Knowl Inf Syst (2011) 28:645–663
DOI 10.1007/s10115-010-0353-5

REGULAR PAPER

On link privacy in randomizing social networks

Xiaowei Ying · Xintao Wu

Received: 8 August 2009 / Revised: 3 March 2010 / Accepted: 19 October 2010 /
Published online: 12 November 2010

Abstract Many applications of social networks require relationship anonymity due to the sensitive, stigmatizing, or confidential nature of relationship. Recent work showed that the simple technique of anonymizing graphs by replacing the identifying information of the nodes with random IDs does not guarantee privacy since the identification of the nodes can be seriously jeopardized by applying subgraph queries. In this paper, we investigate how well an edge-based graph randomization approach can protect sensitive links. We show via theoretical studies and empirical evaluations that various similarity measures can be exploited by attackers to significantly improve their confidence and accuracy of predicted sensitive links between nodes with high similarity values. We also compare our similarity measure-based prediction methods with the low-rank approximation-based prediction in this paper.

Keywords Link privacy · Randomization · Social networks · Similarity measures

1 Introduction

Social networks are of significant importance in various application domains such as marketing, psychology, epidemiology, and homeland security. Many applications of social networks such as anonymous Web browsing require relationship anonymity due to the sensitive, stigmatizing, or confidential nature of relationship. For example, most people prefer to conceal the truth regarding their illegal or unethical behaviors which are customarily disapproved of by society.

One natural approach is to publishing a node-anonymized version of the network that permits useful analysis without disclosing the identity of the individuals represented by the nodes. The recent work [4,11] pointed out that this simple technique of anonymizing graphs by replacing the identifying information of the nodes with random IDs does not

X. Ying · X. Wu (✉)
Department of Software and Information Systems, University of North Carolina at Charlotte, Charlotte, NC 28223, USA
e-mail: xwu@uncc.edu

guarantee privacy since the identification of the vertices can be seriously jeopardized by applying subgraph queries. Another approach is to randomizing edges to protect sensitive links [10,11,17,25,29]. For example, we can remove some true edges and/or add some false edges. After the edge randomization, the randomized graph is expected to be different from the original one. As a result, the true sensitive or confidential relationship will not be much disclosed even if the identification of the vertices is achieved by attackers.

We will explore how well the edge randomization can protect those sensitive links. In Ref. [25], the authors preliminarily investigated the relationship between the amount of randomization and the attacker's ability to infer the presence of a link and presented a randomization strategy that can preserve the spectral properties (and utility) of the graph. In Ref. [24], the authors investigated how well an edge-based graph randomization approach can protect node identities and sensitive links when adversaries have one specific type of background knowledge (i.e., knowing the degrees of target individuals). However, the effect on privacy due to randomization in [24], [25] was quantified by considering only the magnitude information of randomization. It has been well known that graph topological features have close relations with the existence of links and various proximity measures have been exploited to predict the existence of a future link [16]. In this paper, we will investigate formally how attackers may exploit proximity measure values (derived from the released randomized graph) to breach link privacy. We exclude identity privacy from the scope of this paper.

Privacy of a sensitive link is jeopardized if attackers' confidence of prediction is higher than some tolerated threshold or is significantly greater than the prior belief (without the exploit of the released randomized data). Hence it is of great importance for data owners to be aware of potential attacks and quantify the magnitude of perturbation to better protect sensitive links. We would point out that our problem of attacking methods on a randomized graph is different from the classic link prediction problem investigated in [16]. The classic link prediction focuses on network evolution models and is to predict the existence of a future link between two nodes given a snapshot of a current social network. The change due to randomization is different with that due to network evolutions. Nevertheless, various graph proximity measures used in the classic link prediction could be used by attackers.

2 Related work

Social network analysis has increasing interest in the database, data mining, and theory communities. The privacy concerns associated with data analysis over social networks have incurred recent research works [4,5,10,11,17,25,26,28–30].

In Ref. [4], the authors described a family of attacks such that an adversary can learn whether edges exist or not between specific targeted pairs of nodes from node-anonymized social networks. Similarly in Refs. [10,11], the authors further observed that the structure of the graph itself (e.g., the degree of the nodes or the degree of the node's neighbors) determines the extent to which an individual in the network can be distinguished.

In Ref. [17], the authors investigated how to modify a graph via a set of edge addition (or deletion) operations in order to construct a new K-degree anonymous graph, in which every node has the same degree with at least $K-1$ other nodes. In Ref. [29], the authors anonymized the graph by generalizing node labels and inserting edges until each neighborhood is indistinguishable to at least $K-1$ others. In Ref. [30], the authors proposed a systematic model, called K-automorphic network, to protect against multiple structural attacks and

developed an algorithm that ensures K-automorphism. In Refs. [5,28], the authors applied a structural anonymization approach called *edge generalization* that consists of collapsing clusters together with their component nodes' structure, rather than add or delete edges from the social network data set. Although the above proposed approaches would preserve privacy to some extent, however, it is not clear how useful the anonymized graph is since many topological features may be lost.

In Refs. [9,26], the authors studied the problem of how to generate a synthetic graph matching various properties of a real social network in addition to a given degree sequence. They investigated a switching-based algorithm for generating synthetic graphs whose feature values are within a precise range of those of the original graph. In Ref. [26], the authors also studied how adversaries exploit the released graph as well as feature constraints to breach link privacy. The adversary can calculate the posterior probability of existence of a link by exploiting the ensemble of graphs with the given degree sequence and the prescribed feature constraints. However, the attacking model in [26] was based on the probability of existence of a link across all possible graphs in the graph space. In this paper, the attacking model is to exploit the relationship between existence of a link and the similarity measure values of node pairs in one released randomized graph.

Beyond the ongoing privacy preserving social network analysis which mainly focus on un-weighted social networks, in Refs. [6,18], the authors studied the situations in which the network edges as well as the corresponding weights are considered to be private. The authors in Ref. [18] developed privacy preserving strategies that can not only keep a close shortest path length and exactly the same shortest path for certain selected paths but also maximize the weight privacy preservation while the authors in Ref. [6] proposed an edge weight anonymization approach via linear programming, which preserves properties of the graph that are expressible as linear functions of the edge weights.

A large amount of work on privacy preserving data mining for numerical data has been reported in recent years. The random noise addition methods have been well investigated to prevent the disclosure of confidential individual values while preserving general patterns and rules for numerical data (e.g., [2,3]). Most recently, the authors in Ref. [22] proposed a hybrid multi-group approach for privacy preserving data mining. They combined the randomization approach and the secure multi-party computation approach to balance the accuracy and efficiency constraints. The authors in Ref. [7] investigated the problem of the sensitive knowledge hiding in large transactional databases without hiding of nonsensitive patterns in the sanitized data. The authors in Ref. [20] presented an approach for privacy preserving distributed model-based classifier training to support customizable privacy modeling and protection. Point-wise reconstruction methods in numerical settings have also been well developed in privacy preserving data mining community. A spectral filtering-based reconstruction method was first proposed by Kargupta et al. [13,14] to reconstruct original data values from the perturbed data. Similar methods (e.g., PCA-based reconstruction method [12], SVD-based reconstruction method [8]) were also investigated. All methods exploited spectral properties of the correlated data to remove the noise from the perturbed one. One natural idea is to implement a similar low-rank approximation-based prediction method on networked data. We also compare our similarity measure–based prediction methods with the low-rank approximation-based prediction in this paper.

3 Link privacy analysis

A network $G(n, m)$ is a set of n nodes connected by a set of m links. The network considered here is binary, symmetric, connected, and without self-loops. Let $A = (a_{ij})_{n\times n}$ be its

adjacency matrix, $a_{ij} = 1$ if node i and j are connected and $a_{ij} = 0$ otherwise. $\widetilde{G}$ is the randomized graph obtained by randomly adding k false edges followed by deleting k true edges. This strategy keeps the total number of edges in the original graph unchanged. We denote $\tilde{A} = \left(\tilde{a}_{ij}\right)_{n\times n}$ be the adjacency matrix of $\widetilde{G}$.

