# Eight Friends Are Enough: Social Graph Approximation via Public Listings

Joseph Bonneau
Computer Laboratory
University of Cambridge
jcb82@cl.cam.ac.uk

Jonathan Anderson
Computer Laboratory
University of Cambridge
jra40@cl.cam.ac.uk

Frank Stajano
Computer Laboratory
University of Cambridge
fms27@cl.cam.ac.uk

Ross Anderson
Computer Laboratory
University of Cambridge
rja40@cl.cam.ac.uk

## ABSTRACT

The popular social networking website Facebook exposes a "public view" of user profiles to search engines which includes eight of the user's friendship links. We examine what interesting properties of the complete social graph can be inferred from this public view. In experiments on real social network data, we were able to accurately approximate the degree and centrality of nodes, compute small dominating sets, find short paths between users, and detect community structure. This work demonstrates that it is difficult to safely reveal limited information about a social network.

## Categories and Subject Descriptors

K.4.1 [**Computers and Society**]: Public Policy Issues — Privacy; E.1 [**Data Structures**]: Graphs and networks; F.2.1 [**Analysis of Algorithms and Problem Complexity**]: Numerical Algorithms and Problems; K.6.5 [**Management of Computing and Information Systems**]: Security and Protection

## General Terms

Security, Algorithms, Experimentation, Measurement, Theory, Legal Aspects

## Keywords

Social networks, Privacy, Web crawling, Data breaches, Graph theory

## 1. INTRODUCTION

The proliferation of online social networking services has entrusted massive silos of sensitive personal information to social network operators. Privacy concerns have attracted considerable attention from the media, privacy advocates and the research community. Most of the focus has been on *personal data privacy:* researchers and operators have attempted to fine-tune access control mechanisms to prevent the accidental leakage of embarrassing or incriminating information to third parties.

A less studied problem is that of *social graph privacy*: preventing data aggregators from reconstructing large portions of the social graph, composed of users and their friendship links. Knowing who a person's friends are is valuable information to marketers, employers, credit rating agencies, insurers, spammers, phishers, police, and intelligence agencies, but protecting the social graph is more difficult than protecting personal data. Personal data privacy can be managed individually by users, while information about a user's place in the social graph can be revealed by any of the user's friends.

### 1.1 Facebook and Public Listings

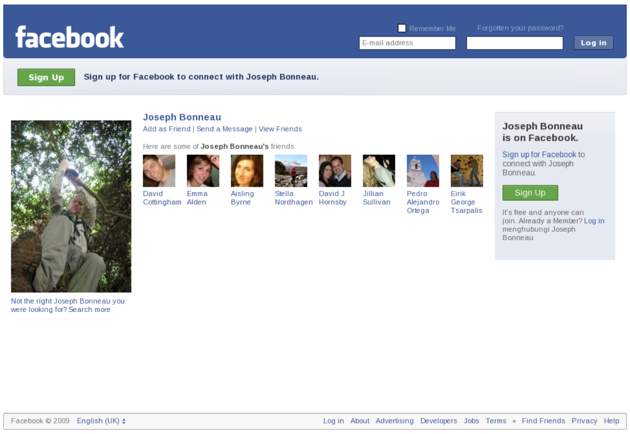
Facebook is the world's largest social network, claiming over 175 million active users, making it an interesting case study for privacy. Compared to other social networking platforms, Facebook is known for having relatively accurate user profiles, as people primarily use Facebook to represent their real-world persona [10]. Thus, Facebook is often at the forefront of criticism about online privacy [2, 17]. In September 2007, Facebook started making "public search listings" available to those not logged in to the site – an example is shown in Figure 1. These listings are designed to encourage visitors to join by showcasing that many of their friends are already members.

Originally, public listings included a user's name, photograph, and 10 friends. Showing a new random set of friends on each request is clearly a privacy issue, as this allows a web spider to repeatedly fetch a user's page until it has viewed all of that user's friends[1]. In January 2009, public listings were reduced to 8 friends, and the selection of users now appears to be a deterministic function of the requestin IP address[2].

*SNS* '09 Nuremberg, Germany

[1] Retrieving $n$ friends by repeatedly fetching a random sample of size $k$ is an instance of the *coupon collector's problem.* It will take an average of $\frac{n \cdot H_n}{k} = \Theta(\frac{n}{k} \log n)$ queries to retrieve the complete set of friends. A set of 100 friends, for example, would require 65 queries to retrieve with $k = 8$.

