Anonymizing Social Networks

Michael Hay, Gerome Miklau, David Jensen,
Philipp Weis, and Siddharth Srivastava
{mhay,miklau,jensen,pweis,siddharth}@cs.umass.edu
University of Massachusetts Amherst
Computer Science Department
Technical Report No. 07-19

March, 2007

Abstract

Advances in technology have made it possible to collect data about individuals and the connections between them, such as email correspondence and friendships. Agencies and researchers who have collected such social network data often have a compelling interest in allowing others to analyze the data. However, in many cases the data describes relationships that are private (e.g., email correspondence) and sharing the data in full can result in unacceptable disclosures. In this paper, we present a framework for assessing the privacy risk of sharing anonymized network data. This includes a model of adversary knowledge, for which we consider several variants and make connections to known graph theoretical results. On several real-world social networks, we show that simple anonymization techniques are inadequate, resulting in substantial breaches of privacy for even modestly informed adversaries. We propose a novel anonymization technique based on perturbing the network and demonstrate empirically that it leads to substantial reduction of the privacy threat. We also analyze the effect that anonymizing the network has on the utility of the data for social network analysis.

1 Introduction

A social network describes entities and connections between them. The entities are often individuals; they are connected by personal relationships, interactions, or flows of information. Social network analysis is concerned with uncovering patterns in the connections between entities. It has been widely applied to organizational networks to classify the influence or popularity of individuals and to detect collusion and fraud. Social network analysis can also be applied to study disease transmission in communities, the functioning of computer networks, and emergent behavior of physical and biological systems.

Technological advances have made it easier than ever to collect the electronic records that describe social networks. However, agencies and researchers who collect such data are often faced with a choice between two undesirable outcomes. They can publish data for others to analyze, even though that analysis will create severe privacy threats, or they can withhold data because of privacy concerns, even though that makes further analysis impossible.

For example, a large corpus of approximately 500,000 email messages, derived from the legal proceedings surrounding the 2001 bankruptcy of the Enron corporation and made public by the Federal Energy Regulatory Commission, has been frequently analyzed by researchers [19]. This data set has greatly aided research on email correspondence, organizational structure, and social network analysis, but it also has likely resulted in substantial privacy violations for individuals involved.

The alternative choice is exemplified by a study involving one of the authors [20] which analyzed a large data set about the behavior and affiliations of nearly all securities brokers in the United States. The study, conducted jointly with the National Association of Securities Dealers (NASD), which collects, manages, and uses this data in its regulatory role, discovered useful statistical models of broker behavior. Unfortunately, the data has not been released to other researchers, partially because of concerns about the privacy of individuals it identifies.
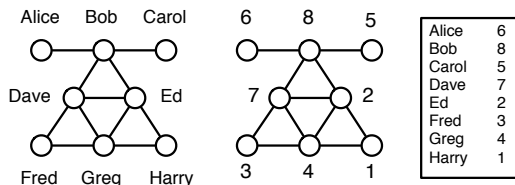
Figure 1: A social network, G; the naive anonymization of G; the anonymization mapping.

Similarly, researchers in the field of computer networking analyze internet topology, internet traffic and routing properties using network traces that can now be collected at line speeds at the gateways of institutions and by ISPs. These traces represent a social network where the entities are internet hosts and the existence of communication between hosts constitutes a relationship. However network traces (even with packet content removed) contain sensitive information because it is often possible to associate individuals with the hosts they use, and because traces contain information about web sites visited, and time stamps which indicate periods of activity. The challenges in protecting network trace data are being actively addressed by the research community [22].

Our goal is to enable the useful analysis of social network data while protecting the privacy of individuals. Much recent work has focused on managing the balance between privacy and utility in data publishing, but only for certain limited datasets. For example, k-anonymity [25], and its variants [12, 15, 17], are data perturbation techniques designed for tabular micro-data, which typically consists of a table of records each describing an entity. While useful for census databases and some medical information, these techniques cannot address the fundamental challenge of managing social network datasets. Other work has considered different kinds of data, such as item set transactions [10] and OLAP data [2]. However, a common assumption underlying all of these techniques is that the records are independent and can be anonymized (more or less) independently. In contrast, social network data forms a graph of relationships between entities. Existing tabular perturbation techniques are not equipped to operate over graphs—they will tend to ignore and destroy important structural properties. Likewise, graph structure and background knowledge combine to threaten privacy in new ways.

1.1 Publishing social network data

We model a social network as an undirected, unlabeled graph. The objective of the *data trustee* is to publish a version of the data that permits useful analysis while protecting the privacy of the entities represented. Invariably, a first step to preparing social network data for release is to remove identifying attributes such as name or social security number. In order to preserve node identity in the graph of relationships, synthetic identifiers are introduced to replace names. We refer to this procedure as the *naive anonymization* of a social network. A graph with named nodes is shown in Figure 1 along with its naive anonymization, in which synthetic identifiers have replaced names. The *anonymization mapping*, shown in (c), is a random, protected mapping.

Naive anonymization is a common practice. For example, the identifying attribute in network trace data is the IP address. Network traces are often released after encrypting the IP address. Naive anonymization meets the utility goals of the data trustee because most social network analysis can be performed in the absence of names and unique identifiers.

We focus on an adversary whose goal is to re-identify a known individual (e.g., Bob) in the naively-anonymized graph.[1] Synthetic identifiers reveal nothing about which node in the graph may be Bob. But by collecting information from external sources about an individual's relationships, an adversary may be able to re-identify individuals in the graph. For example, an adversary may learn that Bob has at least three neighbors. It follows that the node corresponding to Bob in the published graph must be 2, 4, 7 or 8.

---

[1]We discuss other possible threats in Section 6.

Thus, an entity's position in the graph of relationships acts as a quasi-identifier. The extent to which an individual can be distinguished using graphical position depends on the structural similarity of nodes in the graph, and also on the kind of background information an adversary can obtain.

Contributions

In this paper, we formalize the threat of re-identification and various kinds of adversary external information. We study a spectrum of outside information and show its relative power to re-identify individuals in a graph. The threat of re-identification has a connection to results in random graph theory. We contrast these theoretical results with empirical observations of re-identification attacks on several real-world social networks.

Protecting against the threat of re-identification presents novel challenges for graph structured data. In tabular data, identifying attributes can be generalized, suppressed or randomized easily, and the effects are largely restricted to the individual affected. It is much harder to generalize or perturb the structure around a node in a graph, and the impact of doing so can spread across the graph. We propose a novel alternative to naive anonymization based on random perturbation. Our perturbation techniques leave nodes unmodified but perform a sequence of random edge deletions and edge inserts. We show that this technique can significantly reduce the effectiveness of re-identification attacks by an adversary with acceptable distortion of the graph.

We evaluate all our techniques on real datasets drawn from the domains mentioned previously: an organization social network derived from the Enron dataset, a network trace graphs from a major university, and a scientific collaboration network.

2   Privacy in social network data

In this section we describe the threat of entity re-identification in social network data, and we explain the use of external information in identifying anonymized individuals. We develop the intuition behind achieving anonymity in a graph through structural similarity to others.

2.1   Preliminaries

A social network is an undirected graph $G = (V, E)$. The nodes of the graph, V, are named entities from the domain *dom*. In examples, we use $dom = \{$Alice, Bob, Carol . . . $\}$. The data trustee hides G, publishing in its place an anonymized graph. We begin by studying naive anonymization, in which the nodes of G are renamed and the structure of the graph is unmodified.