When it comes to link privacy, it is usually $a_{ij} = 1$ that people want to hide, not $a_{ij} = 0$ and attackers are capable of calculating posterior probabilities. Formally, we use $P(a_{ij} = 1)$ to denote the users' prior belief about the event of $a_{ij} = 1$ and use $P(a_{ij} = 1|\widetilde{G})$ to denote its posterior belief about $a_{ij} = 1$. The released graph $\widetilde{G}$ is regarded as jeopardizing the privacy if $P(a_{ij} = 1|\widetilde{G}) > P(a_{ij} = 1)$.

In [25], we preliminarily investigated the relationship between the amount of randomization and the attacker's ability to infer the presence of a link. The results are shown as follows. When the attacker knows only parameter m and n, the prior belief is

$$P(a_{ij} = 1) = \frac{2m}{n(n-1)}. \tag{1}$$

With the released graph and perturbation parameter k, the posterior belief is

$$P(a_{ij} = 1|\tilde{a}_{ij} = 1) = \frac{m-k}{m}, \quad P(a_{ij} = 1|\tilde{a}_{ij} = 0) = \frac{k}{\binom{n}{2} - m} \tag{2}$$

Equation (2) is based on the Addition/Deletion without replacement.[1]

In this paper, we further investigate whether topological features of the released network can be exploited by attackers to breach the link privacy. More specifically, we focus on to what extent a given sensitive relationship can be breached by attackers who exploit proximity measure values of node pairs. Proximity measures have been shown to be effective in the classic link prediction problem (i.e., predicting the future existence of links among nodes given a snapshot of a current graph). However, link prediction in our context is to predict the likelihood of existence of original links from the randomized graph. This is challenging since the proximity measure values calculated from the randomized graph can be varied from those of the original graph. In Sect. 3.1, we empirically show the close relationship between various similarity measures of node pairs and probability of link existence between them. In Sect. 3.2, we conduct theoretical studies and quantify how much the posterior belief can be enhanced by exploiting those similarity measures.

3.1 Existence of a link versus similarity measure

Let m_{ij} be a similarity measure on node pair (i, j) in graph G (a larger value of m_{ij} indicates that nodes i and j are more similar). We apply four similarity measures in this paper. The first one is the number of common neighbors:

$$CN_{ij} = \sum_{k=1}^{n} a_{ik}a_{kj}.$$

[1] Refer to [25] for the Addition/Deletion with replacement. For large graphs, the difference between the above is small.

The second one is the Adamic/Adar measure [1], which is the weighted number of common neighbors. The weights are assigned based on the information theory:

$$Ad_{ij} = \sum_{k=1}^{n} \frac{1}{\log d_k} a_{ik} a_{kj},$$

where d_k is the degree of node k. The third one is the Katz measure, which is a weighted sum of the number of paths in the graph that connect two nodes. Shorter paths are given the larger weight with parameter β [15]:

$$K_{ij} = \sum_{k=1}^{\infty} \beta^k P_{ij}^{(k)},$$

where $P_{ij}^{(k)}$ denotes the number of paths from i to j with length equal to k while β is a damping factor. In this paper, we take $\beta = 0.1$. The fourth one is the commute time CT_{ij}, which is the expected steps of random walks from i to j and back to i. The commute time is a dissimilarity measure: dissimilar nodes have large CT values. The commute time can be calculated through the eigenvalues and eigenvectors of the graph's normal matrix [19]. Let $N = D^{-\frac{1}{2}} A D^{-\frac{1}{2}}$ where $D = \text{diag}\{d_1, d_2, \ldots, d_n\}$.$N$ has n real eigenvalues: $\nu_1 \geq \nu_2 \geq \nu_3 \cdots \nu_n$ with corresponding eigenvectors $\boldsymbol{z}_1, \boldsymbol{z}_2, \ldots, \boldsymbol{z}_n$, and let z_{ki} denote the k'th entry of $\boldsymbol{z}_i$. Then

$$CT_{ij} = 2m \sum_{k=2}^{n} \frac{1}{1 - \nu_k} \left(\frac{z_{ki}}{\sqrt{d_i}} - \frac{z_{kj}}{\sqrt{d_j}} \right)^2.$$

Let $\rho(\Omega)$ denote the proportion of true edges in the set of node pairs Ω:

$$\rho(\Omega) = \frac{1}{|\Omega|} \sum_{(i,j) \in \Omega} a_{ij},$$

where $|\Omega|$ denotes the number of elements in set Ω. Let $S_x = \{(i, j) : m_{ij} = x\}$ denote the set of all node pairs with the similarity measure $m_{ij} = x$. Hence, $\rho(S_x)$ denotes the proportion of true edges in the S_x, which can be considered as the probability of existence of a link between node pair (i, j) in S_x. Next, we empirically show how $\rho(S_x)$ varies with x in real social networks.

Figure 1 shows how the proportions of true edges in S_x are varied with measure values x in terms of three similarity measures (Common neighbors, Katz, and Adamic/Adar) and one dissimilarity measure (Commute time) in the US political books network (*polbooks*). The *polbooks* network[2] contains 105 nodes and 441 edges, and nodes represent books about US politics sold by the online bookseller Amazon.com while edges represent frequent co-purchasing of books by the same buyers on Amazon. We can observe that $\rho(S_x)$ increases with the similarity measure value x and decreases with the dissimilarity measure x. In other words, the probability that $a_{ij} = 1$ is positively correlated with similarity measures (e.g., Common neighbors) and is negatively correlated with dissimilarity measures (e.g., Commute time).

We then perturbed the *polbooks* network by adding 200 false edges and deleting 200 true edges. From the perturbed graph $\widetilde{G}$, we define $\widetilde{S}_x = \{(i, j) : \tilde{m}_{ij} = x\}$ as the set of node pairs with similarity measure $\tilde{m}_{ij} = x$. Figure 2 shows how the proportions of true edges

[2] http://www-personal.umich.edu/~mejn/netdata/.

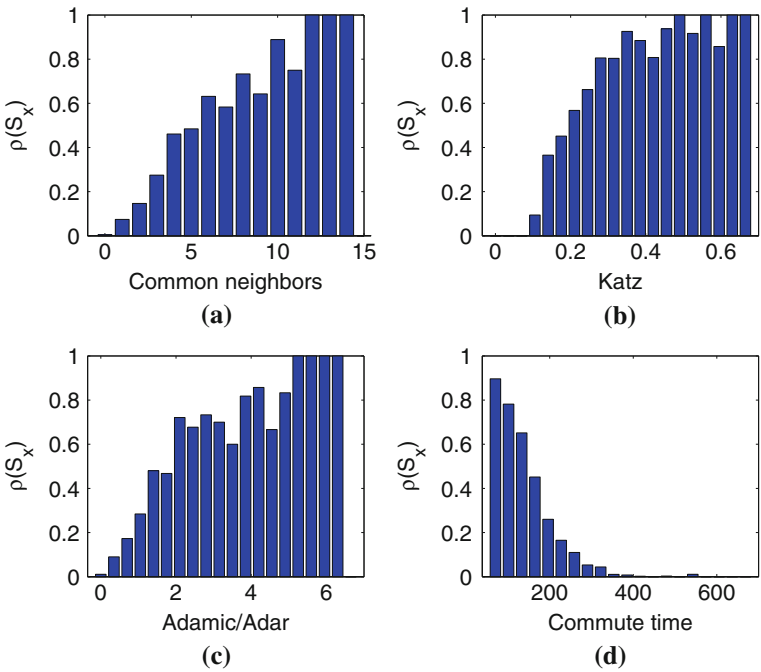

Fig. 1 Similarity/dissimilarity measure versus the prob. of true edges in the original graph ($\rho(S_x)$) for polbooks. Common neighbors, Katz, and Adamic/Adar are similarity measures whereas Commute time is a dissimilarity measure. **a** Common neighbors; **b** Katz; **c** Adamic/Adar; **d** Commute time

in $\widetilde{S}_x$ (i.e., the probability of existence of a link) are varied with similarity (or dissimilarity) measure values x in the randomized *polbooks* network. We can observe that the same pattern still holds even if the randomized graph itself is quite different from the original one (200 false edges out of 441 edges). In the next section, we will show how attackers exploit $\tilde{m}_{ij}$ in the perturbed graph $\widetilde{G}$ to improve their posterior belief on existence of a true link between nodes (i, j) in the original graph.

