice can see that the path $Alice \rightarrow Lara \rightarrow Olivia \rightarrow Bob$, connects her to Bob. An extension of this functionality to paths of arbitrary length would endanger users' confidentiality, as Alice would then acquire intimate knowledge about the relationships of people she is not acquainted with. Yet Alice has a legitimate interest to find out whether she is connected to a certain individual by a path longer than the ones she is already allowed to see, as well as to identify individuals in

her extended neighborhood and thereby possibly attempt to expand her social circle.

Motivated by such needs, we propose a *discretionary* graph publication model that provides useful connectivity and reachability information, along with other rich graph information, yet without correctly revealing the graph's microstructure concerning individuals lying along the presented connections. The connections shown in a graph published by our method are not necessarily true. Still, the published graph is constructed so that it *does* provides fairly correct reachability information. In effect, a bona fide user can use such information to explore the possibility of connecting to others and attempt such a connection by whatever means a given SNS platform provides. Still, a malicious user would not be able to exploit the presented network view without risking being exposed. In effect, graph reachability information is made available in a way that preserves certain properties of the underlying graph, while confining the potential that sensitive information is exposed.

Furthermore, by our proposal, users in the network can specify a distance threshold parameter d, so that they can quantify their own comfortable zone. Figure 3(a) depicts an example of a graph shown to user Alice, in which it is revealed that another user, Mike, is reachable within 4 hops. This happens *under the condition* that Mike has agreed to have the information about being reachable by 4 hops available to such other users; i.e., Mike has set his personal distance threshold to $d = 4$. Alice then gets the highlighted path information if she wants to see her position relative to Mike's position, even though this particular path may *not* be the *exact* path between Alice and Mike. Figure 3(b) shows what Alice would see in case Mike has not opted in to make his information available to users within 4 hops. To encourage users' participation, Alice's ability to view Mike's information can be made conditional on her making her own information available to users within 4 hops, i.e. her own personal distance threshold being at least 4.

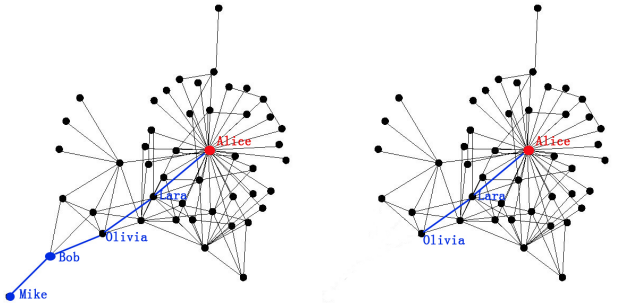


Figure 3: Example of path revelation.

Given such a facility, we expect that users will be willing to accept the discretionary revelation of their own presence in the network, as they stand to gain themselves in terms of increased networking functionality. Naturally, when releasing network data to third parties, we expect end-users to be primarily concerned with the protection of their confidentiality rather than with the utility of the released data. However, when network data is released among SNS end-users themselves, as in our primary motivating scenario, we expect that these end-users will have a stake in data utility and be willing to opt in such a scheme, as they will be among the beneficiaries of the information that will be provided. By setting a personalized exposure distance threshold d, users can tailor the tradeoff to their own needs and sensibilities. In the following discussion, it is always assumed that we are dealing with a set of users whose distance threshold permits their inclusion in the revealed graph.

2.1 Problem Definition

Let $G = (V, E)$ be a simple undirected graph that represents part of a social network; such a graph can consist of a network neighborhood of around a querying user's node. V is a set of vertices representing entities in the network, and E is a set of edges representing relations between entities. We start out by providing the following definition.

DEFINITION 1. *The* k-reachability graph *of* G, G^k, *is a graph having the same vertices* V *as* G, *such that an edge between two vertices exists in* G^k if and only if *the distance between them is at most* k.

For example, the 2-reachability graph of the graph G_1 at the left side of Figure 4 is the graph in the middle of the figure. If k is set to be the longest distance (i.e., the *diameter*) in G, then the k-reachability graph becomes trivially the same as the *transitive closure* of G. However, for intermediate values of k, G^k is rich in information, showing which entities in the network share connections of up to a certain length.

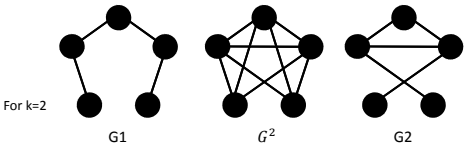


Figure 4: Graphs G_1 and G_2 having the same G^2

Our main claim is that, given a network neighborhood G and a certain k of interest, a graph G', having the same vertices, equal number of edges, and the same k-reachability graph G^k as G, while differing from G in as extensive a way as possible otherwise, provides high-utility information about G in a manner *discretionary* with respect to the confidential information of the users involved. We aim to devise a method that generates G' given G. We define the following problem:

PROBLEM 1. *Given a graph* $G(V, E)$ *and an integer* k, *produce a graph* $G'(E', V)$, *such that* $|E| = |E'|$ *and* $G^k = G'^k$, *while the difference between* G *and* G', *measured as the edit-distance of their edge sets,* $Dist(G, G') = \frac{|E \cup E' \setminus E \cap E'|}{|E|}$, *achieves a required value* θ.