Definition 1 (Naive Anonymization) *The naive anonymization of a graph $G = (V, E)$ is an isomorphic graph, $G_{na} = (V_{na}, E_{na})$, defined by a random bijection $f : V \rightarrow V_{na}$. The edges of G_{na} are $E_{na} = \{(f(x), f(x'))|(x, x') \in E\}$.*

Faced with an anonymized graph, the adversary would like to associate an entity known to be present in G with its representative node in G_{na}. For an entity $x \in V$, called the *target*, its *candidate set* contains the nodes of G_{na} that could feasibly correspond to x. The candidate set is denoted cand(x). Since f is random, in the absence of other information, any node in G_{na} could correspond to the target node x. Thus, given an uninformed adversary, cand(x) = V_{na} for each target individual x.

The adversary may have access to external information about the entities in the graph and their relationships. We use structural queries to formalize the adversary's external information. A structural query, Q, is evaluated on a node x in a graph and returns information about the existence of neighbors of the node, or degree, or the structure of the subgraph in the vicinity of x. The following statements are each examples of external information about the graph in Figure 1: *Bob has two or more neighbors*; *Bob has exactly degree 4*; *Fred is connected to two nodes, each with degree 4*. We leave the precise form of structural queries vague for the moment; we will formalize two classes of queries in Sections 3.1 and 3.2.

The adversary does not have direct access to the original graph G, which is hidden. Instead the adversary relies on an external information source to provide answers to the evaluation of a Q on a limited set of nodes

of G. We assume knowledge gathered by the adversary is correct; that is, no spurious answers are provided to the adversary.

For a target node x, the adversary uses $Q(x)$ to refine the feasible candidate set and to deduce a probability distribution over matching candidates for x. Since G_{na} is published, the adversary can easily evaluate *any* structural query directly on G_{na}. The adversary computes the refined candidate set containing all nodes in the published graph G_{na} that are consistent with answers to the knowledge query on a target node.

Definition 2 (Candidate Set under Q) *For a query Q over a graph, the* candidate set of x w.r.t Q is $cand_Q(x) = \{y \in dom' \mid Q(x) = Q(y)\}$.

Total re-identification of a target x occurs when the adversary can identify a single candidate. Otherwise, partial re-identification has occurred. The seriousness of the disclosure depends on the set of candidates, and their likelihoods. In this discussion, and Sections 3 and 4, we make the natural assumption that the adversary has no other external information allowing him to choose among the candidates. Thus we assume each matching candidate node is equally likely. The probability, given Q, of candidate y for x is denoted $C_{Q,x}[y]$ and defined to be $1/|\text{cand}_Q(x)|$ for each $y \in \text{cand}_Q(x)$, and 0 otherwise.

To ensure anonymity we require that the adversary have a minimum level of uncertainty about the re-identification of any node in V. A successful anonymization is one that meets the following definition:

Definition 3 (K-Candidate Anonymity) *Let Q be a structural query. An anonymized graph satisfies K-Candidate Anonymity given Q if:*

$$\forall x \in V, \forall y \in cand_Q(x) : C_{Q,x}[y] \leq 1/k$$

This condition implies that there are at least k candidate nodes for any node x in the original data, and furthermore, that no candidate is highly likely. It is a generalization of k-anonymity [25]: if the probability distribution over candidates is uniform, this definition simply requires at least k candidates. Graph anonymization by edge perturbation, described in Section 5, results in a more complex probability distribution over candidates, for which this more general definition is required.

2.2   Achieving anonymity through structural similarity

Intuitively, nodes that look structurally similar may be indistinguishable to an adversary, in spite of external information. A strong form of structural similarity between nodes is *automorphic equivalence*. Two nodes $x, y \in V$ are automorphically equivalent (denoted $x \equiv_A y$) if there exists an isomorphism from the graph onto itself that maps x to y.

Example 2.1 *Fred and Harry are automorphically equivalent nodes in the graph of Figure 1. Bob and Ed are not automorphically equivalent: the subgraph around Bob is different from the subgraph around Ed and no isomorphism proving automorphic equivalence is possible.*

Automorphic equivalence induces a partitioning on V into sets whose members have identical structural properties. It follows that an adversary, even with exhaustive knowledge of a target node's structural position, cannot isolate an individual beyond the set of entities to which it is automorphically equivalent. We say that these nodes are *structurally indistinguishable* and observe that nodes in the graph achieve anonymity by being "hidden in the crowd" of its automorphic class members.

Some special graphs have large automorphic equivalence classes. For example, in a complete graph, or in a graph which forms a ring, all nodes are automorphically equivalent. But in most graphs we expect to find small automorphism classes, likely to be insufficient for protection against re-identification. We cite below theoretical evidence suggesting that automorphism classes will tend to have size 1 in random graphs, and we will test this empirically on social network datasets in Section 4.

Despite this negative result, automorphic equivalence is an extremely strong notion of structural similarity. In order to distinguish two nodes in different automorphic equivalence classes it may be necessary to use complete information about their position in the graph. Adversaries are unlikely to have access to such complete information. For example, if a weaker adversary only knows the degree of targeted nodes in the graph, Bob and Ed are indistinguishable (even though they are not automorphically equivalent). Thus for a weaker adversary, the candidate set for either Bob or Ed must contain both their representatives in G_{na}.
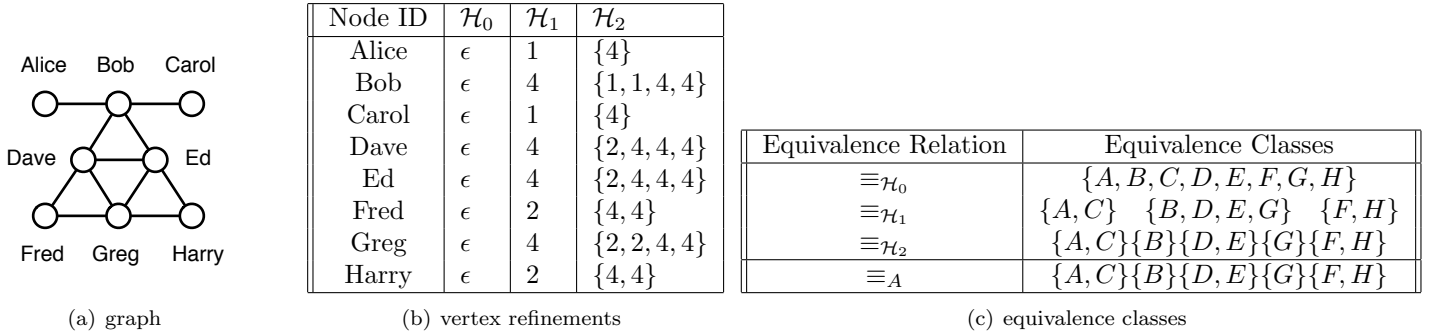
Node ID	$\mathcal{H}_0$	$\mathcal{H}_1$	$\mathcal{H}_2$
Alice	ϵ	1	$\{4\}$
Bob	ϵ	4	$\{1,1,4,4\}$
Carol	ϵ	1	$\{4\}$
Dave	ϵ	4	$\{2,4,4,4\}$
Ed	ϵ	4	$\{2,4,4,4\}$
Fred	ϵ	2	$\{4,4\}$
Greg	ϵ	4	$\{2,2,4,4\}$
Harry	ϵ	2	$\{4,4\}$

Equivalence Relation	Equivalence Classes
$\equiv_{\mathcal{H}_0}$	$\{A,B,C,D,E,F,G,H\}$
$\equiv_{\mathcal{H}_1}$	$\{A,C\}\quad\{B,D,E,G\}\quad\{F,H\}$
$\equiv_{\mathcal{H}_2}$	$\{A,C\}\{B\}\{D,E\}\{G\}\{F,H\}$
$\equiv_A$	$\{A,C\}\{B\}\{D,E\}\{G\}\{F,H\}$

(a) graph   (b) vertex refinements   (c) equivalence classes

Figure 2: (a) A sample graph; (b) External information consisting of vertex refinement queries $\mathcal{H}_1, \mathcal{H}_2$ and $\mathcal{H}_3$ computed for each individual in the graph; (c) The equivalence classes of nodes implied by vertex refinement. For the sample data, $\equiv_{\mathcal{H}_2}$, corresponds to automorphic equivalence, $\equiv_A$.