In Ref. [16], the authors compute the similarity measures of all the node pairs, and regard the node pair with high similarity has greater probability to be connected in the future. The strategy is consistent with our observation.

3.2 Link prediction by exploiting similarity measure

In this section, we quantify how much the posterior belief can be enhanced by exploiting similarity measure between two node (i, j) in the randomized graph. We present our quantification in a series of results and leave detailed proofs in Appendix.

Recall the randomization strategy is to randomly add k false edges followed by deleting k true edges. In other words, every true link is to be deleted independently with probability p_1 and every non-existing link is to be added independently with probability p_2. We can easily derive $p_1 = k/m$ and $p_2 = k/\left[\binom{n}{2} - m\right]$.

Let $\tilde{m}_{ij}$ denote the similarity measure of node i and j in $\widetilde{G}$. We define $\widetilde{S}_x = \{(i, j) : \tilde{m}_{ij} = x\}$ as the set of node pairs with $\tilde{m}_{ij} = x$ in the perturbed graph. Then we have

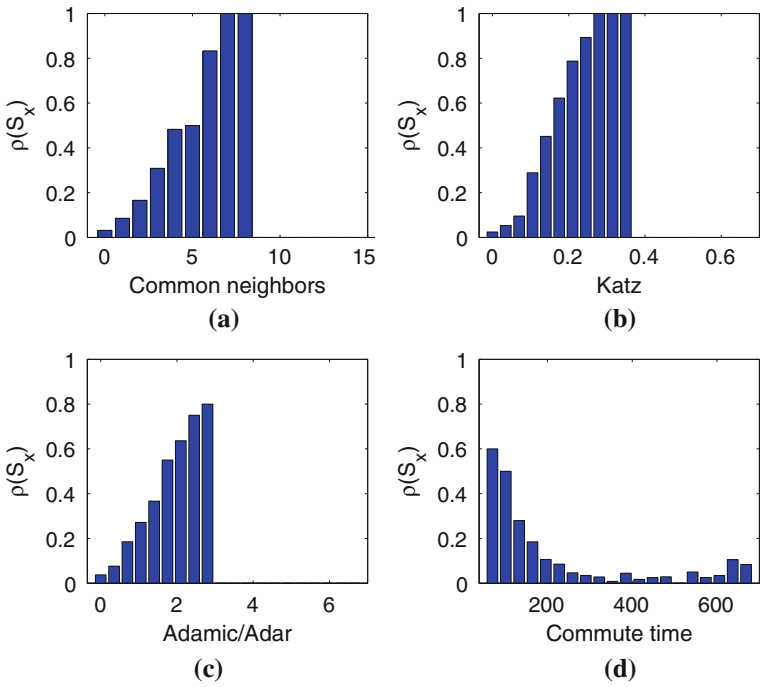

Fig. 2 Similarity/dissimilarity measure versus the prob. of true edges in the randomized graph ($\rho(\widetilde{S}_x)$) for polbooks. Common neighbors, Katz, and Adamic/Adar are similarity measures whereas Commute time is a dissimilarity measure. **a** Common neighbors; **b** Katz; **c** Adamic/Adar; **d** Commute time

$P(a_{ij} = 1|\tilde{m}_{ij} = x) = \rho(\widetilde{S}_x)$, and $P(a_{ij} = 0|\tilde{m}_{ij} = x) = 1 - \rho(\widetilde{S}_x)$. Recall that $\rho(\widetilde{S}_x)$ denotes the proportion of true edges in the set $\widetilde{S}_x$ derived from the perturbed graph. Also notice that $P(\tilde{a}_{ij} = 1|a_{ij} = 1) = 1 - p_1$ and $P(\tilde{a}_{ij} = 1|a_{ij} = 0) = p_2$. With the Bayes' theorem, the posterior belief is then given by

$$P(a_{ij} = 1|\tilde{a}_{ij} = 1, \tilde{m}_{ij} = x) = \frac{(1-p_1)\rho(\widetilde{S}_x)}{(1-p_1)\rho(\widetilde{S}_x)+p_2[1-\rho(\widetilde{S}_x)]}, \tag{3}$$

$$P(a_{ij} = 1|\tilde{a}_{ij} = 0, \tilde{m}_{ij} = x) = \frac{p_1\rho(\widetilde{S}_x)}{p_1\rho(\widetilde{S}_x)+(1-p_2)[1-\rho(\widetilde{S}_x)]}. \tag{4}$$

Equation (3) (Eq. (4)) shows the enhanced posterior belief that an observed (missing) edge (i, j) in the $\tilde{G}$ is a true edge in G. The following property shows that the event of an observed link $\tilde{a}_{ij} = 1$ usually has more indications to be a true link than that of $\tilde{a}_{ij} = 0$.

Property 1 *Let r denote the sparse ratio of the graph,* $r = m/\binom{n}{2}$. *If* $k \leq (1-r)m$, *given a fixed x, we have the following inequality stands:*

$$P(a_{ij} = 1|\tilde{a}_{ij} = 1, \tilde{m}_{ij} = x) \geq P(a_{ij} = 1|\tilde{a}_{ij} = 0, \tilde{m}_{ij} = x). \tag{5}$$

Many real-world social networks are very sparse ($r \approx 0$). Hence, $k \leq (1-r)m$ is usually satisfied. We thus focus on the risk of the released links, $P(a_{ij} = 1|\tilde{a}_{ij} = 1, \tilde{m}_{ij} = x)$.

One issue here is that attackers cannot know the proportion of true edges in $\widetilde{S}_x$ from the perturbed graph. What they can know actually is the proportion of observed edges in $\widetilde{S}_x$.

Our next result shows the maximum likelihood estimate of $\rho(\widetilde{S}_x)$ can be derived from the proportion of observed edges in $\widetilde{S}_x$.

Result 1 Given the perturbed graph and a fixed x, define $\widetilde{S}_x^1 = \widetilde{S}_x \cap \widetilde{E} = \{(i, j) : \tilde{a}_{ij} = 1, \tilde{m}_{ij} = x\}$. Assume $p_1 + p_2 \neq 1$, then the maximum likelihood estimator (MLE) of $\rho(\widetilde{S}_x)$ is given by

$$\hat{\rho}\left(\widetilde{S}_x\right) = \frac{|\widetilde{S}_x^1|/|\widetilde{S}_x| - p_2}{1 - p_1 - p_2}, \tag{6}$$

and the MLE is unbiased.

By replacing $\rho(\widetilde{S}_x)$ in Eq. (3) with $\hat{\rho}(\widetilde{S}_x)$ (shown in Eq. (6)), we have derived our enhanced posterior belief $P(a_{ij} = 1|\tilde{a}_{ij} = 1, \tilde{m}_{ij} = x)$. Attackers may simply calculate the posterior belief of all node pairs in the perturbed graph and choose top-t node pairs as predicted candidate links.

For those similarity measures with continuous ranges (e.g., commute time), the number of node pairs with similarity measure equal exactly to x is usually small. In practice, we can apply histogram approximation by partitioning the value of the similarity measure: $x_0 \leq x_1 \leq \cdots \leq x_i \leq \cdots$, and for $x \in [x_{i-1}, x_i)$

$$\frac{|\widetilde{S}_x^1|}{|\widetilde{S}_x|} = \frac{|\{(i, j) : \tilde{a}_{ij} = 1, \tilde{m}_{ij} = x \in [x_{i-1}, x_i)\}|}{|\{(i, j) : \tilde{m}_{ij} = x \in [x_{i-1}, x_i)\}|}.$$

A probably more statistically preferred method is to use the kernel estimator:

$$\frac{|\widetilde{S}_x^1|}{|\widetilde{S}_x|} = \frac{\sum_{i<j} \tilde{a}_{ij} K[(x - m_{ij})/h]}{\sum_{i<j} K[(x - m_{ij})/h]},$$

where $K(x)$ is the p.d.f. of the standard normal distribution and h is the parameter controlling the smoothness.