[2] Using the anonymity network Tor, we were able retrieve dif-



Sign up for Facebook to connect with Joseph Bonneau.

Joseph Bonneau

Here are some of **Joseph Bonneau's** friends:

**Joseph Bonneau is on Facebook.**

Sign up for Facebook to connect with Joseph Bonneau.

It's free and anyone can join. Already a Member? Log in menghubungi Joseph Bonneau

Not the right Joseph Bonneau you were looking for? Search more

Facebook © 2009

**Figure 1: An author's public Facebook profile**

Facebook has also added public listings for groups, listing 8 members in a similar fashion.

Critically, public listings are designed to be indexed by search engines, and are not defended technically against spidering. Members-only portions of social networks, in contrast, are typically defended by rate-limiting or legally by terms of use. We have experimentally confirme the ease of collecting public listings, writing a spidering script which retrieved ~250,000 listings per day from a desktop computer. This suggests roughly 800 machine-days of effort are required to retrieve the public listing of every user on Facebook, easily within the ability of a serious aggregator. Thus the complete collection of public listings can be considered available to motivated parties.

## 1.2 Privacy Implications

Our goal is to analyse the privacy implications of easily available public listings. The existence of friendship information in public listings is troublesome in that it is not mentioned in Facebook's privacy policy, which states only that: *"Your name, network names, and profile picture thumbnail will be made available to third party search engines"* [1]. Furthermore, public listings are shown by default for all users. Experimenting with users in the Cambridge network, we have found that fewer than 1% of users opt out. We feel this reflects a combination of user ignorance and poorly designed privacy controls, as most users don't want public listings – the primary purpose is to encourage new members to join. Users who joined prior to the deployment of public listings may be unaware that the feature exists, as it is never encountered by members of the site whose browsers store cookies for automatic log-in.

Leaking friendship information leads to obvious privacy concerns. "Social" phishing attacks, in which phishing emails are forged to appear to come from a victim's friend, have been shown to be significantly more effective than traditional "cold" phishing [12]. Private information can be inferred directly from one's friend list if, for example, it contains multiple friends with Icelandic names. A friend list may also be checked against data retrieved from other sources, such as known supporters of a fringe political party [19].

| Network | #Users | Mean $d$ | Median $d$ | Max $d$ |
|---|---|---|---|---|
| Stanford | 15,043 | 125 | 90 | 1,246 |
| Harvard | 18,273 | 116 | 76 | 1,213 |

**Table 1: Summary of Datasets used. $d$ = degree**

In this work though, we evaluate how much social graph structure is leaked by public search listings. While we are motivated by the question of what data aggregators can extract from Facebook, we consider the general question of what interesting properties of a graph one can compute from a limited view. We start with an undirected social graph $G =< V, E >$, where $V$ is the set of vertices (users) and $E$ is the set of edges (friendships). We produce a "publicly sampled" graph $G_k =< V, E_k >$, where $E_k \in E$ is produced by randomly choosing $k$ outgoing friendship edges for each node $v \in V$. We then compute some function of interest $f(G)$ and attempt to approximate it using another function $f_{\text{approx}}(G_k)$. If $f(G) \approx f_{\text{approx}}(G_k)$, we say that the public view of the graph leaks the property calculated by $f$. As $k \to \infty$, all information leaks, so we are most concerned with low values of $k$ such as the current Facebook value $k = 8$.

It is important to note that we assume $E_k$ is a uniformly random sample of $E$. This may not be the case if Facebook is specifically showing friends for marketing purposes. An interesting question for future work is if there are public display strategies which would make it harder to approximate useful functions of the graph.

# 2. EXPERIMENTAL DATA

In order to obtain a complete subgraph to perform experiments on, we crawled data from a large social-network using a special application to repeatedly query user data using the network's developer API. On the social network in question, we found that more than 99% of users had their information exposed to our application, making this approach superior to crawling all profiles visible to a public network, which is typically only 70-90% [14]. We crawled two sub-networks consisting of students from Stanford and Harvard universities, summarised in Table 1.

We note that our crawling method is impractical for crawling significant portions of the graph, because we were subject to rate-limiting and the number of friendship queries required was $O(n^2)$ as $n$, the number of users crawled, increased. Crawling public listings, as search engines are encouraged to do, is a much easier task.

# 3. STATISTICS

We studied five common graph metrics – vertex degree, dominating sets, betweenness centrality, shortest paths, and community detection – and found that even with a limited public view, an attacker is able to approximate them all with some success.