3   Adversary Knowledge

In this section we formalize two classes of knowledge queries describing the external information available to an adversary.

In practice, external information may be acquired in any number of ways. For organizational social networks, knowledge about existing relationships between known individuals may be publicly available or easily deducible. For networking data, an adversary may acquire an excerpt of a weblog for a particular machine listing hosts visited, or may attack an individual's machine to recover a list of visited websites. Active attacks involve an adversary capable of influencing the data from which the social network is constructed. This is a common vulnerability in network trace collection, for example, where the adversary can inject a sequence of identifiable network packets.

It is usually difficult for the data trustee to predict or bound an adversary's information gathering capabilities. Our goal is to develop a reasonable and conservative model of external information. In Section 3.1 we present a class of very expressive structural queries. These are not intended to model the real capabilities of an adversary, but instead to provide a precise and efficient way to capture structural knowledge of increasing diameter around a node. In Section 3.2, we present a less expressive class of queries, intended to model more accurately a realistic process of knowledge acquisition.

3.1   Vertex refinement queries

We define a class of queries, of increasing power, which report on the local structure of the graph around a node. These queries are inspired by iterative vertex refinement, a technique originally developed to efficiently test for the existence of graph isomorphisms [7, 26]. The weakest knowledge query, $\mathcal{H}_0$, simply returns the label of the node. (Since our graphs are unlabeled, $\mathcal{H}_0$ returns ϵ on all input nodes.) The queries are successively more descriptive: $\mathcal{H}_1(x)$ returns the degree of x, $\mathcal{H}_2(x)$ returns the list of each neighbors' degree, and so on. The queries can be defined iteratively, where $\mathcal{H}_i(x)$ returns the multiset of values which are the result of evaluating $\mathcal{H}_{i-1}$ on the set of nodes adjacent to x:

$$\mathcal{H}_i(x) = \{\mathcal{H}_{i-1}(z_1), \mathcal{H}_{i-1}(z_2) \ldots, \mathcal{H}_{i-1}(z_m)\}$$

where $z_1 \ldots z_m$ are the nodes adjacent to x.

Example 3.1 *Figure 2 contains the same graph from Figure 1 along with the computation of $\mathcal{H}_0$, $\mathcal{H}_1$, and $\mathcal{H}_2$ for each node. For example: $\mathcal{H}_0$ is uniformly ϵ. $\mathcal{H}_1(Bob) = \{\epsilon, \epsilon, \epsilon, \epsilon\}$, which we abbreviate in the table simply as 4. Using this abbreviation, $\mathcal{H}_2(Bob) = \{1, 1, 4, 4\}$ which represents Bob's neighbors' degrees.*

For each query $\mathcal{H}_i$ we define an equivalence relation on nodes in the graph in the natural way.

Definition 4 (Relative equivalence) *Two nodes x, y in a graph are equivalent relative to $\mathcal{H}_i$, denoted $x \equiv_{\mathcal{H}_i} y$, if and only if $\mathcal{H}_i(x) = \mathcal{H}_i(y)$.*

Example 3.2 *Figure 2(c) lists the equivalence classes of nodes according to relations $\equiv_{\mathcal{H}_0}, \equiv_{\mathcal{H}_1}$, and $\equiv_{\mathcal{H}_2}$. All nodes are equivalent relative to $\mathcal{H}_0$ (for an unlabeled graph). As i increases, the values for $\mathcal{H}_i$ contain successively more precise structural information about the node's position in the graph, and as a result, equivalence classes are divided.*

To an adversary limited to knowledge query $\mathcal{H}_i$, nodes equivalent with respect to $\mathcal{H}_i$ are indistinguishable. The following proposition formalizes this intuition:

Proposition 1 *Let $x, x' \in V$. If $x \equiv_{\mathcal{H}_i} x'$ then $cand_{\mathcal{H}_i}(x) = cand_{\mathcal{H}_i}(x')$.*

Iterative computation of $\mathcal{H}$ continues until no new vertices are distinguished. We call this query $\mathcal{H}^*$. In the example of Figure 2, $\mathcal{H}^* = \mathcal{H}_2$. The vertex refinement technique is the basis of efficient graph isomorphism algorithms which can be shown to work for almost all graphs [3]. In our setting, this means that equivalence under $\mathcal{H}^*$ is very likely to coincide with automorphic equivalence. Furthermore, for random graphs it has been shown that values for $\mathcal{H}_3$ uniquely identify nodes in the graph with very high likelihood.

Theorem 1 (Babai, Kucera [3]) *Let G be a random graph on n nodes with edge probability $p = 1/2$. The probability that there exist two nodes $x, y \in V$ such that $x \equiv_{\mathcal{H}_3} y$ is less than 2^{-cn}, for constant value $c > 0$.*

This theorem suggests that if an adversary is capable of collecting information sufficient to deduce $\mathcal{H}_3(x)$, that information will almost surely allow total re-identification of the node. This result is interesting because it suggests a sufficient condition for total re-identification in a graph. Of course, social networks are generally very different than random graphs. We are not aware of results similar to this theorem that apply to models of random graphs more appropriate to social networks (e.g., preferential attachment [5]). In contrast to the above Theorem, in Section 4 we find that in real social networks there are nodes not uniquely identified under $\mathcal{H}_3$. Overall however, $\mathcal{H}_3$ is a powerful knowledge query that results in high risk of re-identification for a large percentage of nodes.

3.2 Subgraph knowledge queries

Vertex refinement queries are a concise way to describe locally expanding structural queries, and also relate nicely to results on random graphs. However, as a model of realistic adversary knowledge they have two significant drawbacks. First, they always provide complete information about the nodes adjacent to the target. For example, $\mathcal{H}_1(x)$ returns the *exact* number of neighbors of x. Often an adversary can obtain only a partial list of neighbors of a targeted node. That is, the adversary knows of three people connected to Bob, but cannot conclude Bob has no additional connections. Second, $\mathcal{H}$ queries can describe arbitrarily large subgraphs around a node if that node is highly connected. For example, if $\mathcal{H}_1(x) = 100$, the adversary learns about a large subgraph in G, while $\mathcal{H}_1(y) = 2$ provides much less information. The index of the $\mathcal{H}$ query is therefore a coarse and somewhat unreliable measure of the information contained in the query result.

As an alternative, we consider a class of queries which assert the existence of a subgraph around the target node. We measure the descriptive power of a query by counting the number of edges in the described subgraph; we refer to these as *edge facts*. For example, Figure 3 illustrates three subgraphs centered around *Bob*. The first simply asserts that *Bob* has (at least) three distinct neighbors, the second describes a tree of nodes near *Bob*, and the third relates nearby nodes in a subgraph. These informal query patterns use 3, 4, and 4 edge facts, respectively.