In this problem, the graph G represents the network neighborhood around a querying user's node u. The parameter k defines the view of that neighborhood that user wishes to obtain. By definition, the obtained graph G' effectively reveals which users are within k hops of u or of each other.

Furthermore, we propose that each user u in the network may set: (i) an one-to-one distance threshold d_u, which defines that any user u' lying at most d_u away from u can obtain information about their connection; and (ii) a universal distance threshold k_u, which defines that the information of u lying k_u *or more* hops away from any user u' can be revealed to a third user u''. A cautious user u would

set a *low* d_u threshold (i.e, would prefer to reveal distance information only to close connections), and a *high* k_u threshold (i.e., would prefer not to let one's close connections to be accurately known by strangers). Generous default values could be set as $d_u = 3$ and $k_u = 2$.

For a user's node in the network, say u', let $d(u, u')$ be the actual network distance between u' and the querying user u. Then, if $k \geq d(u, u') > d_u$, i.e., if u' has not consented for her network distance from u to be revealed to u, then u' shall not be included in the neighborhood graph G we examine, as presented to u. Furthermore, if $k_u > k$, i.e., if u' has not consented for the information that her network distance from any other node is at most k hops to be revealed to *third* parties, then, again, u' is not to be included in G as presented to such third parties. In effect, G as presented to a querying user u would only contain the nodes of those users who are comfortable having their distance from u, being less than k, revealed to u (or whose distance from u is larger than k) *and* are comfortable with information about their at-most-k-hop connections to other users being revealed to u as well. Thus, users can define their own privacy objectives [12].

The requirement that $G^k = G'^k$ in Problem 1 defines our ideal objective. A graph G' that satisfies this *reachability requirement* for a large value of θ may not exist, and, even if it exists, may be hard to find. After all, this reachability requirement is strict, and does not allow much flexibility. In many practical circumstances, a more flexible version of the same requirement may still satisfy our objectives. Therefore, we suggest such a *relaxed* version of the reachability requirement that would be easier to satisfy while still maintaining much of the information we wish to preserve.

2.2 Relaxing the Reachability Requirement

Let $d(v_1, v_2)$ ($d'(v_1, v_2)$) be the distance of vertex v_2 from vertex v_1 in G (G'). Then the standard reachability requirement, i.e., the requirement that $G^k = G'^k$, can be analytically expressed as follows:

Definition 2. **Reachability Requirement (RR)** *A graph $G'(V, E')$ is said to satisfy the* reachability requirement *with respect to an original graph $G(V, E)$ for a given integer k, if and only if $|E| = |E'|$, and, for any pair of nodes $v_1, v_2 \in V$, it holds that $d(v_1, v_2) \leq k \Leftrightarrow d'(v_1, v_2) \leq k$.*

The strictness of the standard reachability requirement emanates from the fact that a distance that does not exceed k in G should not exceed k in G' either, and vice versa. A slightly less rigorous version of this requirement would impose a lighter constraint by allowing for some laxness in the preservation of distances with a definite threshold k. In effect, we can relax the requirement by demanding only that a distance not exceeding $k-1$ in G does not exceed k in G', and vice versa. This relaxation is twofold: First, we reduce the amount of distances involved, as we now care only for distances in the range $[1, k-1]$ instead of the range $[1, k]$. Second, we introduce some laxness in the preservation of distances within this range, by allowing that each distance in the range $[1, k-1]$ in G is mapped to a distance in a wider range, namely the range $[1, k]$ in G', and vice versa. We express this relaxed requirement as follows:

Definition 3. **Relaxed Reachability Requirement (RRR)** *A graph $G'(V, E')$ satisfies the* relaxed reachability requirement *with respect to an original graph $G(V, E)$ for a given integer k, if and only if $|E| = |E'|$, and, for any pair of nodes $v_1, v_2 \in V$, the following implications hold:*

$$d(v_1, v_2) < k \quad \Rightarrow \quad d'(v_1, v_2) \leq k$$
$$d'(v_1, v_2) < k \quad \Rightarrow \quad d(v_1, v_2) \leq k$$

Under this relaxation, G' still presents representatively small distance values (i.e., values $d' \leq k$) for short distances in G (i.e., $d < k$) and avoids the misrepresentation of longer distance values in G (i.e., values $d > k$) as short in G' (i.e., as $d < k$). Thus, we contend that a graph G' satisfying the relaxed, instead of the standard, reachability requirement with respect to G provides slightly less precise, but still rich, information about the distances between vertices of interest, yet allows for much-desired higher flexibility in modifying the graph, which allows for a higher degree of protection against structural attacks. In the following section we present an algorithm that generates graphs satisfying either the RR or the RRR with respect to an original graph G, and hence provides an avenue for revealing a modified, utility-preserving and discretionary version of G.