## 3.1 Degree

Information about the degree $d$ of nodes in the network is exposed in the sampled graph, particularly for low-degree

ferent sets of friends for the same user by sending requests from different IP addresses around the globe. We noticed a correlation between the set of friends shown and the geographic location of the requesting IP address. We suspect this is a marketing feature and not a security feature – showing a visitor a group of nearby people makes the site more appealing.

nodes. This is not surprising, because nodes with $d < k$ must be shown with fewer than $k$ links in their public listing, leaking their precise degree. For most social networks, however, finding the set of users with few friends is less interesting than finding the most popular users in the network, who are useful for marketing purposes.

Degree information is leaked for users with $d \geq k$ because they will likely be displayed in some of their friends' listings, meaning that there will be $\geq k$ edges for most nodes in the sampled graph. We can consider the sampled graph to be a directed graph with edges originating from the user whose public listing they were found in. The out-degree $d_{\text{out}}(n)$ of a node $n$ is the number of edges learned from its public listing, and the in-degree $d_{\text{in}}(n)$ is the number of edges learned from other public friend listings which include $n$. Note that we always have $d_{\text{out}}(n) \leq k$, but for a high degree node in the complete graph, we expect $d_{\text{in}}(n) > k$. We can make a naive approximation of $n$'s degree in the complete graph:

$$d_*(n) \approx \max[d_{\text{out}}(n), d_{\text{in}}(n) \cdot \frac{\bar{d}}{k}] \tag{1}$$

where $\bar{d}$ is the average degree of all nodes in the complete graph (we assume this can be found from public statistics). This method works because, for a node $n$ with $d(n)$ friends, $n$ will show up in each friend $m$'s listings with probability $\frac{k}{d(m)}$. Since we don't know $d(m)$, we can approximate with the average degree for the complete graph. This approximation, however, has an intrinsic bias, as nodes with many relatively low-degree friends will have an artificially high estimated degree. In a sampled graph, it is more meaningful for a node to have in-edges from a higher-degree node, conceptually similar to the PageRank algorithm [6]. Thus, we can improve our estimates by implementing an iterative algorithm: first initialise all estimates $d_0(n)$ to the naive estimate of equation 1, and then iteratively update:

$$d_i(n) \leftarrow \max[d_{\text{out}}(n), \sum_{m \in \text{in-sample}(n)} \frac{k}{d_{i-1}(m)}] \tag{2}$$

In order to keep the degree estimates bounded, it is necessary to normalise after each iteration. We found it works well to force the average of all estimated degrees to be our estimated average $\bar{d}$ for the complete graph.

To evaluate the performance of this approach, we define a cumulative degree function $D(x)$, which is the sum of the degrees of the $x$ highest-degree nodes. This is motivated by the likely use of degree information, to locate a set of very high-degree nodes in a network. In in Figure 2, there is a comparison of the growth of $D(x)$ when selecting the highest degree nodes given complete graph knowledge against our naive and iterated estimation algorithms. We also plot the growth of $D(x)$ if nodes are selected uniformly at random, which is the best possible approach with no graph information.

With $k = 8$, the iterative estimation method works very well; for the Harvard data set the highest-degree 1,000 nodes identified by this method have a cumulative degree of 407,746, compared with 452,886 using the complete graph, making our approximation 90% of optimal. We similarly found 89% accuracy in the Stanford network.

## 3.2 Dominating Sets

A more powerful concept than simply finding high-degree nodes is to compute a minimal *dominating set* for the graph. A dominating set is a set of nodes $N$, such that its *dominated set* $N \cup \text{friends}(N)$ is the complete set of users $V$. In the case of a social network, this is a set of people who are, collectively, friends with every person in the network. Marketers can target a small dominating set to indirectly reach the entire network. If an attacker can compromise a small dominating set of accounts, then the entire network is visible.