We do not model an adversary capable of constructing and evaluating arbitrary subgraph queries. Instead, we assume the adversary is capable of gathering some fixed number of edge facts focused around the target x. By exploring the neighborhood of x, the adversary learns the existence of a subgraph around x representing partial information about the structure around x. The existence of this subgraph can be expressed as a query, and we model the adversary's knowledge by granting the answer to some such query.

Naturally, for a fixed number of edge facts there are many subgraph queries that are true around a node x. These correspond to different strategies of knowledge acquisition that could be employed by the adversary. In testing the distinguishing power of subgraph queries in Section 4, we assume the adversary follows a breadth-first strategy (other strategies are certainly possible). Thus with a fixed budget of edge facts, the adversary will learn about successive neighbors of Bob until all four have been discovered. He may then explore the neighbors of one of these 4 neighbors.
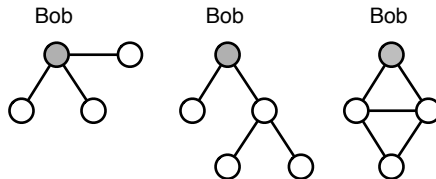
Figure 3: Three instances of the partial information about the entity *Bob* that can be expressed as a subgraph knowledge query.

An important defining characteristic of this form of background knowledge is whether the adversary can learn that the collection of neighbors of a node is a complete list. We allow this, but only when the adversary has exhaustively explored the neighborhood of a node. Thus, if the adversary has 3 edge facts he learns that Bob has 3 or more neighbors; with 4 edge facts he learns that Bob has exactly 4 neighbors. We study the impact on re-identification of this choice in Section 4.

Another important factor is whether the adversary can recognize individuals discovered through different paths as identical. This determines whether the adversary learns of the existence of a subgraph around the target or only learns about the tree expansion of the subgraph around the target. We focus on the tree version; we did not observe a significant empirical difference in re-identification using the graph version.

Note that it is possible to acquire, using a subgraph query, the same information available through a vertex refinement query. However, depending on the degree of the target node (and its neighbors' degrees, etc.), it may take many edge facts to represent $\mathcal{H}_i$. So the relationship between edge facts and the i in $\mathcal{H}_i$ is data dependent and node specific.

Finally, we note that without exact constraints on node degree, a subgraph query can be represented as a conjunctive query with disequalities. The number of edge facts used corresponds to the number of subgoals in the query. The graph version is expressed using additional equality conditions on variables as compared with the tree version. The adversary learns simply that the target node is in the answer set. When the query is evaluated on the anonymized graph, it returns the candidate set. With exact degree constraints the queries are no longer conjunctive, but require universal quantification. We omit the details of the formal specification of subgraph queries.

4   Re-identification in real social networks

In this section we evaluate the impact of external information on the adversary's ability to re-identify individuals in real social networks.

4.1   Experiments and Datasets

We study four networked datasets, drawn from diverse domains. We simulate an adversary attempting to re-identify individuals present in the network using external information as formalized in Section 3. For each dataset, we consider each node in turn as a target. We assume the adversary computes a vertex refinement or subgraph knowledge query on that node, and then compute the corresponding candidate set for that node. We report the distribution of candidate set sizes across the population of nodes to characterize how many nodes are protected and how many are identifiable.

Our datasets are derived as follows:

- The **Hep-Th** database is based on data from the arXiv archive and the Stanford Linear Accelerator Center SPIRES-HEP database. It describes papers and authors in theoretical high-energy physics. We extracted a subset of the authors and considered them linked if they wrote at least two papers together.

Statistic	Data Set			
			Net-	Net-
	Hep-Th	Enron	trace	common
Nodes (total)	2671	117	4248	201
Nodes (component)	2510	111	4213	187
Edges (total)	4888	290	5531	5399
Edges (component)	4737	287	5507	5398
Minimum degree	1	1	1	1
Maximum degree	36	20	1656	157
Median degree	2	5	1	71
Average degree	3.77	5.17	2.61	57.73
Avg clust. coeff	.25	.34	0	.76
Diameter	14	9	10	4

Figure 4: Summary statistics for the four social networks studied. The *clustering coefficient* of a node is the fraction of all pairs of neighbors who are linked. The *diameter* of a graph is the longest shortest path between any two nodes.

- The **Enron** dataset is derived from a corpus of email sent to and from managers at Enron Corporation, made public by the Federal Energy Regulatory Commission during its investigation of the company. We consider two individuals connected if they corresponded at least 5 times.

- The **Net-trace** dataset was derived from an IP-level network trace collected at a major university. The trace monitors traffic at the gateway, so it produces a bi-partite graph between IP addresses internal to the institution, and external IP addresses. We restricted the trace to 201 internal addresses from a single campus department and the 4047 external addresses to which at least 20 packets were sent on port 80 (http traffic).

- The **Net-common** dataset was derived from the **Net-trace**. Its nodes represent only the 201 internal IP addresses. There are links between two internal IP addresses if they visited at least one common external address in **Net-trace**. This graph is very dense.

All datasets have undirected edges, with self-loops removed. We eliminated a small percentage of disconnected nodes in each dataset, focusing on the largest connected component in the graph. Detailed statistics for the datasets are shown in Figure 4.

4.2 Re-identification under vertex refinement queries

Recall from Section 3 that nodes contained in the same candidate set for knowledge $\mathcal{H}_i$ share the same value for $\mathcal{H}_i$, are indistinguishable according to $\mathcal{H}_i$, and are therefore protected if the candidate set size is sufficiently large.

Figure 5 is an overview of the likelihood of re-identification under $\mathcal{H}_1, \mathcal{H}_2, \mathcal{H}_3$ and $\mathcal{H}_4$ knowledge queries. For each $\mathcal{H}_i$, the graph reports on the percentage of nodes whose candidate sets fall into the following buckets: [1] , [2, 4], [5, 10], [11, 20], [21, ∞]. The total percentage of nodes with candidate sets in these buckets is 1. Nodes with candidate set size 1 have been uniquely identified, and nodes with candidate sets between 2 and 4 are at high risk for re-identification. Nodes are at fairly low risk for re-identification if there are more than 20 nodes in their candidate set. Each $\mathcal{H}_i$ is represented as a different point on the x-axis.

Figure 5 shows that for the **Hep-Th** data, $\mathcal{H}_1$ leaves nearly all nodes at low risk for re-identification, and it requires $\mathcal{H}_3$ knowledge to uniquely re-identify a majority of nodes. For **Enron** under $\mathcal{H}_1$ about 15% of the nodes have candidate sets smaller than 5, while only 19% are protected in candidate sets greater than 20. Under $\mathcal{H}_2$, re-identification jumps dramatically so that virtually all nodes have candidate sets less than 5.