We would emphasize that our enhanced posterior belief $P(a_{ij} = 1|\tilde{a}_{ij} = 1, \tilde{m}_{ij} = x)$ more accurately reflect the existence of a true link than the posterior belief $P(a_{ij} = 1|\tilde{a}_{ij} = 1)$ without exploiting the similarity measure derived in previous work [25]. We can see that $P(a_{ij} = 1|\tilde{a}_{ij} = 1)$ (shown in Eq. (2)) is the same for all observed links. On the contrary, our enhanced posterior belief $P(a_{ij} = 1|\tilde{a}_{ij} = 1, \tilde{m}_{ij} = x)$ tends to be larger for those observed links with higher similarity values and tends to be smaller for links with lower similarity values. Hence, it can more accurately reflect the existence of true links. We show our theoretical explanations in Results 2 and 3 and will compare the precisions of top-t predicted links derived from these two posterior beliefs in our empirical evaluations.

Result 2 $P(a_{ij} = 1|\tilde{a}_{ij} = 1, \tilde{m}_{ij} = x)$ is an increasing function of $\rho(\widetilde{S}_x)$, and when $\rho(\widetilde{S}_x) \geq \frac{p_2}{p_1+p_2}$, we have the following inequality stands:

$$P(a_{ij} = 1|\tilde{a}_{ij} = 1, \tilde{m}_{ij} = x) \geq P(a_{ij} = 1|\tilde{a}_{ij} = 1). \tag{7}$$

Our next result shows more clearly the relationship between a priori belief (Eq. (1)), posterior belief without exploiting similarity measures (Eq. (2)), and our enhanced posterior belief with exploiting similarity measures (Eqs. (3, 4)).

Result 3 Both the sum of a priori belief over all node pairs and the sum of posterior belief (without exploiting similarity measures) overall all node pairs are equal to the number of edges:

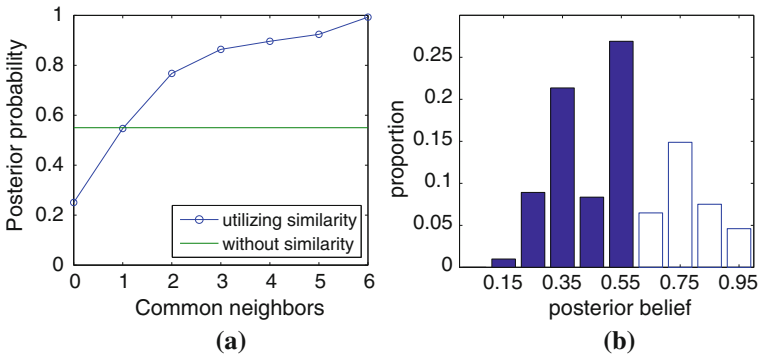

Fig. 3 Posterior belief for polbooks network. **a** Posterior belief versus common neighbors; **b** posterior belief distribution

$$\sum_{i<j} P(a_{ij} = 1) = \sum_{i<j} P(a_{ij} = 1|\tilde{a}_{ij}) = m.$$

The expectation of the sum of our enhanced posterior belief (with exploiting similarity measures) is equal to the number of edges:

$$\mathbf{E}\left[\sum_{i<j} P(a_{ij} = 1|\tilde{a}_{ij}, \tilde{m}_{ij})\right] = m.$$

Figure 3 shows the relationship between the two posterior beliefs and the common neighbors for the *polbooks* data. We set $k = 200$. We can observe that the posterior belief without exploiting the similarity measure, $P(a_{ij} = 1|\tilde{a}_{ij} = 1)$, is 0.55 for all observed links. However, our enhanced posterior belief $P(a_{ij} = 1|\tilde{a}_{ij} = 1, \tilde{m}_{ij})$ are greater than 0.55 for those links with more than 2 common neighbors as shown in Fig. 3a. Figure 3b shows the distribution of the calculated posterior belief values. We can observe that 33.5% of released links have their posterior beliefs enhanced with similarity measures.

3.3 Privacy protection measure

In the privacy preserving data mining, one natural question from data owner is how many perturbations we need such that we can guarantee the protection for all sensitive individual edges are above some tolerated threshold. When attackers utilize the similarity measure, the absolute measure of protection for an individual link (i, j) can be defined as

$$\tau_a(i, j) = 1 - \max_x \left\{ \max_{t=0,1} P(a_{ij} = 1|\tilde{a}_{ij} = t, \tilde{m}_{ij} = x) \right\} \tag{8}$$

where the second term denotes the maximal suspicion of existing $a_{ij} = 1$. Compared with the protection under the attack without exploiting similarity measures, we define the relative measure of protection as

$$\tau_r(i, j) = \frac{\tau_a(i, j)}{1 - \max_{t=0,1} P(a_{ij} = 1|\tilde{a}_{ij} = t)}.$$

The measures of protection (τ_a and τ_r) are defined in terms of one individual edge. In the privacy preserving data mining, one natural question is how many perturbations

we need such that we can guarantee the protection for all individual edges are above the threshold. Our next result shows the formula of the minimum number of perturbations to achieve the protection of all individual links. It is of great importance to evaluate the relationship between the required minimum number of perturbations and the utility loss of the perturbed graph. Due to space limitations, we leave this as our future work.

Result 4 In the original graph, let $S_x = \{(i, j) : m_{ij} = x\}$, $\rho_{\max} = \max_x \rho(S_x)$, and sparse ratio $r = m/\binom{n}{2}$. When the protection threshold $\epsilon < \frac{1-\rho_{\max}}{1-r}$, there exists the minimum k such that $\tau_r(i, j) \geq \epsilon$ stands for all the node pair (i, j). $k_{\min}$ **is given by**:

$$k_{\min} = \frac{[(1-r)\epsilon\rho_{\max} - r(1-\rho_{\max})]m}{\epsilon(\rho_{\max} - r)}. \tag{9}$$

4 Empirical evaluation

We used four network data sets (*polbooks, Enron, email, and polblogs*) in our evaluation. The *Enron* network was built from email corpus of a real organization over the course covering a 3-year period. We used a pre-processed version of the data set provided by Shetty and Adibi [21]. This data set contains 252,759 emails from 151 Enron employees, mainly senior managers. The *email* graph is the network of e-mail interchanges between members of the Univeristy Rovira i Virgili (Tarragona).[3] The *polblogs* compiles the data on the links among US political blogs, containing over 1,000 vertices and 15,000 edges, which is based on incoming and outgoing links and posts around the time of the 2004 presidential election.[4]

For each graph G, we randomly add k false edges and delete k true edges. We set $k = 0.3, 0.5, 0.7\,\mathrm{m}$ in this paper. We applied four similarity measures (Common neighbors, Katz, Adamic/Adar, Commute time) to predict top-t candidate links. The prediction performance was evaluated by the precision of the top-t predicted links. We varied t values from 0.1m to 0.5 m for all four data sets.

For each t, we calculated the precision of prediction links with different similarity measures. We also calculated the precision of prediction links using the posterior belief without exploiting the similarity measure. Figure 4 plots our results on four data sets. We can observe that for all four data sets, we can achieve very high accuracy (greater than 0.8) by using our enhanced posterior belief for a subset (top 0.1 m) of released links, which indicates severe privacy disclosures for those sensitive links. We can also see that our enhanced posterior belief achieves higher precisions than the previous posterior belief without exploiting similarity measures for most links (0.5 m) with high similarity measure values, indicating that the network topology does indeed contain latent information from which to infer interactions. From Fig. 4, we can also observe that we achieve different precisions using different similarity measures: one measure that achieves the highest precision for one data set is not necessarily the one for another data set. It is of great significance to explore what similarity measures can be exploited by attackers to achieve the highest privacy disclosure for a given social network. In addition to node similarity(dissimilarity) measures, the attackers may further exploit the graph topology to enhance their confidence on predicting sensitive links.

[3] http://deim.urv.cat/~aarenas/data/welcome.htm.

[4] http://www-personal.umich.edu/~mejn/netdata/.