Algorithm 1: SRG

Input: graph G with V vertices and E edges; reachability k; distortion threshold θ;
Result: Modified Graph G'

```
compute distance matrix D(G);
initialize G' as G;
initialize delete-candidate edge list L1, length l1;
initialize add-candidate edge list L2, length l2;
while Dist(G, G') < θ do
    for λ ← 1 to min{l1, l2} do
        for each edge set C1 ← (l1 choose λ) do
            for each edge set C2 ← (l2 choose λ) do
                delete C1 from and add C2 to G';
                if G' satisfies (R)RR wrt G then
                    update L1 and L2;
                    Break for loops;
                else
                    add back C1 and delete C2;
Return G'(V, E');
```

2.3 Algorithm

The problem could be tackled by an exhaustive-search algorithm that would try out all combinations of edges that could make a modified graph. However, such an exhaustive search becomes computationally prohibitive as the size of the graph grows. Instead, our Similar Reachability Graph (SRG) algorithm (Algorithm 1) modifies the graph step by step, by alternatively adding or deleting one edge at time. At each step, we opt for a modification that satisfies the standard (or relaxed) reachability requirement. As long as modifications that satisfy the requirement are possible, we keep updating the graph, while keeping track of the distortion inflicted thereon (i.e., the number of edges altered). Once the inflicted distortion reaches a desired level θ, the algorithm terminates and the modified graph is output.

Our SRG algorithm makes use of a basic operation that computes the distance matrix $\mathcal{D}$ of a graph G. Having the $\mathcal{D}$ of the original graph G, as well as the distance matrix $\mathcal{D}'$

of a modified graph G', we can check whether the standard or relaxed reachability condition is satisfied, and calculate the respective k-reachability graphs G^k and G'^k as well. To that end, we employ the Warshall-Floyd algorithm [7], with extra pruning and optimization provisions, so as to eschew the computation of distances larger than the k threshold, which is, unnecessary for our problem.

At first, SRG constructs lists of edges that are candidate for addition (deletion). All edges in G are candidates for deletion, while edges candidate for addition are those that do not exist in G but exist in G^k. In more detail, SRG starts out with the original graph G, and proceeds to perform iterative modification steps. At each iteration, it progressively checks all allowed combinations of λ edges to delete and λ edges to add, starting with $\lambda = 1$ and increasing λ progressively, until it detects an add/delete combination that produces a modified graph G' satisfying the (relaxed) reachability requirement, (R)RR, with respect to G. Having succeeded in this iteration, it proceeds to modify the obtained graph G' further in the next iteration.

We emphasize that the satisfaction of the (R)RR is always checked with respect to the original graph G, not to the modified graph of the preceding step. Thus, throughout the modification iterations, we always maintain a modified graph G' that satisfies the (R)RR with respect to G.

These modification iterations terminate when the modified graph G' has achieved a *desired* difference from the original graph G, for the sake of withstanding structural attacks. We measure the difference between graphs $G(V, E)$ and $G'(V, E')$ in terms of *distortion*, defined as the ratio of the number of edges they do *not* share to $|E|$: $Dist(G, G') = \frac{|E \cup E' \setminus E \cap E'|}{|E|}$; since $|E|$ is not changed by the algorithm, the distortion depends on the amount of edges altered, $|E \cup E' \setminus E \cap E'|$. Distortion values near 100% (i.e., half the maximum possible value of 200%) provide the highest obfuscation, as one can tell with confidence neither that an edge in G' also appears in G, nor that it does not. This metric has also been used as a vague way of measuring *information loss* in previous research [19]; we employ simply as a measure that show how much a graph is being distorted, without making any claim that correctly captures any other quality.

Our SRG algorithm works with both the standard reachability requirement (RR) and the relaxed one (RRR). The satisfaction of this requirement is checked in Step 10, by comparing the distance matrix of the modified graph, (G'), to that of the original graph. In the next section we proceed to an experimental study, in which we opt for using the RRR; this choice allows for higher flexibility, while still preserving, as we will show, rich structural information.

The SRG algorithm is a heuristic, and its practicability rests largely on the expectation that a modified graph G' satisfying the (R)RR will be arrived at early, before the value of λ grows beyond value 2. This expectation is verified by our experiments. For the sake of completeness, we provide a worst-case complexity analysis. In a worst-case scenario, half of the possible edges are present in the graph, i.e., $\ell_1 = \ell_2 = \ell = \frac{n(n-1)}{4}$, yielding $\sum_{\lambda=1}^{\ell} \binom{\ell}{\lambda} = O\left(2^{\frac{n^2}{4}}\right)$ selections of edge sets for addition and removal, hence $O\left(2^{\frac{n^2}{2}}\right)$ graph modifications in total. Since the distance matrix computation by the Warshall-Floyd algorithm costs $O\left(n^3\right)$, the overall complexity is $O\left(2^{\frac{n^2}{2}} n^3\right)$. In practice, we expect our algorithm to terminate without raising such high computational demands, as soon as a graph G' satisfying the (R)RR is discovered (Lines 10-12).