The **Net-trace** and **Net-common** traces are also quite different from one another. **Net-trace** has very few identified nodes under $\mathcal{H}_1$ while in **Net-common** 60% nodes have candidate sets of size less than 5.
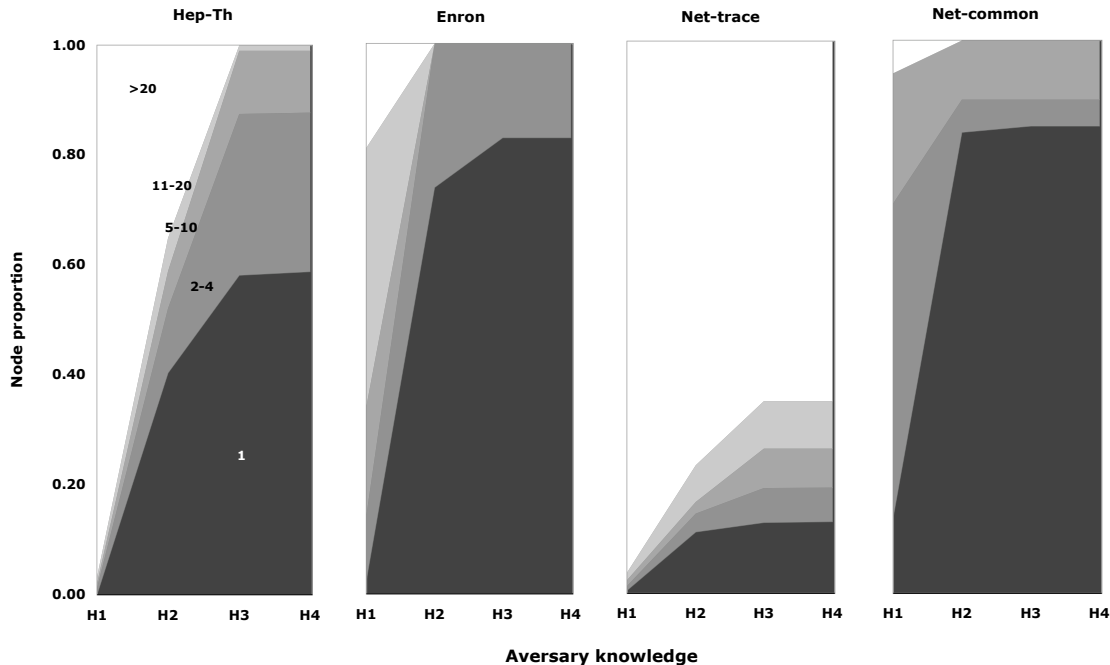
Figure 5: The distribution of candidate sets for four social network datasets under vertex refinement knowledge $\mathcal{H}_1, \mathcal{H}_2, \mathcal{H}_3$, and $\mathcal{H}_4$. The trend lines show the percentage of nodes whose candidate sets have sizes in the following buckets: [1] (black), $[2, 4], [5, 10], [11, 20], [21, \infty]$ (white). Nodes with candidate sets of size 1 have been uniquely re-identified (in black); nodes with candidates sets greater than 20 are well-protected (in white).

A natural precondition for publication is a very low percentage of high-risk nodes under a reasonable assumption about adversary knowledge. Two datasets meet that requirement for $\mathcal{H}_1$ (**Hep-Th** and **Net-trace**), but no datasets meet that requirement for $\mathcal{H}_2$.

Overall, we observe that there can be significant variance across different datasets in their vulnerability to different adversary knowledge. In addition, it is clear that a result similar to that stated in Theorem 1 is unlikely to hold on social networks: in each of these datasets, a substantial number of individuals are *not* uniquely identified under $\mathcal{H}_3$. Further, re-identification tends to stabilize after $\mathcal{H}_3$—more information in the form of $\mathcal{H}_4$ does not lead to an observable increase in re-identification in any dataset.

4.3 Re-identification under subgraph exploration queries

Recall from Section 3 that we also model an adversary exploring the local graph around a known individual, and measure that knowledge by counting the number of edge facts acquired. In the present experiments we simulate an adversary who explores in a breadth-first manner using a bounded number of edge facts.

Figure 6 describes the evolution of candidate set sizes as edge facts are acquired, covering subgraph queries consisting of 0 to 50 edge facts. Most datasets can tolerate subgraph queries of size 10 with modest impact on re-identification. **Enron** is the exception, for which 10 edge facts can uniquely re-identify about 11% of the nodes, and leaves over 45% of the nodes with dangerously small candidate sets under 5.

Figures 5 and 6 can be readily compared. In the case of **Hep-Th**, **Enron**, and **Net-trace** we see a strong correspondence between the two models of adversary knowledge, although subgraph queries on this scale (less than 50 edge facts) cause uniformly lower re-identification rates. The exception is the **Net-common** dataset. It is highly identified by vertex refinement queries, but is resilient against subgraph knowledge
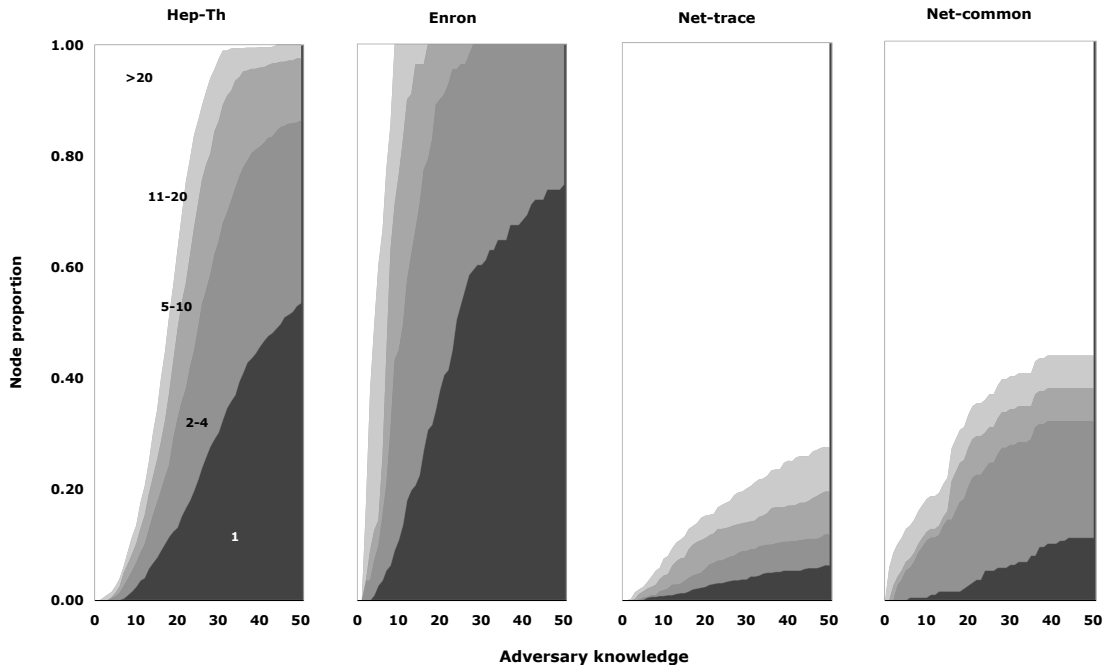
Figure 6: Evolution of candidate set sizes for subgraph queries composed of up to 50 edge facts. The trend lines show the percentage of nodes whose candidate sets fall into each of the following buckets: $[1]$, $[2, 4]$, between $[5, 10]$, $[11, 20]$, $[21, \infty]$. Nodes with candidate sets of size 1 have been uniquely re-identified (in black); nodes with candidates sets greater than 20 are well-protected (in white).

queries. In this network, the median degree is 71, so an adversary with 50 edge facts still cannot exhaust neighbors of most nodes. In contrast, with vertex refinement queries, the adversary learns the exact degree of all nodes.