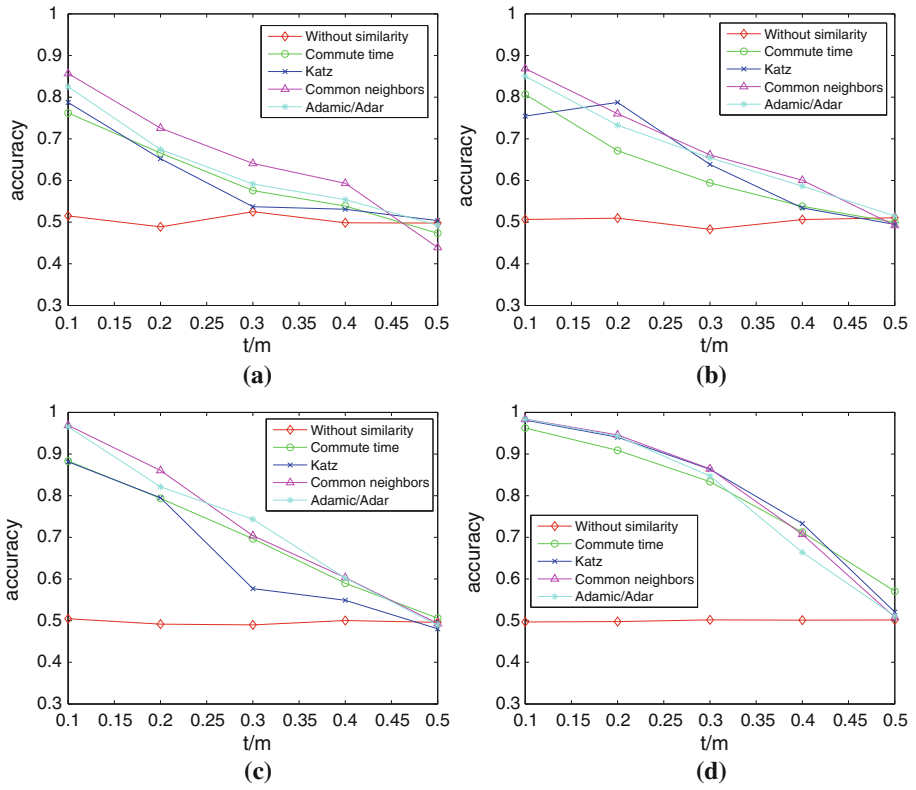

Fig. 4 Precision of top t predictions by the posterior belief w/o similarity measures for four data sets, $k = 0.5$ m. **a** Polbooks $G(105, 441)$; **b** enron $G(151, 1377)$; **c** email $G(1133, 5451)$; **d** polblogs $G(1222, 16714)$

For example, two node pairs (u, v) and (s, t) have the same number of common neighbors, but the common neighbors of (u, v) are strongly connected to each other while the common neighbors of (s, t) do not connect to each other. In this case, (u, v) are more likely to be connected than (s, t). Merely using the common neighbors measure in our current method cannot capture the difference. We will investigate the potential privacy disclosure risk when the attackers combine topology information with the node similarity/dissimilarity measures.

In the next experiment, we vary the noise magnitude k from 0.3 to 0.7 m. Table 1 shows the precisions of top t predictions using different similarity measures on four networks. We can see that for every noise magnitude, predictions that utilize similarity measures achieve a higher accuracy than those without exploiting similarity measures. We can also observe that, for any t, the precision decreases as noise magnitude k increases. This is intuitively reasonable, for large noises can greatly reduce the correlation between the similarity measures and existences of links, and thus decrease the prediction precision. We would point out that $k = 0.7$ m corresponds to a large randomization (i.e., 70% original links have been removed). The posterior belief without exploiting similarity measures $P(a_{ij} = 1|\tilde{a}_{ij} = 1)$ is only 0.3. However, the posterior belief with exploiting similarity measures is significantly improved. For example, the precision of top 0.1 m predictions using common neighbors is 0.87 for *polblogs* data.

Table 1 Precision of top t predictions by the posterior belief w/o similarity measures for four data sets, $k = 0.3, 0.5, 0.7$ m

	Polbooks			Enron			Email			Polblogs		
k:	0.3 m	0.5 m	0.7 m	0.3 m	0.5 m	0.7 m	0.3 m	0.5 m	0.7 m	0.3 m	0.5 m	0.7 m
(a) Without similarity measures												
t: 0.1 m	0.69	0.52	0.28	0.70	0.51	0.30	0.71	0.50	0.30	0.69	0.49	0.29
0.2 m	0.70	0.49	0.33	0.70	0.51	0.30	0.69	0.49	0.30	0.70	0.49	0.29
0.3 m	0.69	0.53	0.30	0.71	0.48	0.30	0.70	0.49	0.31	0.69	0.50	0.30
0.4 m	0.71	0.50	0.30	0.70	0.51	0.28	0.70	0.50	0.30	0.71	0.50	0.29
0.5 m	0.72	0.50	0.28	0.69	0.51	0.31	0.70	0.50	0.29	0.70	0.51	0.30
(b) Commute time												
t: 0.1 m	0.93	0.76	0.39	0.93	0.81	0.42	0.94	0.88	0.68	0.98	0.96	0.87
0.2 m	0.85	0.67	0.36	0.86	0.67	0.41	0.90	0.79	0.48	0.96	0.91	0.69
0.3 m	0.82	0.58	0.39	0.81	0.59	0.39	0.88	0.70	0.36	0.95	0.83	0.48
0.4 m	0.74	0.54	0.36	0.78	0.54	0.32	0.83	0.59	0.33	0.90	0.71	0.33
0.5 m	0.70	0.47	0.30	0.72	0.50	0.28	0.76	0.51	0.29	0.84	0.57	0.23
(c) Katz												
t: 0.1 m	0.94	0.79	0.59	0.95	0.75	0.39	0.97	0.88	0.69	1.00	0.98	0.90
0.2 m	0.81	0.65	0.42	0.91	0.79	0.36	0.98	0.79	0.53	0.98	0.94	0.73
0.3 m	0.75	0.54	0.30	0.87	0.64	0.32	0.94	0.58	0.40	0.97	0.86	0.49
0.4 m	0.76	0.53	0.23	0.80	0.53	0.32	0.88	0.55	0.30	0.94	0.73	0.32
0.5 m	0.70	0.50	0.27	0.75	0.49	0.30	0.79	0.48	0.24	0.88	0.52	0.20
(d) Common neighbors												
t: 0.1 m	0.97	0.85	0.45	0.97	0.86	0.41	0.99	0.96	0.70	0.99	0.98	0.87
0.2 m	0.94	0.72	0.35	0.96	0.76	0.34	0.98	0.86	0.49	0.98	0.94	0.58
0.3 m	0.90	0.64	0.33	0.93	0.66	0.32	0.96	0.70	0.44	0.97	0.86	0.39
0.4 m	0.84	0.59	0.26	0.89	0.60	0.31	0.91	0.60	0.34	0.95	0.70	0.26
0.5 m	0.82	0.43	0.28	0.83	0.49	0.28	0.82	0.49	0.27	0.90	0.50	0.22
(e) Adamic/Adar												
t: 0.1 m	0.98	0.83	0.43	0.98	0.85	0.42	1.00	0.97	0.67	1.00	0.98	0.86
0.2 m	0.94	0.67	0.37	0.96	0.73	0.36	0.99	0.82	0.54	0.99	0.94	0.57
0.3 m	0.90	0.59	0.33	0.93	0.65	0.31	0.95	0.74	0.45	0.97	0.85	0.41
0.4 m	0.83	0.55	0.34	0.89	0.59	0.29	0.90	0.60	0.34	0.94	0.66	0.27
0.5 m	0.81	0.49	0.29	0.84	0.51	0.28	0.84	0.49	0.28	0.91	0.51	0.23

5 Comparison with low-rank approximation-based prediction

The authors in Refs. [8,12,13] have investigated point-wise reconstruction methods (spectral filtering, PCA based, and SVD based), which may be exploited by attackers to breach individual privacy in the numerical data setting. Those methods work well because real-world data are usually highly correlated in a low-dimensional space while the additive noise is distributed (approximately) equally over all dimensions. Then, more accurate individual data can be reconstructed by projecting the randomized data into a proper low-dimensional space where the majority information of the original data is preserved. We implemented a similar

