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

Yi Song[⋆]
Giorgos Cheliotis[⋆]
[⋆]NUS, Singapore

Panagiotis Karras[#]
Mingqiang Xue[◊]
[#]Rugters University, USA

Sadegh Nobari[⋆]
Stéphane Bressan[⋆]
[◊]I²R, Singapore

ABSTRACT

The proliferation of online social networks has created intense interest in studying their nature and revealing information of interest to the end user. At the same time, such revelation 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 a clear and quantifiable utility measure over graph data. In this paper, we take a different approach: we define a *utility guarantee*, in terms of certain graph 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 network graph, in which that user as well as 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 data, we demonstrate that our method preserves several properties of interest more successfully than methods that randomly distort the graph to an *equal* extent, while withstanding structural attacks proposed in the literature.

Categories and Subject Descriptors

G.2 [**Discrete Mathematics**]: Graph Theory; H.2 [**Database Management**]: Database Applications; K.4.1 [**Computers and Society**]: Public Policy Issues—*Privacy*

Keywords

Data utility, social network, security and privacy

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 [16]. The capacity to exchange information in such networks rests on an assumed underlying trust among users [7]. 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 [12]. 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 [8]. SNSs are valuable for the development of social capital, in particular for bridging social capital [5]. In short, the technological affordances of SNSs provide leverage in building weak ties, while the value of these ties for an individual is mediated by interpersonal trust [12].

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. 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 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, deterring malicious users, while at the same time provides certain utility for benevolent users. This problem of *discretionary user-centric network data release* 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 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 loss of utility low [1, 9, 13, 2, 4, 18, 22]. 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*, which do not capture the extent to which an object as complex as a graph maintains its original properties. Nevertheless, in case the information recipients are end-users of the social network site, they would like to have a guarantee precisely on the *utility* of the released data, in terms of certain *graph properties*, no less than they would desire a certain *privacy guarantee* about their own information being revealed.

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.

A *naive anonymization* of G would substitute all entity identifiers in G by synthetic identifiers. However, such an anonymiza-

CIKM'12, October 29–November 2, 2012, Maui, HI, USA.

tion does not conceal the identities behind the published graph, as the structural information in the network can itself serve to identify nodes [1, 9, 13, 23, 20]. 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. 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. To be truly useful, this subgraph should correctly reveal the identities of individuals within its scope and also provide some indication as to their relative positions. However, for the sake of confidentiality, the subgraph should not reveal their *precise* relationships 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 poses problems both from a privacy and a utility point of view: In terms of privacy, the revelation of individuals along the path poses a risk, as the relationships among distant connections to the querying user may be sensitive. 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 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 1: Visualization of connections in LinkedIn

Meanwhile, Xing shows all intermediate connections and even provides names along a single path. If taken further, i.e., to longer paths, 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.

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

1.2 Our proposal

Motivated by the above discussion, we suggest a methodology for revealing social network data to relevant users following a *utility-driven* paradigm, similar in spirit to [21]. 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].

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.

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 shortest-path *distance* between two individuals is usually rather small, not exceeding *six* steps. Milgram's small world experiment [14] 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 [19] 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 [11] found the average distance among users of an instant messaging system to be 6.6.

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 [4], 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. 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.

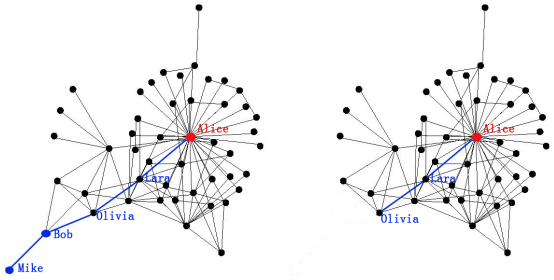
As we discussed, professional networking platforms provide a function that concerns us: when users search for someone, they can see the path that leads from their node to the searched-for person, possibly under the condition that the path is not longer than 3 hops.

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

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.

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 2(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 2(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.



(a) (b)

Figure 2: Example of path revelation.

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.

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.

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 iff *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 3 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.

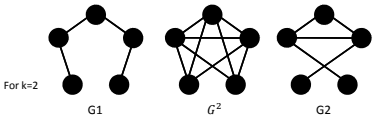


Figure 3: Graphs G_1 and G_2 having the same G^2

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 edge-edit-distance between* G *and* G', $Dist(G, G') = \frac{|E \cup E' \setminus E \cap E'|}{|E|}$, *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. The obtained graph G' reveals users within k hops of u or of each other.

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 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, *iff* $|E| = |E'|$, *and, for any pair of nodes* $v_1, v_2 \in V$, *it holds that* $d(v_1, v_2) \leq k \Leftrightarrow d'(v_1, v_2) \leq k$.

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: we reduce the amount of distances involved, as we now care only for distances in the range $[1, k - 1]$ instead of the range $[1, k]$, and 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:*

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

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 $d < k$). 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.

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. Yet such an exhaustive search becomes computationally prohibitive as the size of the graph grows. Our Similar Reachability Graph (SRG) algorithm (Algorithm 1) modifies the graph 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 such modifications are possible, we keep updating the graph, monitoring the distortion inflicted thereon. Once the distortion reaches a desired level θ, the algorithm terminates and the modified graph is output.