The results in Figure 6 are for subgraph knowledge queries that allow the adversary to learn the exact degree of a node with sufficient edge facts. Without this ability the adversary has uncertainty about the existence of other undiscovered neighbors in the vicinity of a node, which provides only a lower bound on degree. We found a very substantial decrease in node re-identification without exact degree, shown in Figure 7 for the **Enron** dataset. With exact degree, uniquely identified nodes increase rapidly with increasing edge facts until approximately 75% of the nodes have been uniquely identified under 50 edge facts. Without exact degree, very few nodes are uniquely identified even with 50 edge facts. This suggests that adversaries capable of acquiring information on exact node degree deserve special precautions. On the other hand, if such information is not readily available to an adversary, the dataset can tolerate large graph exploration queries.

A metric for structural re-identification risk   As a practical matter, we believe these results confirm that the $\mathcal{H}_i$ queries can serve as a useful metric for evaluating the likelihood of node re-identification on a dataset being considered for publication. The complexity of computing $\mathcal{H}^*$ is linear in the number of edges in the graph, and is therefore efficient even for large datasets. The result would measure the number of individuals at risk for re-identification under an appropriate $\mathcal{H}_i$, and reports on the structural diversity of the population. It is not clear how a data trustee could evaluate the risk of structural re-identification in the absence of this technique.
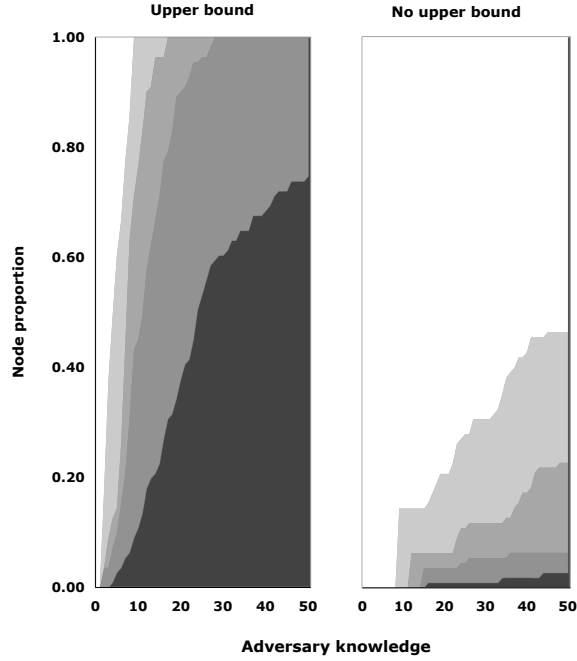
Figure 7: A comparison of the candidate set sizes for subgraph queries that allow an adversary to learn exact degree constraints (left) and subgraph queries without exact degree constraints (right).

5 Random perturbation of social network data

In this section we describe a graph transformation technique which can protect against re-identification by distorting structural features. The perturbation of the graph forces the adversary to attempt re-identification in many equally-likely possible worlds, increasing his uncertainty about the true identity of individuals. Our goal is for legitimate users to analyze the perturbed graph "as-is" without undue information loss. After describing our perturbation scheme we study both the increase in anonymity, and the decrease in utility that result from perturbation.

5.1 Graph perturbation

Our perturbation procedure is applied to the graph after it has been naively anonymized. The new graph $G_p = (V_p, E_p)$ is constructed from G_{na} through a sequence of m edge deletions followed by m edge insertions. Deletions are chosen uniformly at random from the set of all existent edges in G_{na}. Insertions are chosen uniformly at random from the set of all non-existent edges of the interim graph. The nodes are unchanged, so $V_p = V_{na}$. The process of perturbation and the perturbation parameter m are assumed to be publicly known. Note that if $m = |E_{na}|$, G_p is simply a random graph, which clearly contains no information about G beyond its edge density. We intend m to be a small fraction of $|E_{na}|$.

The adversary attempts to re-identify individuals using external information, as before. However perturbation of the graph means the adversary cannot simply exclude from the candidate set nodes that do not match the structural properties of the target. The adversary must consider the set of *possible worlds* implied by G_p and m. Informally, the set of possible worlds consists of all graphs that could result in G_p under m perturbations. Using $\overline{E_{na}}$ to refer to all edges not present in G_{na}, we have the following formal definition:

Definition 5 (Possible Worlds) *The set of possible worlds of G_p under m perturbations, denoted $\mathcal{W}_m(G_p)$, is the set of all graphs g over dom' such that there exists $E^- \subseteq E_{na}$ and there exists $E^+ \subseteq \overline{E_{na}} \cup E^-$, $|E^-| = |E^+| = m$, such that $E_p = E_{na} - E^- \cup E^+$.*

The candidate set of a target node x includes all nodes $y \in G_p$ such that y is a candidate in *some* possible world. Any node that would be a candidate for x under naive anonymization will also be a candidate under graph perturbation (since G_{na} is a possible world of G_p). We expect the candidate sets to grow large with increased perturbation. However, it is no longer sufficient to consider only the size of the candidate sets because there could be many extremely low-probability candidates. Recall that our K-Candidate Anonymity condition (Definition 3) does not permit any candidate to have a probability greater than $1/k$.

Because the deleted and inserted edges are chosen uniformly at random, the adversary must assume that each possible world is equally likely. For each $g \in \mathcal{W}$,

$$ \Pr(g) = \frac{1}{\binom{|E_{na})|}{m}\binom{|\overline{E_{na}}|+m}{m}} $$

where the denominator is the total number of possible worlds for m perturbations. The probability associated with a candidate y is the probability of choosing a possible world in which y is candidate.

Definition 6 (Candidate Probability in G_p) *Let Q be a knowledge query, $x \in V, y \in V_p$. The probability of candidate y for target x is:*

$$ C_{Q,x}[y] \propto \sum_{\{g|Q(x)=Q(y)\}} \Pr(g) $$

We assume a powerful adversary capable of computing the candidate set probability distribution consistent with his external information. However, we note that this computation is equivalent to computing a query answer over a probabilistic database and is likely to be intractable. For example, some forms of the subgraph knowledge described in Section 2 can be expressed as a conjunctive query with disequality and computation of the candidate sets is equivalent to evaluating such a query on the anonymized database. Evaluating a query over a probabilistic database was shown to be #P-hard for a wide class of conjunctive queries, even under the relatively simple probability distributions such as that described above [8]. Nevertheless, it is not clear how to use the intractability of this problem as a guarantee that individuals will remain anonymous, since these are worst-case asymptotic bounds for computing exact answers. They do not guarantee the adversary will not succeed in computing answers, or sufficiently accurate estimates, for selected individuals.

5.2 Re-identification resistance from perturbation

In this section we study the increase in anonymity that results from perturbation, as a function of m. We focus on node re-identification using $\mathcal{H}_1$, for which it is possible to compute candidate set probability distributions directly. We focus on the **Enron** dataset, which was significanltly vulnerable to $\mathcal{H}_1$ knowledge.

Figure 8 shows the impact on privacy as a result of increased perturbation of the **Enron** dataset. The perturbation rate, m, is on the x-axis, represented as a percentage of the total number of edges in the graph. For each rate of perturbation, we measure the *equivalent* candidate set size for each node. This measure is defined as $\lfloor \min_y 1/C_{Q,x}[y] \rfloor$, which corresponds to the maximum k for which the K-Candidate Anonymity holds. For example, if the adversary's most likely candidate has probability $1/4$, then the equivalent candidate set size is 4. We see the basic trend we expect: candidate set sizes shrink quickly with increased perturbation.

The number of high risk nodes (the black area) decreases quickly with perturbation, with significant impact achieved in less than a 5% perturbation rate. However, significant increase in the number of well-protected nodes (the white area) is not achieved until 5% to 10% perturbation rate is reached.