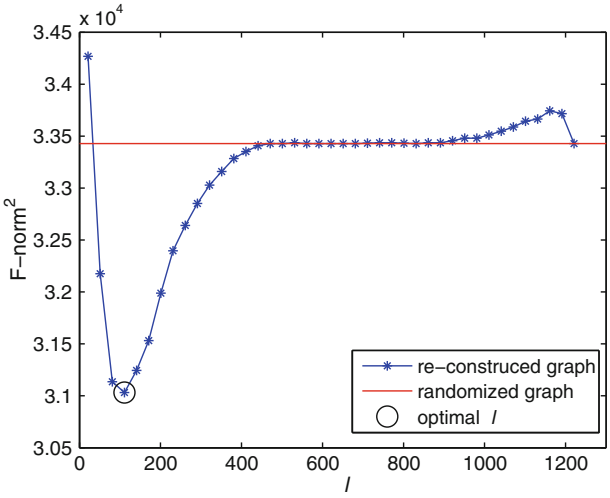

Fig. 5 $\|\hat{A} - A\|_F^2$ as l varies for *polblogs* network, $k = 0.5$ m. when l is chosen properly, the low-rank approximation-based prediction method can reduce the squared Frobenius norm

point-wise reconstruction method for graph data and reported our detailed findings in Wu et al. [23]. In this section, we conduct comparison between the low-rank approximation-based prediction and our similarity measures–based prediction.

The edge randomization process can be written in the matrix form $\tilde{A} = A + E$, where E is the perturbation matrix. We set $E(i, j) = E(j, i) = 1$ if edge (i, j) is added, $E(i, j) = E(j, i) = -1$ if edge (i, j) is deleted, and 0 otherwise. Let $\tilde{\lambda}_i$ be $\tilde{A}$'s ith largest eigenvalue in magnitude: $|\tilde{\lambda}_1| \geq |\tilde{\lambda}_2| \geq \cdots \geq |\tilde{\lambda}_n|$, and $\tilde{\boldsymbol{x}}_i$ denotes the eigenvector of $\tilde{\lambda}_i$. Then, the rank l approximation of $\tilde{A}$ are given by:

$$\tilde{A}_l = \sum_{i=1}^{l} \tilde{\lambda}_i \tilde{\boldsymbol{x}}_i \tilde{\boldsymbol{x}}_i^T.$$

By choosing a proper l, we expect that $\tilde{A}_l$ can still preserve the major information of the original graph and filter out noises added in the rest dimensions. In $\tilde{A}_l$, those entries whose values are close to 1 are more likely to have true edges while those entries whose values are close to 0 are less likely to have edges. We can simply get the reconstructed graph $\hat{A}$ by setting the $2m$ largest off-diagonal entries in $\tilde{A}_l$ as 1, and 0 otherwise.

Figure 5 plots $\|\hat{A} - A\|_F^2$ (denoted by "re-constructed graph") and $\|\tilde{A} - A\|_F^2$ (denoted by "randomized graph") as l varies for the *polblogs* network, setting $k = 0.5$ m. Notice that the squared Frobenius norm is exactly four times the number of different links between two graphs. We can observe that when l is too small (or too large), the reconstructed graph $\hat{A}$ is no better than the randomized graph $\tilde{A}$. This is because $\tilde{A}_l$ contains too little information of the original graph (or too much noise). However, when $l = 110$ (which is circled in Fig. 5), the accuracy of the low-rank approximation-based prediction reaches the maximum, which is significantly better than the prediction accuracy based on the randomized graph.

One problem here is how to determine the optimal l. In Refs. [8,13], the authors proposed strategies to determine the l by comparing $\tilde{\lambda}_l$ with the largest eigenvalue ε_1 of the noise matrix. Those strategies work well in the numerical data setting since the added noises have the independent identical distribution. However, the randomization mechanism in social networks (based on the positions of randomly chosen edges) is much different from the

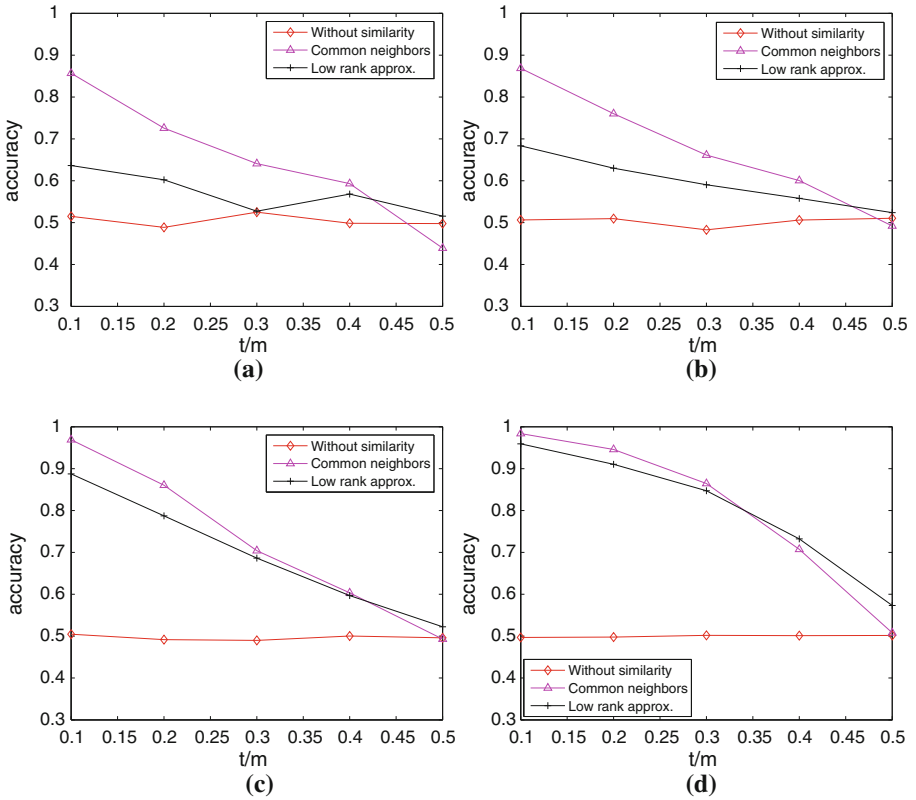

Fig. 6 Precision of top t predictions via common neighbors and the low-rank approximation, $k = 0.5\,\mathrm{m}$. **a** Polbooks $G(105, 441)$; **b** enron $G(151, 1377)$; **c** email $G(1133, 5451)$; **d** polblogs $G(1222, 16714)$

additive noise randomization (based on random values for all entries). When the magnitude of noise k is large, ε_1 can be even greater than $\tilde{\lambda}_1$ or λ_1. Hence, those strategies fail here. We would emphasize that it is a challenging problem to determine the optimal l in the graph randomization setting. In Ref. [23], the authors presented some heuristic method to determine the l by comparing some feature values of the reconstructed graphs with the original/estimated feature values of the original graph. In this paper, we simply assume that attackers know the optimal l. Please note that this is the worst case of link privacy disclosure.

Next, we compare our similarity measure–based prediction methods with the above low-rank approximation-based prediction method. Figure 6 plots the precisions of top t predicted links of the low-rank approximation prediction and our similarity measure–based prediction using common neighbors for the four networks ($k = 0.5\,\mathrm{m}$). We can observe that for *polbooks* and *Enron* networks, the common neighbors measure-based prediction achieves much higher precisions than the low-rank approximation-based prediction while there is no significant difference between these two methods for *email* and *polblogs* networks. Note that in our experiments, the low-rank approximation-based prediction method adopted the optimal l values in the reconstruction. In practice, it is difficult for attackers to derive the optimal l since they have no access to the original graph. Hence, we tend to conclude that the similarity measure–based prediction methods can incur larger link disclosures. We also conducted evaluations using other similarity measures with other k values on these four

networks. We skip those results in this paper due to space limitations. The skipped results have similar observations.