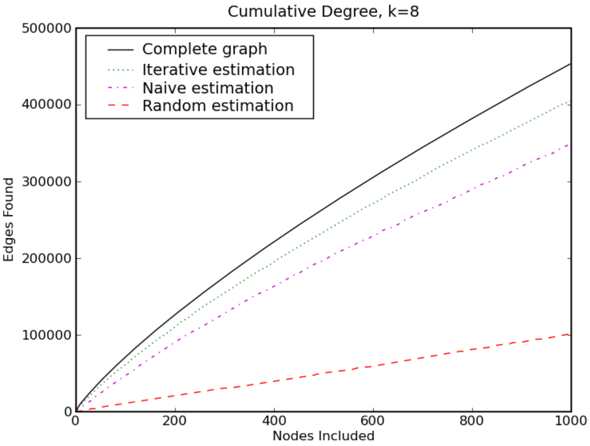


**Figure 2: Degree estimation methods**

Even given the complete graph, computing the minimum dominating set is an NP-complete problem [13]. The simple strategy of picking the highest-degree nodes can perform poorly in social networks, as many high-degree users with overlapping groups of friends will be selected which add relatively little coverage. However, a greedy algorithm which repeatedly selects the node which brings the most new nodes into the dominated set has been shown to perform well in practice [7].

To evaluate this greedy approach, we measured the growth of the dominated set as additional nodes are added to the dominating set, if nodes are selected using the complete graph or the sampled graph. The greedy algorithm performs very well given the sampled view. In Figure 3 we compared these two selection strategies against a "degree selection" strategy of always picking the next highest-degree node (given complete graph knowledge). The greedy algorithm outperforms this approach even with only sampled data, demonstrating that significant network information is leaked beyond an approximation of degrees[3].

Our experiments showed that very small sets exist which dominate most of the network, followed by a long tail of diminishing returns to dominate the entire network, consistent with previous research [9]. With complete graph knowledge, we found a set of 100 nodes which dominate 65.2% of the Harvard network, while we were able to find a set of 100 nodes giving 62.1% coverage using only the sampled graph,

[3] In fact, we were surprised to find that selecting the highest-degree nodes based on our naive degree-estimates from the sampled graph outperformed selecting based on actual degree! This is because a bias in favor of nodes with low-degree friends is useful for finding a dominating set, as the graph's fringes are reached more quickly.

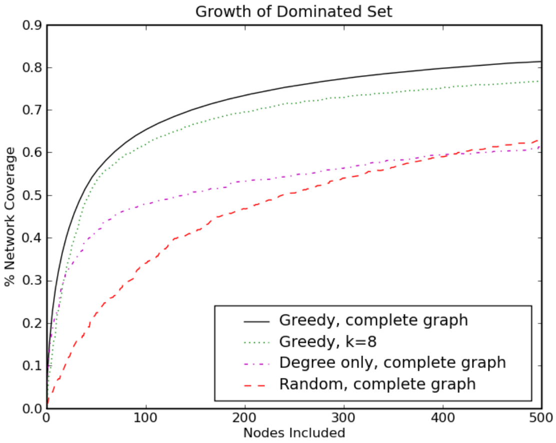


**Figure 3: Dominating set estimation**

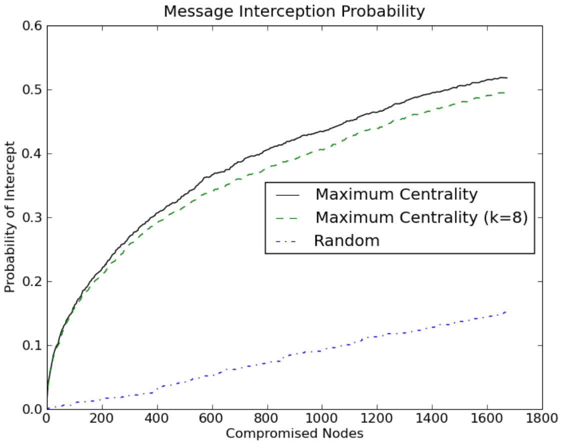


**Figure 4: Message interception**

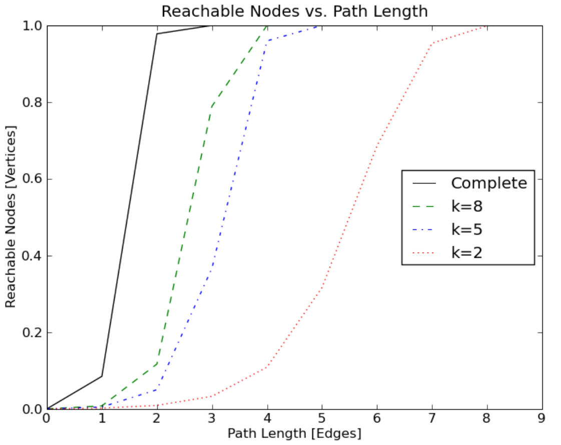


**Figure 5: Reachable nodes**

for 95% accuracy. Similarly, for Stanford, our computed 100-node dominating set had 94% of the optimal coverage.