5.3 Utility of perturbed data

The perturbed graph G_p would be published for analysis by legitimate users. Although it is possible for users to analyze the set of possible worlds, this is likely to be intractable, as mentioned above. Instead, our goal is for legitimate users to analyze the published social network "as-is". Thus, we hope to maintain acceptable information loss in G_p.

Our ultimate goal is to allow effective analysis of social networks that have been anonymized. Rather than choose an arbitrary analysis technique, we study a set of common graph metrics. For each node, we measure:
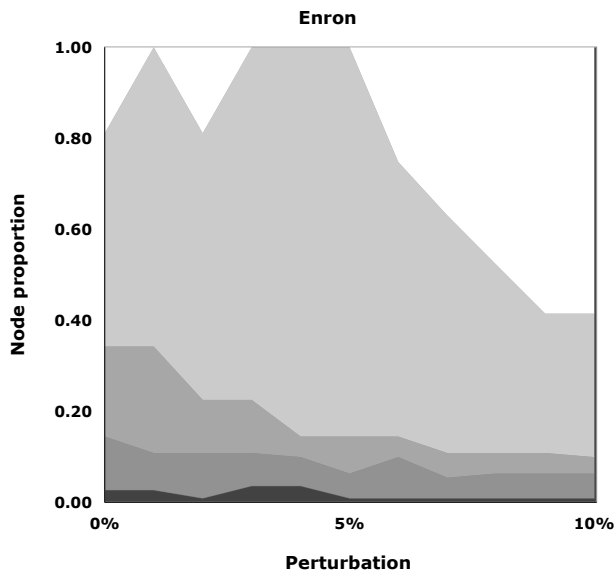
Figure 8: The effect of perturbation on privacy for the Enron dataset. The amount of random edge perturbation varies along the horizontal axis from 0% (the original graph) to 10%. Privacy is measured in terms of the *equivalent* candidate set size — the largest k for which K-Candidate Anonymity holds. The trend lines show the percentage of nodes whose equivalent candidate set size falls into each of the following buckets: [1] (black), [2, 4], [5, 10], [11, 20], [21, ∞] (white).

closeness centrality (average shortest path from the node to every other node) and betweenness centrality (proportion of all shortest paths through the node). We study the path length distribution, computed from the shortest path between each pair of nodes. For the graph as a whole we study the degree distribution and the diameter (the maximum shortest path between any two nodes). We believe reasonable preservation of these measures is a sufficient condition for accurate analysis of a published social network.

Recall that as m approaches $|E_{na}|$ (the total number of edges), G_p is simply a random graph with density matching the original graph. On a random graph with fixed edge density many of the measures above have well-known expected values. For example, the average clustering coefficient of a random graph is inversely proportional to the number of nodes [21]. As we increase the level of perturbation, the graph measures will converge on these expected values, constituting information loss. The fundamental question is how quickly information is lost as perturbation increases. More specifically, can we preserve sufficient accuracy in graph measures for perturbation of m between 5% and 10% of $|E_{na}|$, which we have shown above to offer substantial gains in anonymity?

Figure 9 presents graph measures for the original graph, 5% perturbation and 10% perturbation for the **Enron** dataset. While the perturbed graphs are often distinct from a completely random graph, the information loss after a perturbation of 10% of the edges appears to be substantial. For example, at this level of perturbation, clustering coefficient has moved one-third the distance to the random-graph value, both closeness and betweenness centrality have moved half the distance, and path length has converged to the random-graph value.

5.4   Model based perturbation

Perturbation is a promising technique for enhancing anonymity, but the utility results above show that accuracy of graph measures may be threatened. A strategy for maintaining accuracy under perturbation is for the data trustee to derive a statistical model of the original data, and to use that model to "bias" the random perturbation towards those that respect properties of the graph. Model-driven data perturbation is well-known in the statistical database community [24]. Recently, link prediction models [16] have been proposed that, after a learning phase, predict edges likely to be created as a graph evolves. These models

Measure	Enron			
	Original	Perturbed 5%	Perturbed 10%	Random (100%)
Degree	5.0	4.5	4.6	5.0
Diameter	9.0	8.7	7.6	6.1
Path length	4.0	3.2	3.0	3.0
Closeness	0.276	0.293	0.304	0.337
Betweenness	0.005	0.009	0.010	0.014
Clust. Coeff.	0.286	0.242	0.191	0.000

Figure 9: Summary of the effect of perturbation on key graph measures, described in 5.3, for the **Enron** dataset. Random corresponds to 100% perturbation. For each graph metric, we measure the median value (except for diameter which is a single value for a graph). The numbers reported above are averaged over 10 graphs at each level of perturbation.

could be easily incorporated into the perturbation scheme described above. Presumably, with model-driven edge deletions and insertions, even randomly chosen from a set of likely candidates, the perturbed graph will not converge on a random graph but will preserve more graph properties of interest. The effect on anonymity is less clear. On the one hand, the model-driven perturbations will preserve more structure, possibly making it easier for an adversary to re-identify nodes. On the other hand, the model will be imperfect and thus some of the graph structure will be perceived by the model as outlying. As the amount of perurbation increases, such outlying regions will 'regress toward the mean,' providing greater anonymity.

6 Discussion

Research into the unintended disclosures from publicly accessible social networks is in an early stage. We have focused here on what we believe to be one of the most basic and distinctive challenges—understanding the extent to which graph structure acts as an identifier and the cost in accuracy required to obscure this identifier using perturbation. In this section we briefly describe alternatives to assumptions we have made and promising directions for future study.

6.1 Alternative disclosures

We studied an adversary whose goal was re-identification of a known individual in an anonymized trace. Other threats are also important, and some may exist even if re-identification is prevented. For example, an adversary may wish to know whether Alice and Bob are linked directly in the graph. Even if the adversary is faced with large candidate sets which hide the identity of both Alice and Bob, their candidate node sets may be highly connected so that Alice and Bob will be connected with high probability. This risk is conceptually similar to the lack of diversity in the sensitive attributes of k-anonymized tabular data [17].

We have also assumed the adversary targets one node at a time. That is, re-identification is focused on node x, and is considered independently of attempts to reidentify x'. Targeting sets of nodes simultaneously can have some subtle consequences. For example, if $\text{cand}(x) = \{y, y'\}$ and $\text{cand}(x') = \{y'\}$ then x is uniquely identified since x' can only correspond to y'. These overlapping but non-identical candidate sets are impossible for queries $\mathcal{H}_i$ (see Section 2). But they are possible for other knowledge queries that do not provide complete information. The general observation is that for some inference processes by the adversary, $\text{cand}(x, x')$ (the feasible assignments to the pair of targets x, x') is not equal to $\text{cand}(x) \times \text{cand}(x')$ (the cross product of the candidate sets for the individuals). Against an adversary seeking to re-identify a group of individuals, this aspect of the reasoning process must be taken into account. An analogous issue for tabular anonymization has been considered in recent work [18].

14

6.2   Alternative forms of external information

We have treated our social network as unlabeled, though there are descriptive attributes of individuals that are clearly important for analysis. Both vertex refinement queries and subgraph queries can be extended to model a combination of structural and attribute information. The discriminating power of the queries increases accordingly.

The degree to which attribute knowledge helps to re-identify individuals depends first, on the selectivity of published attributes, and second, on the correlation between attributes and structure. It is natural to imagine adopting tabular generalization and suppression techniques from the k-anonymity literature to bin node attributes, making them less distinguishing. An interesting direction for future work is to integrate attribute anonymization with structural anonymization.