3. EXPERIMENTAL EVALUATION

In this section we evaluate our algorithm using real data sets. The experiments ran on an Intel Core, 2 Quad CPU, 2.83GHz, 4GB machine running Windows 7. The algorithm was implemented in Standard C, while computations of matrix utility measures were conducted in Python.

3.1 Data Description

We used two real data sets, representative of social network graphs, which are made freely available for research purposes. The former, Flickr[3][21], contains user-to-user links in an online social network for image and video hosting. Five subgraphs used in our experiments are uniformly sampled with 50 vertices and around 100 edges for each. The latter data, Gnutella[4], describes a peer-to-peer file sharing network. Nodes represent hosts in the network topology and edges connections between hosts. We uniformly sample 5 connected subgraphs of the 2002 Gnutella network snapshot, containing 50 vertices and around 52 edges for each subgraph. We emphasize that the data sizes we test are representative of the small neighborhoods graphs that arise in the applications we envisage. We focus on how the *structure* of such graphs can be published in a discretionary and information-rich manner. We are not making any assumption on how, and to what extent, other information in those graphs, e.g., node attributes, may be revealed. The problem of publishing such attributes can be treated using techniques for microdata anonymization [4], as in [5], and is orthogonal to the problem of publishing the graph structure.

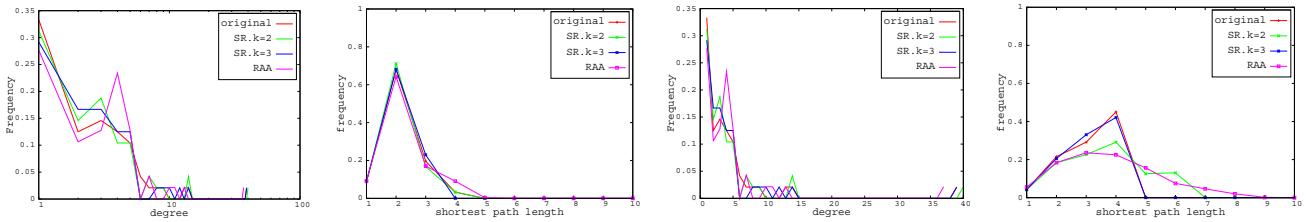
3.2 Utility Assessment

We claim that, apart from, and because of, satisfying the reachability constraint, graphs generated by our SRG algorithm preserve other structural properties of the original graph G. To demonstrate our claim, we compare graphs obtained by our methods to graphs *of the same distortion* obtained by the randomized anonymization technique proposed by Hay et al [14]. This technique modifies the original graph by randomly deleting a prescribed number of edges and randomly adding the same number of edges; thus, the resulting graph has the same number of edges as the original graph. We refer to the algorithm of Hay et al. as "RAA".

In our first experiment, we present the degree distribution and distribution of pairwise shortest-path (geodesic) distances, of the original Flickr graph, its SRG-generated modification (SR), and a random perturbation of the same graph by RAA having the same distortion. Figure 5 (a)(b) shows the results for graphs in which we allow distortion 0.16 and set $k = 2$ and $k = 3$ separately. We observe that, as we expected, the distribution of those features with SRG resembles those of the original graph more faithfully than those of RAA.

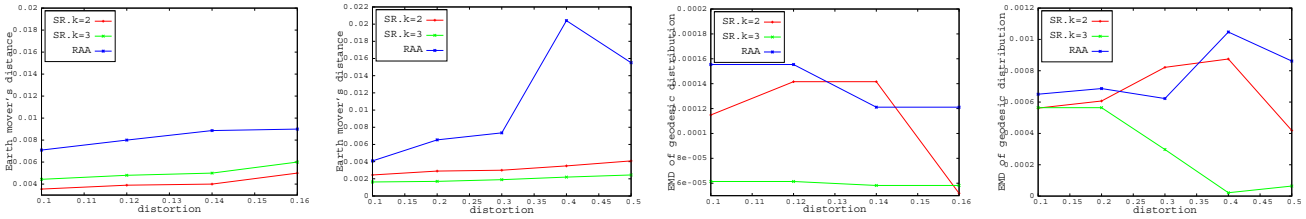
Next, we present results on the same measures with the Gnutella data. Figure 5 (c)(d) presents our measurements when we allow distortion to reach 0.5 and set $k = 2$ and $k = 3$ separately. In both figures the distributions for SRG

[3]Available online at http://socialnetworks.mpi-sws.org/
[4]Available online at http://snap.stanford.edu/data/



(a) DD, Flickr, distortion 0.16 (b) GD, Flickr, distortion 0.16 (c) DD, Gnutella, dist 0.5 (d) GD, Gnutella, dist 0.5

Figure 5: Degree distribution (DD) and geodesic distribution (GD) results



(a) EMD(DD), Flickr (b) EMD(GD), Flickr (c) EMD(DD), Gnutella (d) EMD(GD), Gnutella

Figure 6: Earth mover's distance of degree distribution and geodesic distribution

graphs stand relatively closer to those of the original graph. This outcome further confirms our contention that our method provides a solid way to keep other structural graph properties under tight control. Interestingly, we observe how the SRG graph with $k = 3$, even under the relaxed reachability requirement, does not allow any shortest-path distance to exceed the original graph's diameter 4 (Figure 5 (d)).