## 3.3 Centrality and Message Interception

Another important metric used in the analysis of social networks is *centrality*, which is a measure of the importance of members of the network based on their position in the graph. We use the *betweenness centrality* metric, for which an efficient algorithm is described in [5]. Betweenness centrality is defined as:

$$C_B(v) = \sum_{s \neq v \neq t \in V} \frac{\sigma_{st}(v)}{\sigma_{st}} \qquad (3)$$

where $\sigma_{st}$ is the number of shortest paths from node $s$ to node $t$ and $\sigma_{st}(v)$ is the number of such paths which contain the node $v$. Thus, the higher a node's betweenness centrality, the more communication paths it is a part of.

A node with high betweenness centrality, therefore, is one which could intercept many messages travelling through the network. We simulated such interception occurring on the Stanford sub-network to determine the probability that an attacker controlling $N$ nodes could intercept a message sent from any node to any other node in the network via shortest-path social routes. The nodes designated as compromised are selected according to one of three policies: maximum centrality, maximum centrality for a sampled graph with $k = 8$, or random selection.

4 shows that, while randomly selecting nodes to compromise leads to a linear increase in intercepted traffic, a selective targeting of highly central nodes yields much better results. After compromising 10% of nodes in this sub-network, an attacker could intercept 15.2% of messages if she used random selection, but using centrality to direct the choice of nodes, the attacker can intercept as much as 51.9% of messages. Even if the attacker uses only the information available from a $k = 8$ public listing, she can still successfully intercept 49.8% of all messages – just 4% less than she could do with full centrality information.

## 3.4 Shortest Paths

We tested the extent that the *small world property* is maintained in a sampled graph by computing the minimum path length between every pair of nodes, using the Floyd-Warshall shortest path algorithm [11]. As this algorithm has a complexity of $O\left(|V|^3\right)$, where $V$ is the set of vertices in a graph, we only calculated shortest paths a subset of our example data. Within the 2,000 most popular Stanford users, the minimum-length shortest path was a single edge and the maximum-length path was just three edges. Figure 5 shows the effect of limiting friendship knowledge on reachability of nodes in the graph.

The maximum-length shortest path increases from 3 to 7 as we reduce visible friendships to 2 per person, but the graph remains fully connected. The average path length for Stanford was 1.94 edges, and with $k = 8$, it increased to just 3.09.

## 3.5 Community Detection

Given a complete graph, it is possible to automatically cluster the users into highly-connected subgroups which signify natural social groupings. Community structure in social graphs could be used to market products that other members of one's social clique have purchased, or to identify dis-

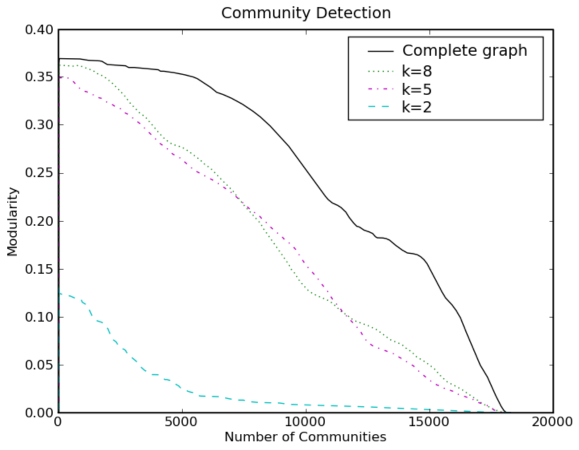


**Figure 6: Community Detection**

sident groups, among other applications. For our purposes, we will use attempt find communities with maximal *modularity* [16], a clustering metric for which efficient algorithms can partition realistically-sized social graphs with thousands of nodes.

Modularity is defined to be the number of edges which exist within communities beyond those that would be expected to occur randomly:

$$Q = \frac{1}{2m} \sum_{v,w} \left[ A_{vw} - \frac{d(v)d(w)}{2m} \right] \qquad (4)$$

where $m$ is the total number of edges in the graph, and $A_{vw} = 1$ if and only if $v$ and $w$ are connected. We implemented the greedy algorithm for community detection in large graphs described in [8]. The results on our sample graph were striking, as shown in Figure 6. With a sampling parameter of $k = 8$, we were able to divide the graph into communities nearly as well as using complete graph knowledge. Using sampled data, we divided the graph into 18,150 communities with a modularity of 0.341, 92% as high as the optimal 18,205 communities with a modularity of 0.369.