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');
```

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. To that end, we employ the Warshall-Floyd algorithm [6], with extra pruning and optimization provisions, eschewing the computation of distances larger than the k threshold.

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; it 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 we always maintain a modified graph G' that satisfies the (R)RR with respect to G.

The modification terminates when the modified graph G' has achieved a *desired* difference from the original graph G. 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.

The 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, which allows for higher flexibility.

3. EXPERIMENTAL EVALUATION

We now 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 utility measure computations were done in Python.

3.1 Data Description

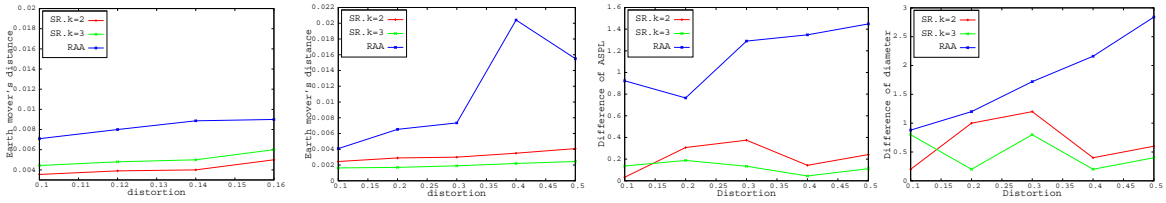
We used two real data sets, representative of real social networks, which are made freely available for research purposes. The former, Flickr[3][15], contains user-to-user links in an online social network for image and video hosting. Five subgraphs used in our experiments are uniformly sampled with 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. In all our experiments, results are averaged over 5 subgraphs, with 5 runs for each subgraph. The data sizes we test are representative of the neighborhoods graphs that arise in the applications we envisage.

3.2 Utility Assessment

We claim that 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 algorithm (RAA) of Hay et al. [10]. 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.

To assess the degree to which SRG graphs resemble the original ones, we measure the Earth-Mover's Distance (EMD) [17] between the original and modified degree distributions, for different distortion values. Figure 4(a) shows the EMD between the degree distributions on SRG graphs with $k = 2$ and $k = 3$, and RAA-perturbed versions of the Flickr graphs, and the original ones, as a function of distortion, while Figure 4(b) shows the EMD between their geodesic distance distributions. As expected, the measured metric on the SRG graphs diverge from those of the original graph much less than those on the RAA graph.

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

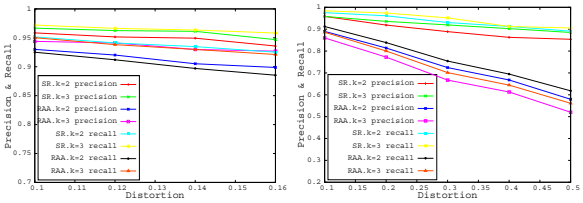


(a) EMD(DD), Flickr (b) EMD(GD), Flickr (c) avg shortest path length (d) diameter

Figure 4: EMD of degree distribution and geodesic distribution, Flickr (a-b), properties with increasing distortion, Gnutella (c-d)

Then, we assess the divergence between original and anonymized graphs on other graph properties: the average shortest path length and graph diameter. Figure 4(c-d) show the results for the Gnutella data. Again, the SRG graphs produce measures much closer to those of the original graphs than the RAA graphs do.

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. We measure each of these metrics on each vertex, and average our results over all vertices in the graph, over 5 extracted subgraphs, and 5 runs for each subgraph. Figure 5 shows our results with both the Flickr and Gnutella data, for graphs modified by the SRG and RAA algorithms, for queries involving $k = 2$ and $k = 3$ hops. In examined cases, the SRG algorithm achieves higher precision and recall measures than RAA.

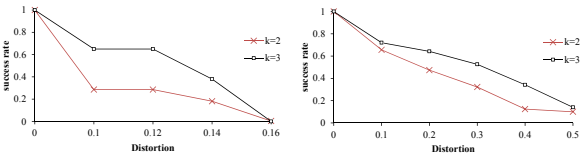


(a) precision and recall (b) precision and recall

Figure 5: Precision and Recall Flickr (a) and Gnutella (b)

3.3 Resistance to Structural Attacks

We now assess the extent to which distorted graphs can resist structural attacks of the kind suggested in [1]. We measure the *success rate* for *any* attack based on the identification of an embedded subgraph, vs. the distortion of the graph in which a malicious subgraph is 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 non-deterministic results; thus we obtain 10 different distorted versions of the original attacked graph, at the same distortion. The attack's success rate is measured as the ratio of successful attacks over 10×50 runs. Figure 6 shows our results for two different values of the reachability parameter k. Remarkably, we obtain low success rates even at distortion levels in which we preserve structural graph properties with satisfactory fidelity.



(a) Flickr (b) Gnutella

Figure 6: Success rate of structural attack

4. CONCLUSION

This work addressed the problem of social network data sharing under confidentiality concerns, from a utility-oriented standpoint, focusing on revealing a subgraph of connections in a user's neighborhood. We defined a utility guarantee involving a reachability property and suggested a method to distort the graph to a desired extent while observing this requirement. Our technique preserves crucial properties while blurring 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 random perturbation, while (ii) they also pose satisfactory resistance to structural attacks.

5. 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. *J. of Computer-Mediated Communication*, 13:210–230, 2008.

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

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

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

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

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

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

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

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

[12] 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.

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

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

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

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

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

[18] Y. Song, P. Karras, Q. Xiao, and S. Bressan. Sensitive label privacy protection on social network data. In *SSDBM*, 2012.

[19] D. J. Watts. *Six Degrees: The Science of a Connected Age*. Norton, 2003.

[20] M. Xue, P. Karras, C. Raïssi, P. Kalnis, and H. K. Pung. Delineating social network data anonymization via random edge perturbation. In *CIKM*, 2012.

[21] M. Xue, P. Karras, C. Raïssi, and H. K. Pung. Utility-driven anonymization in data publishing. In *CIKM*, 2011.

[22] M. Xue, P. Karras, C. Raïssi, J. Vaidya, and K.-L. Tan. Anonymizing set-valued data by nonreciprocal recoding. In *KDD*, 2012.

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