Next, to obtain a more precise estimation of the degree to which SRG graphs resemble the original ones, we measure the metric that express their structural divergence: the Earth-Mover's Distance (EMD) [23] between the original and modified degree distributions, for different distortion values. Figure 6 (a)(c) shows the EMD between the degree distributions on SRG graphs with $k = 2$ and $k = 3$, and RAA-perturbed versions of the original Flickr and Gnutella graphs, respectively, and the original ones, as a function of their distortion, while Figure 6 (b)(d) shows the EMD between the geodesic distributions. We observe that, as expected, the measured metric on the SRG graphs diverge from those of the original graph much less than those on the RAA graph, even though all graphs are obtained with the same distortion.

Remarkably, the SRG graph with $k=2$ fares better than that for $k=3$ with Flickr data, but not with Gnutella data. This deviation is not surprising; the parameter k that allows for the best preservation of other structural graph properties under the same distortion depends on the nature of the data at hand; in some cases, a lower k may be advantageous, as it enforces the preservation of short-distance links; in other cases, a higher k may be preferable, as it encompasses more vertex pairs under its scope.

Then, we assess the divergence between original and anonymized graphs on other graph properties: the average local clustering coefficient, the average shortest path length, the graph diameter and radius. For each data set, the results are averaged over 5 subgraphs, with 5 runs for each subgraph. Figure 7 shows the results for the Flickr and Gnutella data. Again, we observe that the SRG graphs produce measures much closer to those of the original graphs than the RAA graphs do. These results corroborate our claim that SRG graphs can maintain properties of the original despite the inflicted distortion.

Given that we employ the relaxed reachability requirement in our experiments, the results to reachability queries are expected to have a slight error. We end our utility assessment by quantifying this error in terms of *precision* and *recall* measures on reachability queries, in which a user asks whether a target node is reachable within a certain number of k hops. In addition, we present our measures of *false negatives* and *false positives* under the same settings.

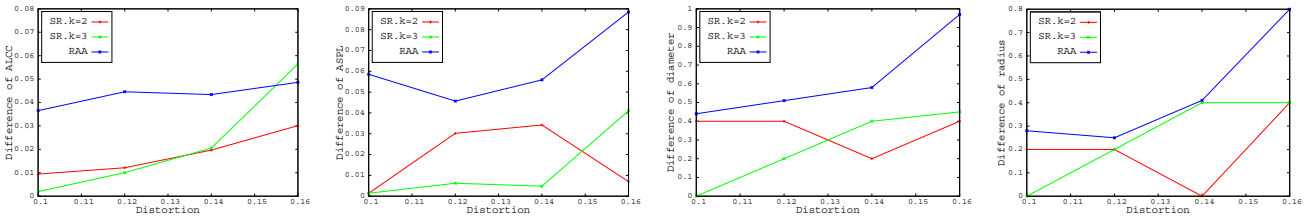
In particular, let V_o (V_m) be the set of vertices within k hops of the querying node in the original (modified) graph. The precision $\mathcal{P}$ and recall $\mathcal{R}$ are measured as follows:

$$\mathcal{P} = \frac{|V_o \cap V_m|}{|V_m|} \quad \mathcal{R} = \frac{|V_o \cap V_m|}{|V_o|} \tag{1}$$

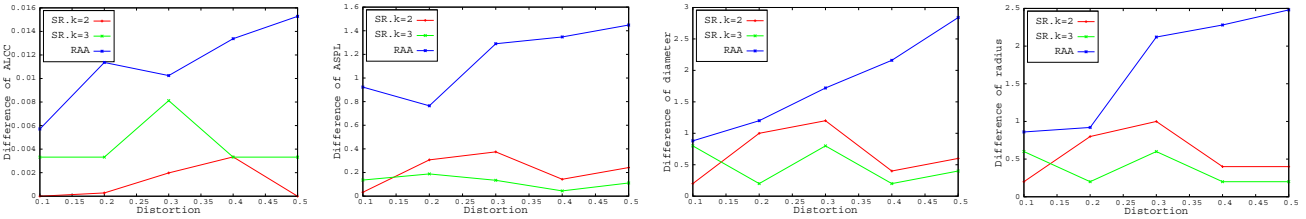
Similarly, our *false negatives* and *false positives* metrics are measured as:

$$\mathcal{FN} = \frac{|V_o \setminus V_m|}{|V|} \quad \mathcal{FP} = \frac{|V_m \cap V_o|}{|V|} \tag{2}$$

where V is the graph's complete vertex set. We measure each of these metrics on each vertex and average our results over all vertices in the graph. Figure 8 shows our results with both the Flickr and Gnutella data, for graphs modified by the SRG and RAA algorithms, for queries involving number of hops $k = 2$ and $k = 3$. For example, each dot on the red line in Figure 8(a) represents the average precision for 2-hop queries. As in our previous measurements of graph properties, all results are averaged over 5 extracted subgraphs and 5 runs for each subgraph, so as to diminish the effect of randomness. In all examined cases, the SRG algorithm achieves higher precision and recall measures, and lower false negatives and positives, than RAA; the difference is more conspicuous with the Gnutella data. This outcome reconfirms that the SRG algorithm preserves reachability information more accurately than random distortion does, which is exactly the aim this algorithm is made for.