Throughout the discussion we have assumed all adversary knowledge was *correct* (meaning only true facts about the original graph are provided to the adversary) and often *complete* (meaning there are no missing facts). Weaker models of an adversary may be more appropriate in some settings. We showed (in Section 4.3) that completeness of external information significantly increases the power of the adversary. We would also like to consider approximate graph information, in which an adversary learns an estimate of a true fact, but does not have precise information.

In general, there are many interesting questions concerning optimal information gathering strategies for adversaries. Given limited time or resources, should an adversary try to acquire structural information around a node or attribute information? When gathering structural information by discovering relationships, how should the adversary select the next edge for exploration? Understanding the relative distinguishing power of these strategies is important for data owners who may be in a position to control aspects of publicly available information used to identify individuals in published social networks.

7   Related Work

The work closest to ours is a forthcoming publication by Backstrom et al. [4]. The authors look at social network data that has been naively anonymized, but consider different attacks than those in our work. Their main result regards an active attack, where the adversary does not have knowledge of the graph, but is capable of adding nodes and edges *before* the graph is anonymized. The adversary's strategy is to construct a highly distinguishable subgraph with edges to a set of target nodes, and then to re-identify the subgraph (and consequently the targets) in the published network. The authors provide an algorithm for constructing a subgraph that will be distinguished with high probability.

Much of the work in anonymization has focused on tabular microdata, a database consisting of a single table where each record corresponds to an individual. K-anonymity, introduced in [25], protects tabular microdata against linking attacks in which an adversary uses external information to re-identify individuals. There has been considerable follow-on work on algorithms for constructing k-anonymous tables [6, 13]; on improving the utility of the anonymized data [12, 14]; and on subtler aspects of disclosure, such as inferring properties of the target even without perfect re-identification [17, 18]. While a graph can be represented in a single table (e.g., an adjacency matrix or edge relation), it does not have the same semantics as tabular microdata (where the records are independent). Applying tabular anonymization to a tabular representation of a graph will either destroy the graph or fail to provide anonymity.

Recently, there have been extensions of anonymization techniques to more complex datasets. In [9], the authors present a mechanism, along with strong privacy guarantees, that can be used to share any kind of data, including network data. Rather than publish a perturbed version of the data, they propose an interactive mechanism where the private data is shared with analysts through a privacy-preserving database access protocol. While an interactive approach might be well-suited for some kinds of analysis, it has some disadvantages. The analyst can only ask queries, limiting common social network analysis practices such as visualization and clustering, and he/she can only ask a limited number of queries, limiting data exploration. In [1], the authors consider anonymizing data in which the individuals are interrelated through employment histories. They use a model-based approach, generating partially synthetic records. The privacy risks are not explicitly considered; the emphasis is on utility, generating replicates that preserve the statistical correlations of the original data.

Achieving privacy through random perturbation has been an area of active work. The author of [23] empirically evaluates the disclosure risk of random perturbation for tabular micro-data, using a framework that is similar to ours. The author observes that while privacy is improved for most records, outliers remain distinguishable. The first perturbation approach that presents a strong guarantee of privacy protection was presented in [10]. The authors consider a scenario where the data collector is untrusted, so the data is perturbed prior to being collected. This technique has been extended to OLAP data [2]. Whether such approaches can be extended to social network data remains an area of future work.

8   Conclusion

We have studied the extent to which structural properties of a node can serve as a basis of re-identification in anonymized social networks. We proposed two models for external information which can serve as a practical basis on which to evaluate the risk of re-identification in real datasets. Finally, we have studied the tradeoff between increased anonymity and information loss as a result of random perturbation.

References

[1] J. M. Abowd and S. Woodcock. Disclosure limitation in longitudinal linked data. In *Confidentiality, Disclosure, and Data Access*, 2001.

[2] R. Agrawal, R. Srikant, and D. Thomas. Privacy preserving OLAP. In *SIGMOD*, 2005.

[3] L. Babai and L. Kucera. Canonical labeling of graphs in linear average time. In *Foundations of Computer Science*, pages 39–46, 1979.

[4] L. Backstrom, C. Dwork, and J. Kleinberg. Wherefore art thou R3579X? anonymized social networks hidden patterns and structural steganography. In *World Wide Web Conference (Forthcoming)*, 2007.

[5] A. Barabási and R. Albert. Emergence of Scaling in Random Networks. *Science*, 286, 1999.

[6] R. Bayardo and R. Agrawal. Data privacy through optimal k-anonymization. In *ICDE*, 2005.

[7] D. G. Corneil and C. C. Gotlieb. An efficient algorithm for graph isomorphism. *J. ACM*, 17(1):51–64, 1970.

[8] N. Dalvi and D. Suciu. Efficient query evaluation on probabilistic databases. In *Conference on Very Large Databases*, pages 864–875, 2004.

[9] C. Dwork, F. McSherry, K. Nissim, and A. Smith. Calibrating noise to sensitivity in private data analysis. In *Third Theory of Cryptography Conference*, 2006.

[10] A. Evfimievski, J. Gehrke, and R. Srikant. Limiting privacy breaches in privacy preserving data mining. In *PODS*, 2003.

[11] M. Hay, G. Miklau, D. Jensen, D. Towsley, and P. Weis. Resisting structural re-identification in anonymized social networks. In *VLDB*, 2008.

[12] D. Kifer and J. Gehrke. Injecting utility into anonymized datasets. In *SIGMOD*, 2006.

[13] K. LeFevre, D. DeWitt, and R. Ramakrishnan. Mondrian multidimensional k-anonymity. *ICDE*, 2006.

[14] K. LeFevre, D. DeWitt, and R. Ramakrishnan. Workload-aware anonymization. In *KDD*, 2006.

[15] N. Li and T. Li. t-closeness : privacy beyond k-anonymity and l-diversity. In *ICDE*, 2007.

[16] D. Liben-Nowell and J. Kleinberg. The link prediction problem for social networks. In *CIKM*, 2003.

[17] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam. ℓ-diversity: privacy beyond k-anonymity. *ICDE*, 2006.

[18] D. Martin, D. Kifer, A. Machanavajjhala, J. Gehrke, and J. Halpern. Worst-case background knowledge. ICDE, 2007.

[19] A. McCallum, A. Corrada-Emmanuel, and X. Wang. Topic and role discovery in social networks. In *IJCAI*, 2005.

[20] J. Neville, O. Simsek, D. Jensen, J. Komoroske, K. Palmer, and H. Goldberg. Using relational knowledge discovery to prevent securities fraud. In *KDD*, 2005.

[21] M. E. J. Newman, D. J. Watts, and S. H. Strogatz. Random graph models of social networks. *PNAS*, 99(90001):2566–2572, 2002.

[22] R. Pang, M. Allman, V. Paxson, and J. Lee. The devil and packet trace anonymization. *SIGCOMM Comput. Commun. Rev.*, 2006.

[23] J. P. Reiter. Estimating risks of identification disclosure in microdata. *Journal of the American Statistical Association*, 100:1103–1112(10), December 2005.

[24] D. B. Rubin. Discussion: Statistical disclosure limitation. *Journal of Official Statistics*, 9(2):461–468, 1993.

[25] L. Sweeney. k-anonymity: a model for protecting privacy. *International journal of uncertainty, fuzziness, and knowledge-based systems*, 2002.

[26] J. R. Ullmann. An algorithm for subgraph isomorphism. *J. ACM*, 23(1):31–42, 1976.