6 Conclusion and future work

In this paper, we have investigated how well the edge randomization approach via addition/deletion can protect privacy of sensitive links. We have conducted theoretical analysis and empirical evaluations to show that node proximity measures can be exploited by attackers to enhance the posterior belief and prediction accuracy of the existence of sensitive links among nodes with high similarity values. We have also compared our similarity measure–based prediction methods with the low-rank approximation-based prediction method.

There are some other aspects of this work that merit further research. Among them, we will continue the line of this research by investigating other edge randomization approaches (e.g., edge switches) and other proximity measures. Since how to preserve utility (in terms of various structural features) and privacy in the released graph is an important issue in privacy preserving social network analysis, we will study the tradeoff between privacy and utility for various randomization strategies. In this paper, we limit our scope as link disclosure. In our future work, we will investigate how well anonymization and randomization together can protect identity and link privacy when attackers exploit various complex background knowledge (e.g., attributes of vertices, vertex degrees, neighborhoods of some target individuals, embedded subgraphs) in their attacks. We will also study the scalability issue and conduct empirical evaluations on large social networks.

Acknowledgments This work was supported in part by U.S. National Science Foundation IIS-0546027 and CNS-0831204. All opinions, findings, conclusions, and recommendations in this paper are those of the authors and do not necessarily reflect the views of the funding agencies. Part of this work has been presented in PAKDD09 [27]. This extended version includes comparisons with the newly implemented low-rank approximation-based prediction method and complete empirical evaluations. We would like to thank Leting Wu for sharing us software codes of the low-rank approximation-based prediction method. We would also like to thank the anonymous reviewers for their insightful comments on our PAKDD09 submission and this journal submission, which help to improve the quality of the paper.

Appendix: Proofs

Proof of Property 1

It is easy to verify that when $1 - p_1 - p_2 \geq 0$, Inequality (5) stands if and only if

$$(1 - p_1 - p_2)[1 - \rho(\widetilde{S}_x)] \geq 0.$$

We need only guarantee $1 - p_1 - p_2 \geq 0$. Notice that $p_1 = \frac{k}{m}$, and $p_2 = \frac{k}{\binom{n}{2} - m}$, then we have

$$1 - p_1 - p_2 \geq 0 \Leftrightarrow 1 - \frac{k}{m} - \frac{k}{\binom{n}{2} - m} \geq 0 \Leftrightarrow \binom{n}{2} k \leq m \left[\binom{n}{2} - m \right]$$

$$\Leftrightarrow k \leq \left[1 - m / \binom{n}{2} \right] m = (1 - r)m.$$

□

Proof of Result 1 Let $N = |\widetilde{S}_x|$, $N_1 = |\widetilde{S}_x^1|$ and $\rho = \rho(\widetilde{S}_x)$. Then, for a randomly selected node pair (i, j), $\tilde{a}_{ij}$ is a Bernoulli random variable:

$$P(\tilde{a}_{ij} = 1|\tilde{m}_{ij} = x) = (1 - p_1)\rho + p_2(1 - \rho)$$
$$P(\tilde{a}_{ij} = 0|\tilde{m}_{ij} = x) = p_1\rho + (1 - p_2)(1 - \rho)$$

Then the likelihood function of $\widetilde{S}_x$ is

$$L = [(1 - p_1)\rho + p_2(1 - \rho)]^{N_1} [p_1\rho + (1 - p_2)(1 - \rho)]^{N-N_1}.$$

Take derivative to $\ln L$ with respect of ρ, we have

$$\frac{d \ln L}{d\rho} = \frac{N_1(1 - p_1 - p_2)}{(1 - p_1)\rho + p_2(1 - \rho)} - \frac{(N - N_1)(1 - p_1 - p_2)}{p_1\rho + (1 - p_2)(1 - \rho)}.$$

Set $\frac{d \ln L}{d\rho} = 0$, we have $\hat{\rho} = \frac{N_1/N - p_2}{1 - p_1 - p_2}$, and the unbiasedness is then obvious. □

Proof of Result 2 Notice that $P(a_{ij} = 1|\tilde{a}_{ij} = 1) = \frac{m-k}{m} = 1 - p_1$, and with Eq. (3), it is easy to verify this result. □

Proof of Result 3 $\sum_{i<j} P(a_{ij} = 1) = m$ is obvious. Notice that the number of edges does not change along the perturbation, then we have

$$\begin{aligned}\sum_{i<j} P(a_{ij} = 1|\tilde{a}_{ij}) &= \sum_{(i,j)\in\widetilde{E}} P(a_{ij} = 1|\tilde{a}_{ij} = 1) + \sum_{(i,j)\notin\widetilde{E}} P(a_{ij} = 1|\tilde{a}_{ij} = 0) \\ &= m \cdot \frac{m - k}{m} + \left[\binom{n}{2} - m\right] \cdot \frac{k}{\binom{n}{2} - m} = m. \end{aligned} \tag{10}$$

Given a randomized graph $\widetilde{G}$, $\tilde{a}_{ij}$ and $\tilde{m}_{ij}$ are fixed for any node pair (i, j). Let Φ denote the set of $\tilde{m}_{ij}$ values in $\widetilde{G}$: $x \in \Phi$ iff there is at least one node pair (i, j) in $\widetilde{G}$ such that $\tilde{m}_{ij} = x$. We have

$$\begin{aligned}\mathbf{E}\left[\sum_{i<j} P(a_{ij} = 1|\tilde{a}_{ij}, \tilde{m}_{ij})\right] = \sum_{x\in\Phi} \Bigg\{ &\sum_{(i,j)\in\widetilde{S}_x^1} \mathbf{E}[P(a_{ij} = 1|\tilde{a}_{ij} = 1, \tilde{m}_{ij} = x)] \\ &+ \sum_{(i,j)\in\widetilde{S}_x - \widetilde{S}_x^1} \mathbf{E}[P(a_{ij} = 1|\tilde{a}_{ij} = 0, \tilde{m}_{ij} = x)] \Bigg\} \end{aligned} \tag{11}$$

When attackers utilize the similarity measures with MLE, with the MLE in Eq. (6), we have

$$\begin{aligned}&\sum_{(i,j)\in\widetilde{S}_x^1} \mathbf{E}[P(a_{ij} = 1|\tilde{a}_{ij} = 1, \tilde{m}_{ij} = x)] = \frac{(1 - p_1)\hat{\rho}_1(\widetilde{S}_x)}{(1 - p_1)\hat{\rho}_1(\widetilde{S}_x) + p_2[1 - \hat{\rho}_1(\widetilde{S}_x)]} |\widetilde{S}_x^1| \\ &= (1 - p_1)|\widetilde{S}_x| \,\mathbf{E}[\hat{\rho}_1(\widetilde{S}_x)] \quad \text{(substitute Equation (6))} \\ &= (1 - p_1)|\widetilde{S}_x|\rho(\widetilde{S}_x) \\ &= (1 - p_1) \sum_{(i,j)\in\widetilde{S}_x} a_{ij} \quad \text{(by the definition of } \rho(\cdot)) \end{aligned} \tag{12}$$

Similarly, we have

$$\sum_{(i,j)\in\widetilde{S}_x-\widetilde{S}_x^1}\mathbf{E}[P(a_{ij}=1|\tilde{a}_{ij}=0,\tilde{m}_{ij}=x)]=p_1\sum_{(i,j)\in\widetilde{S}_x}a_{ij} \tag{13}$$

Combining Eqs. (11), (12) and (13) together, we have

$$\mathbf{E}\left[\sum_{i<j}P(a_{ij}=1|\tilde{a}_{ij},\tilde{m}_{ij})\right]=\sum_{x\in\Phi}\sum_{(i,j)\in\widetilde{S}_x}a_{ij}=\sum_{i,j}a_{ij}=m.$$

We prove the result. □

Proof of Result 4 When $k\le(1-r)m$, with Result 1 and 2, we have that

$$\max_x\left\{\max_{t=0,1}P(a_{ij}=1|\tilde{a}_{ij}=t,\tilde{m}_{ij}=x)\right\}=P(a_{ij}=1|\tilde{a}_{ij}=1,\tilde{m}_{ij}=x_0),$$

where x_0 is the value such that $\rho(\widetilde{S}_x)$ is maximized: $\rho(\widetilde{S}_{x_0})=\max_x\rho(\widetilde{S}_x)$. Let $\tilde{\rho}_{\max}=\rho(\widetilde{S}_{x_0})$. Meanwhile,we can also conclude

$$\max_{t=0,1}P(a_{ij}=1|\tilde{a}_{ij}=t)=P(a_{ij}=1|\tilde{a}_{ij}=1).$$

Then we have

$$\tau_r(i,j)=\frac{p_2[1-\tilde{\rho}_{\max}]}{p_1[(1-p_1)\tilde{\rho}_{\max}+p_2(1-\tilde{\rho}_{\max})]}. \tag{14}$$

Substitute

$$p_1=\frac{k}{m}=\frac{k}{rN}\quad\text{and}\quad p_2=\frac{k}{N-m}=\frac{k}{(1-r)N}$$

into Eq. (14), we can verify that $\tau_r(i,j)$ is an increasing function of k, and the maximum value is $\frac{1-\tilde{\rho}_{\max}}{1-r}$ when $k=(1-r)m$.