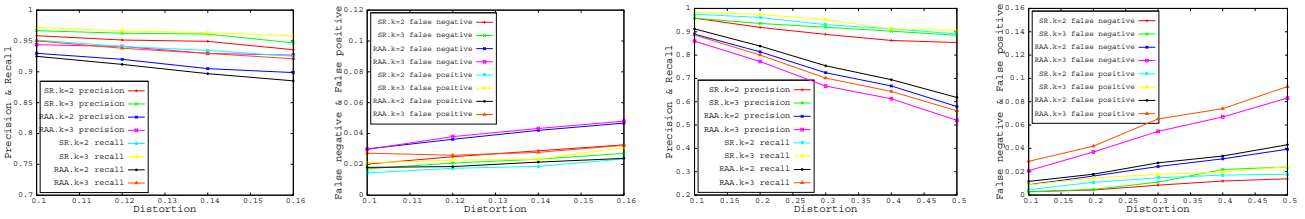


(a) avg local clustering coeff (b) avg shortest path length (c) diameter (d) radius



(e) avg local clustering coeff (f) avg shortest path length (g) diameter (h) radius

Figure 7: Graph properties with increasing distortion, Flickr (a-d) and Gnutella (e-h)



(a) precision and recall (b) false negatives/positives (c) precision and recall (d) false negatives/positives

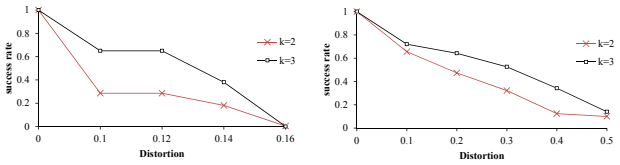
Figure 8: Precision and Recall, False negatives and False positives, Flickr (a-b) and Gnutella (c-d)

3.3 Resistance to Structural Attacks

We now turn our attention to assessing the extent to which are graphs can resist attacks based on an adversary's structural knowledge. The resistance to such attacks ensures that the network's structure is released in a way that does not allow the inference of individual users' identity, while at the same time providing the utility that we expect, as we have witnessed in the previous section. We contend that the graphs released by our method are capable to withstand structural identification attacks with high probability, hence provide a measurable amount of protection on that front.

To illustrate this protection, we experimentally measure the extent to which our distorted graphs can resist structural attacks of the kind suggested in [1]. While [1] propose a specific attack algorithm, the *walk-based attack*, we go one step further and measure the success rate for *any* attack based on the identification of an embedded subgraph in the distorted graphs, as a function of the amount of distortion incurred on it. Such a structural attack is assumed to succeed *if* the adversary can identify an embedded graph in the released graph; as we have discussed, such an embedded graph may consists of fake accounts created before graph releasing and connected among themselves and to other, victim nodes, so as to follow a unique and identifiable pattern.

The identification of the maliciously embedded subgraph depends on the information of degree and internal structure. Intuitively, the more distorted a graph is, the less likely it becomes than a structural attack will succeed, and hence higher protection of individual users is afforded. Besides, the more distorted a graph is, the less it can be relied upon to provide truthful information at its microstructure. Arguably, a graph that presents high distortion at its microstructure while still maintaining truthful overall structural properties at its macrostructure would satisfy our purposes. On the other hand, in case all edges in the embedded subgraph are preserved after the transformation process and no others are added, then the attack can be launched successfully. We contend that this state of affairs arises rarely and its likelihood drops with increasing distortion.



(a) Flickr (b) Gnutella

Figure 9: Success rate of structural attack

We measure the *success rate* of the described structural attack vs. the distortion of the graph in which a malicious subgraph has been embedded. For each data set, we embed 50 different subgraphs prior to the graph's distortion. For each of the resulting *attacked* graphs, we conduct 10 separate runs of SRG perturbation, where we randomly shuffle the order in which edges are examined so as to obtain nondeterministic results; thus we obtain 10 different distorted versions of the original attacked graph, at the same desired distortion level. The success rate of the attack on the original data, for the obtained distortion, is measured as the total ratio of successful attacks over the total 10×50 runs.

Figure 9 (a) and (b) shows our results, for the Flickr and Gnutella data sets, respectively, and for two different values of the reachability parameter k. Our results confirm our expectations: as distortion grows, it becomes harder for the attack to succeed. Remarkably, we obtain low success rates even at distortion levels in which, as our utility assessment experiments show, we also preserve structural graph properties with satisfactory fidelity.