The algorithm produced a slightly different set of communities using the sampled graph, because a well-connected social network like a college campus contains many overlapping communities, but using modularity as our metric, the communities identified were almost as significant. Community detection worked nearly as well with k=5, but performance deteriorated significantly for k=2 and below.

# 4. RELATED WORK

Graph theory is a well-studied mathematical field; a good overview is available in [3]. Sociologists have long been interested in applying graph theory to social networks, the definitive work is [18]. Approximating information from a social graph when presented with incomplete knowledge, however is far less studied. The closest example we could find in the literature was a study which evaluated community detection given only the edges from a small subset of nodes [15]. This work used a different sampling strategy, however, and only aimed to classify nodes into one of two communities in a simulated graph. Another set of experiments [9] examined strategies for placing a small set of nodes under surveillance to gain coverage of a larger network, similar to our calculation of dominating sets. Related to our problem of limiting the usefulness of observable graph data is anonymising a social network for research purposes. It has been shown that, due to the unique structure of groups withing a social graph, it is often easy to de-anonymise a social graph by correlating it with known data [4].

# 5. CONCLUSIONS

We have examined the difficulty of computing graph statistics given a random sample of $k$ edges from each node, and found that many interesting properties can be accurately approximated. This has disturbing implications for online privacy, since leaking graph information enables transitive privacy loss: insecure friends' profiles can be correlated to a user with a private profile. Social network operators should be aware of the importance of protecting not just user profile data, but the structure of the social graph. In particular, they shouldn't assist data aggregators by giving away public listings.

# Acknowledgements

We would like to thank George Danezis, Claudia Diaz, Eiko Yoneki, and Joshua Batson, as well as anonymous reviewers, for helpful discussions about this research.

# 6. REFERENCES

[1] Facebook privacy policy. *http://www.facebook.com/policy.php* (2009).

[2] Acquisti, A., and Gross, R. Imagined Communities: Awareness, Information Sharing, and Privacy on the Facebook. In *Privacy Enhancing Technologies – LNCS 4258* (2006), Springer Berlin / Heildelberg, pp. 36–58.

[3] Albert, R., and Barabási, A.-L. Statistical mechanics of complex networks. *Rev. Mod. Phys. 74*, 1 (Jan 2002), 47–97.

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

[5] Brandes, U. A Faster Algorithm for Betweenness Centrality. *Journal of Mathematical Sociology 25*, 2 (2001), 163–177.

[6] Brin, S., and Page, L. The anatomy of a large-scale hypertextual web search engine. In *Seventh International World-Wide Web Conference (WWW 1998)* (1998).

[7] Chvatal, V. A Greedy Heuristic for the Set-Covering Problem. *Mathematics of Operations Research 4*, 3 (1979), 233–235.

[8] Clauset, A., Newman, M. E. J., and Moore, C. Finding community structure in very large networks. *Physical Review E 70*, 6 (Dec 2004), 066111.

[9] Danezis, G., and Wittneben, B. The economics of mass surveillance and the questionable value of anonymous communications. *WEIS: Workshop on the Economics of Information Security* (2006).

[10] Dwyer, C., Hiltz, S. R., and Passerini, K. Trust and privacy concern within social networking sites: A comparison of facebook and myspace. *America's Conference on Information Systems* (2007).

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

[12] Jagatic, T. N., Johnson, N. A., Jakobsson, M., and Menczer, F. Social phishing. *Commun. ACM 50*, 10 (2007), 94–100.

[13] Karp, R. Reducibility Among Combinatorial Problems. *Complexity of Computer Computations* (1972).

[14] Krishnamurthy, B., and Wills, C. E. Characterizing Privacy in Online Social Networks. In *Workshop on Online Social Networks – WOSN 2008* (2008), pp. 37 – 42.

[15] Nagaraja, S. The economics of covert community detection and hiding. *WEIS: Workshop on the Economics of Information Security* (2008).

[16] Newman, M. E. J., and Girvan, M. Finding and evaluating community structure in networks. *Physical Review E 69* (2004), 026113.

[17] Rosenblum, D. What Anyone Can Know: The Privacy Risks of Social Networking Sites. *IEEE Security & Privacy Magazine 5*, 3 (2007), 40.

[18] Wasserman, S., and Faust, K. *Social Network Analysis*. Cambridge University Press, 1994.

[19] Xu, W., Zhou, X., and Li, L. Inferring privacy information via social relations. *International Conference on Data Engineering* (2008).