When $k\ge(1-r)m$, we similarly have the following:

$$\max_x\left\{\max_{t=0,1}P(a_{ij}=1|\tilde{a}_{ij}=t,\tilde{m}_{ij}=x)\right\}=P(a_{ij}=1|\tilde{a}_{ij}=0,\tilde{m}_{ij}=x_0),$$
$$\max_{t=0,1}P(a_{ij}=1|\tilde{a}_{ij}=t)=P(a_{ij}=1|\tilde{a}_{ij}=0).$$

In this case, $\tau_r(i,j)$ is a decreasing function of k, and the maximum is also $\frac{1-\tilde{\rho}_{\max}}{1-r}$ when $k=(1-r)m$.

Therefore, $k_{\min}$ exists if and only if $\epsilon\le\frac{1-\tilde{\rho}_{\max}}{1-r}$, and $k_{\min}<(1-r)m$. Then, $\tau_r(i,j)$ is given by Eq. (14). Solving the inequality $\tau_r(i,j)\ge\epsilon$, we have that

$$k\ge\frac{[(1-r)\epsilon\tilde{\rho}_{\max}-r(1-\tilde{\rho}_{\max})]m}{\epsilon(\tilde{\rho}_{\max}-r)}.$$

However, $\tilde{\rho}_{\max}=\max_x\rho(\widetilde{S}_x)$ varies from time to time due to the perturbation, and data owner can substitute it with the true maximum value $\rho_{\max}=\max_x\rho(S_x)$, then we get the result. □

References

1. Adamic LA, Adar E (2003) Friends and neighbors on the web. Soc Netw 25(3):211–230
2. Agrawal D, Agrawal C (2001) On the design and quantification of privacy preserving data mining algorithms. In: Proceedings of the 20th symposium on principles of database systems
3. Agrawal R, Srikant R (2000) Privacy preserving data mining. In: Proceedings of the ACM SIGMOD international conference on management of data. Texas, Dallas, pp 439–450
4. Backstrom L, Dwork C, Kleinberg J (2007) 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. ACM Press, New York, pp 181–190
5. Campan A, Truta TM (2008) A clustering approach for data and structural anonymity in social networks In: Proceedings of the 2nd ACM SIGKDD international workshop on privacy, security, and trust in KDD (PinKDD08)
6. Das S, Egecioglu Ömer, Abbadi AE (2009) Anonymizing edge-weighted social network graphs, Technical report, UCSB CS
7. Gkoulalas-Divanis A, Verykios VS (2009) Hiding sensitive knowledge without side effects. Knowl Inf Syst 20(3):263–299
8. Guo S, Wu X, Li Y (2008) Determining error bounds for spectral filtering based reconstruction methods in privacy preserving data mining. Knowl Inf Syst 17(2):217–240
9. Hanhijarvi S, Garriga GC, Puolamaki K (2009) Randomization techniques for graphs. In: Proceedings of the 9th SIAM conference on data mining
10. Hay M, Miklau G, Jensen D, Towsely D, Weis P (2008) Resisting structural re-identification in anonymized social networks. In: VLDB
11. Hay M, Miklau G, Jensen D, Weis P, Srivastava S (2007) Anonymizing social networks. University of Massachusetts Technical Report 07-19
12. Huang Z, Du W, Chen B (2005) Deriving private information from randomized data. In: Proceedings of the ACM SIGMOD conference on management of data. Baltimore, MA
13. Kargupta H, Datta S, Wang Q, Sivakumar K (2003) On the privacy preserving properties of random data perturbation techniques. In: Proceedings of the 3rd international conference on data mining. pp 99–106
14. Kargupta H, Datta S, Wang Q, Sivakumar K (2005) Random-data perturbation techniques and privacy-preserving data mining. Knowl Inf Syst 7(4):387–414
15. Katz L (1953) A new status index derived from sociometric analysis. Psychometrika 18(1):39–43
16. Liben-Nowell D, Kleinberg J (2003) The link prediction problem for social networks. In: 'CIKM '03: proceedings of the twelfth international conference on information and knowledge management. ACM, New York, pp 556–559
17. Liu K, Terzi E (2008) Towards identity anonymization on graphs. In: Proceedings of the ACM SIGMOD conference. ACM Press, Vancouver, Canada
18. Liu L, Wang J, Liu J, Zhang J (2009) Privacy preservation in social networks with sensitive edge weights. In: SDM. pp 954–965
19. Lovasz L (1993) Random walks on graphs. Combinatorics 2:1–46
20. Luo H, Fan J, Lin X, Zhou A, Bertino E (2009) A distributed approach to enabling privacy-preserving model-based classifier training. Knowl Inf Syst 20(2):157–185
21. Shetty J, Adibi J (2004) The Enron email dataset database schema and brief statistical report. Information sciences institute technical report, University of Southern California
22. Teng Z, Du W (2009) A hybrid multi-group approach for privacy-preserving data mining. Knowl Inf Syst 19(2):133–157
23. Wu L, Ying X, Wu X (2010) Reconstruction from randomized graph via low rank approximation. In: Proceedings of the 10th SIAM conference on data mining
24. Ying X, Pan K, Wu X, Guo L (2009) Comparisons of randomization and k-degree anonymization schemes for privacy preserving social network publishing. In: SNA-KDD '09: proceedings of the 3rd SIGKDD workshop on social network mining and analysis
25. Ying X, Wu X (2008) Randomizing social networks: a spectrum preserving approach. In: Proceedings of the 8th SIAM conference on data mining
26. Ying X, Wu X (2009a) Graph generation with prescribed feature constraints. In: Proceedings of the 9th SIAM conference on data mining
27. Ying X, Wu X (2009b) On link privacy in randomizing social networks. In: Proceedings of the 13th Pacific-Asia conference on knowledge discovery and data mining
28. Zheleva E, Getoor L (2007) Preserving the privacy of sensitive relationships in graph data. In: Proceedings of the 1st ACM SIGKDD international workshop on privacy, security, and trust in KDD (PinKDD07). pp 153–171

29. Zhou B, Pei J (2008) Preserving privacy in social networks against neighborhood attacks. IEEE 24th international conference on data engineering. pp 506–515
30. Zou L, Chen L, Özsu MT (2009) K-automorphism: a general framework for privacy preserving network publication. In: Proceedings of 35th international conference on very large data base

Author Biographies

Xiaowei Ying is a Ph.D. candidate in Information Technology at the University of North Carolina at Charlotte. He received his BA degree in Mathematics from Fudan University of China in 2006. His major research interests include privacy preserving data mining and social network analysis.

Xintao Wu is an Associate Professor in the Department of Software and Information Systems at the University of North Carolina at Charlotte, USA. He got his Ph.D. degree in Information Technology from George Mason University in 2001. He received his BS degree in Information Science from the University of Science and Technology of China in 1994, an ME degree in Computer Engineering from the Chinese Academy of Space Technology in 1997. His major research interests include data mining and knowledge discovery, data privacy and security.