3.4 Discussion

Previous suggestions on discretionary social network publication follow a similar format: they define a privacy principle the published graph should obey, and proceed to alter the given graph so as to satisfy this constraint (see Section 4). Still, they do not offer a respectively comprehensible utility guarantee; as a consequence, they do not focus on providing data that a social network user may find useful. After all, such techniques are designed with the assumption that the whole-network data is published to an external data recipient, e.g., a researcher; their aim is not to enable user-centric revelation of network subgraphs to SNS users themselves.

We suggested a user-oriented alternative, aiming to provide a picture of a subgraph of interest that preserves certain structural properties, thereby offering a *utility guarantee*. The subgraph consists of an end-user, as a central user, and a neighborhood of other users that the central user is interested in. In this case, users' confidentiality is catered for by distorting the graph as far as possible under a utility constraint. Unlike the common scenario in the literature, our method is mainly designed for daily usage scenarios where SNS users want to assess their position among their peers and their ability to expand their online network in a desired direction. This facility would be especially useful for users that subscribe to SNSs with the aim of expanding their social or professional circles. Users who value such information would be able to opt in such an information revelation scheme in a give-and-take manner, as they would also be willing to disclose some of their own information to gain from the networking potential the scheme provides.

By our approach the graph is distorted in some respects, so as to forestall attacks by adversaries with structural knowledge, yet preserves certain topological properties in other respects. Arguably, our methodology facilitates certain desirable user behaviors. Eventually, we argue that we can promote networking in an SNS by benevolent users while protecting such users from malicious attackers aiming to exploit the same information to undesirable ends.

Our results show that by using our proposed approach not only is reachability information guaranteed, but other structural properties are also preserved, which means that users can be provided with views of their extended neighborhood that will be representative of the real network, even if somewhat distorted in order to thwart malicious users. We envision that such views could consist of abstracted visual representations of one's extended neighborhood, e.g. in the form of concentric circles, or clouds that will indicate reachability and overall structure, so that a user may assess the distance to another user of interest as well as the density, complexity, clustering and other structural properties of the network neighborhood. Our method allows for the creation of various such network views that would be beneficial to the benevolent user. For instance, one looking at the graph formed by one's friends can accurately infer how tightly connected those friends are with each other; for example, a large diameter implies one's social connections are wide spread, while a small one implies that one is connected only to people already well-connected with each other. Such information may be of particular utility to a public personality (election candidate, actor, athlete) visualizing ones fan club. Alternatively, someone using a network to promote their work (e.g., a musician) may be interested to identify the most influential nodes in that network, and the number of such nodes. The preservation of properties such as degree distribution is instrumental for that purpose.

4. RELATED WORK

Research on the overall problem of revealing social network graphs to third parties was initiated, with a focus on protecting privacy, by a cardinal observation by [1]: even if a network graph is de-annotated before being published, an adversary can infer the identity of nodes by solving a restricted graph isomorphism problem. While pointing out the arising challenge and presenting ways in which an adversary can carry out an attack based on structural knowledge, [1] did not propose ways to alleviate this problem.

A particular technique for publishing a graph in a privacy-preserving manner was first proposed by Zheleva and Getoor [28], aiming to conceal a particular sensitive subset of the graph's edges. Korolova et al. [16] considered the problem posed by an adversary who breaks into SNS user accounts, gains information on a set of local neighborhoods, and tries to re-assemble the network graph using those.

Two works study the problem of preventing structural re-identification of a node by adversaries who know a target's local neighborhood. Zhou and Pei [29] study the problem on *node-labeled* graphs, while Hay et al. [13] address the same problem on unlabeled graphs; both proposed notions of *anonymity* that aim to render a node's neighborhood isomorphic to (i.e., indistinguishable from) $k-1$ others; the techniques they propose for achieving these objectives provide only coarse information on the graph's structure. Liu and Terzi [19] were the first to suggest an algorithmic anonymization technique designed for simple graphs with unlabeled nodes and uniform edges. This anonymization method first renders the graph's degree sequence k-anonymous, using the algorithm in [9], and then inserts edges into the graph so as to render it *k-degree anonymous*, i.e., ensure that any degree value appears at least k times. Zhang and Zhang [27] discerned that, even if a graph preserves vertex anonymity, it may still not preserve edge anonymity; they suggested the notion of *τ-confidence*, which limits an adversary's confidence that an edge exists between the vertices corresponding to two entities, and suggest heuristics to achieve this objective by edge swaps and deletions. In another direction, both [30] and [25] suggested methods to transform a data graph so that each node in the resulting graph is structurally indistinguishable from $k-1$ others. The ensuing property is dubbed *k-automorphism* in [30] and *k-symmetry* in [25]. [15] suggests an akin anonymization method based on partitioning. Cheng et al. [5] reiterated Zhang and Zhang's [27] observation that the protection against identity disclosure does not guarantee protection against the disclosure of *sensitive linkages*. To overcome this problem, they proposed a method that divides the graph into k *disjoint* subgraphs, rendering it *k-isomorphic*. While this method effectively protects against link disclosure, it severely alters the nature

of the published network. Last, [26] have examined the social network data revelation problem with a new twist, in which they consider that most of the nodes in the network face no privacy threats related to structural knowledge at all, while only a few nodes have such needs, arising from an adversary's knowledge of degrees and edge labels.

5. CONCLUSION

In this work, we have taken an alternative view to the problem of social network data sharing under confidentiality concerns. While most works in the area assume that a graph is to be published as a whole for research purposes, and attempt to maximize a vague measure of utility while observing a privacy condition, we take a user-centric and utility-oriented standpoint. We focus on the problem of revealing a subgraph of connections in a querying user's neighborhood. To that end, we define a utility guarantee involving a reachability property and suggest computational methods that distort the graph to a desired extent while observing this requirement. Our technique provides a view of a subgraph that preserves crucial properties while blurring the accuracy of individual linkages; thus, it offers a perturbed, albeit informative, view of the network. Our experimental study confirms that (i) graphs obtained with our scheme *do* preserve large-scale structural properties of the original graphs more faithfully than graphs that have undergone the same amount of distortion by a previously proposed random perturbation technique, while (ii) they also pose satisfactory resistance to previously proposed structural attacks.

6. REFERENCES

[1] L. Backstrom, C. Dwork, and J. Kleinberg. Wherefore art thou R3579X?: Anonymized social networks, hidden patterns, and structural steganography. In *WWW*, 2007.

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

[3] D. M. Boyd and N. B. Ellison. Social network sites: Definition, history, and scholarship. *Journal of Computer-Mediated Communication*, 13:210–230, 2008.

[4] B.-C. Chen, D. Kifer, K. LeFevre, and A. Machanavajjhala. Privacy-preserving data publishing. *Found. Trends databases*, 2(1-2):1–167, January 2009.

[5] J. Cheng, A. W.-C. Fu, and J. Liu. k-isomorphism: Privacy-preserving network publication against structural attacks. In *SIGMOD*, 2010.

[6] N. B. Ellison, C. Steinfield, and C. Lampe. The benefits of facebook "friends": Social capital and college students' use of online social network sites. *Journal of Computer-Mediated Communication*, 12:1143–1168, 2007.

[7] R. W. Floyd. Algorithm 97: Shortest path. *Communications of the ACM*, 5(6):345, 1962.

[8] F. Fukuyama. *Trust: The Social Virtues and the Creation of Prosperity*. Free Press, New York, first edition, 1995.

[9] G. Ghinita, P. Karras, P. Kalnis, and N. Mamoulis. Fast data anonymization with low information loss. In *VLDB*, 2007.

[10] S. Goel, R. Muhamad, and D. Watts. Social search in "small-world" experiments. In *WWW*, 2009.

[11] M. Granovetter. The strength of weak ties: A network theory revisited. *Sociological Theory*, 1:201–233, 1983.

[12] S. Gürses and B. Berendt. The Social Web and Privacy: Practice, Reciprocity and Conflicts in Social Networks. In F. Bonchi and E. Ferrari, editors, *Privacy-Aware Knowledge Discovery: Novel Applications and New Techiques*. Chapman and Hall/CRC, 2010.

[13] M. Hay, G. Miklau, D. Jensen, D. Towsley, and P. Weis. Resisting structural re-identification in anonymized social networks. *PVLDB*, 1(1), 2008.

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

[15] X. He, J. Vaidya, B. Shafiq, N. Adam, and V. Atluri. Preserving privacy in social networks: A structure-aware approach. In *WI-IAT*, 2009.

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

[17] J. Leskovec and E. Horvitz. Planetary-scale views on a large instant-messaging network. In *WWW*, 2008.

[18] D. Z. Levin and R. Cross. The strength of weak ties you can trust: The mediating role of trust in effective knowledge transfer. *Management Science*, 50(11):1477–1490, 2004.

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

[20] S. Milgram. The small world problem. *Psychology Today*, 2:60–67, 1967.

[21] A. Mislove, H. S. Koppula, K. P. Gummadi, P. Druschel, and B. Bhattacharjee. Growth of the flickr social network. In *WOSN*, 2008.

[22] C. R. Palmer, P. B. Gibbons, and C. Faloutsos. ANF: a fast and scalable tool for data mining in massive graphs. In *KDD*, 2002.

[23] Y. Rubner, C. Tomasi, and L. J. Guibas. The earth mover's distance as a metric for image retrieval. *International Journal of Computer Vision*, 40(2):99–121, 2000.

[24] D. J. Watts. *Six Degrees: The Science of a Connected Age*. W. W. Norton, New York, 2003.

[25] W. Wu, Y. Xiao, W. Wang, Z. He, and Z. Wang. k-symmetry model for identity anonymization in social networks. In *EDBT*, 2010.

[26] M. Yuan, L. Chen, and P. S. Yu. Personalized privacy protection in social networks. *PVLDB*, 4(2):141–150, 2010.

[27] L. Zhang and W. Zhang. Edge anonymity in social network graphs. In *CSE*, 2009.

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

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

[30] L. Zou, L. Chen, and M. T. Özsu. k-automorphism: A general framework for privacy-preserving network publication. *PVLDB*, 2(1), 2009.