Reconstruction from Randomized Graph via Low Rank Approximation

Leting Wu, Xiaowei Ying, Xintao Wu
Department of Software and Information Systems
Univ. of North Carolina at Charlotte
{lwu8,xying,xwu}@uncc.edu

Abstract

The privacy concerns associated with data analysis over social networks have spurred recent research on privacy-preserving social network analysis, particularly on privacy-preserving publishing of social network data. In this paper, we focus on whether we can reconstruct a graph from the edge randomized graph such that accurate feature values can be recovered. In particular, we present a low rank approximation based reconstruction algorithm. We exploit spectral properties of the graph data and show why noise could be separated from the perturbed graph using low rank approximation. We also show key differences from previous findings of point-wise reconstruction methods on numerical data through empirical evaluations and theoretical justifications.

1 Introduction

Social networks are of significant importance in various application domains such as marketing, psychology, epidemiology and homeland security. The privacy concerns associated with data analysis over social networks have spurred recent research on privacy-preserving social network analysis, particularly on privacy-preserving publishing of social network data.

To protect privacy, one common practice is to publish a naive node-anonymized version of the network, e.g., by replacing the identifying information of the nodes with random IDs. While the naive node-anonymized network still permits useful analysis, as first pointed out in [4,14], this simple technique does not guarantee privacy since adversaries may re-identify a target individual from the anonymized graph by exploiting some known structural information of his neighborhood.

The state-of-the-art anonymization methods on network data have three categories: K-anonymity privacy preservation via edge modification [17, 27, 28], edge randomization [14, 22–24], and clustering-based generalization [5, 7, 8, 13, 26]. These above anonymization approaches have been shown as a necessity in addition to naive anonymization to preserve privacy in publishing social network data.

In a social network, nodes usually correspond to individuals or other social entities, and an edge corresponds to the relationship between two entities. Each entity can have a number of attributes, such as age, gender, income, and a unique identifier. In this paper, we consider social networks in which node identities (and even entity attributes) are not confidential but sensitive links between individuals are confidential and should be protected. For example, in a transaction network, an edge denoting a financial transaction between two individuals is considered confidential while nodes corresponding to individual accounts is non-confidential.

$$\begin{array}{ccccc} G & \xrightarrow{\text{modification}} & \tilde{G} & \xrightarrow{\text{reconstruction}} & \hat{G} \\ \downarrow & & \downarrow & & \downarrow \\ f(G) & & f(\tilde{G}) & & f(\hat{G}) \end{array}$$

Figure 1: The process of graph modification and reconstruction

For a network $G(V, E)$ with a set of n nodes connected by a set of m links, where V denotes the set of nodes and $E \subseteq V \times V$ is the set of links, the link structure can be expressed as an the adjacency matrix $A = (a_{ij})_{n\times n}$: $a_{ij} = 1$ if node i and j are connected and $a_{ij} = 0$ otherwise [1].

We focus on one specific edge randomization strategy, *Rand Add/Del*, which randomly adds one edge followed by deleting another edge and repeats this process for k times. This strategy preserves the total number of edges in the original graph. Figure 1 shows the process of graph modification and reconstruction. The edge randomization process can be written in the matrix form $\tilde{A} = A + E$, where the perturbation matrix E is defined as $e_{ij} = e_{ji} = 1$ if edge (i, j) is added, $e_{ij} = e_{ji} = -1$ if edge (i, j) is deleted, and $e_{ij} = 0$ otherwise. The process of randomization and the randomization parameter k are assumed to be published along with the released graph $\tilde{G}$.

For randomization approach, there are two fundamentally conflicting requirements: privacy for the individual entry (a_{ij}) and utility of the perturbed data ($\tilde{A}$). It has been

[1] Note that, for ease of presentation, we use the following pairs of terms interchangeably: "graph" and "network", "node" and "vertex", "edge" and "link".

shown in [14, 22] that a medium or large perturbation is needed in order to protect the privacy of the individual entry under feature based attacks or structural attacks. However, as shown in our empirical evaluation, the utility of the released randomized graph (in terms of topological features) is significantly lost in the randomized graph when a medium or large perturbation is applied.

To preserve utility, several advanced randomization strategies have been investigated recently. In [22], Ying and Wu presented a randomization strategy that can preserve the spectral properties of the graph. They presented two spectrum preserving randomization methods, *Spctr Add/Del* and *Spctr Switch*, which keep graph spectral characteristics (i.e., the largest eigenvalue of the adjacency matrix and the second smallest eigenvalue of the Laplacian matrix) not much changed during randomization by examining eigenvector values of nodes to choose where edges are added/deleted or switched. In [12,23], the authors studied the problem of how to generate a synthetic graph matching given features of a real social network in addition to a given degree sequence. They proposed a Markov Chain based feature preserving randomization. Although the proposed advanced randomization strategies generally can preserve more structural properties, it is very challenging to quantify disclosure risks since the process of feature preserving strategies are complicated.

In this paper, we adopt a different approach. We focus on whether we can reconstruct a graph $\hat{G}$ from the randomized one $\tilde{G}$ such that $\hat{G}$ is closer to the original graph G than $\tilde{G}$ in terms of some feature f, i.e., $|f(\hat{G}) - f(G)| \leq |f(\tilde{G}) - f(G)|$. In particular, we study the use of low rank approximation approach to reconstruct structural features from the randomized graph. We exploit spectral properties of the graph data and show that the noise could be separated from the perturbed graph.

1.1 Contribution

Our contributions are as follows.

- We first propose the use of low rank approximation to reconstruct the graph topology from the randomized network. While edge randomization can naturally be considered as an additive-noise perturbation, we shall show those point-wise reconstruction methods [11, 15, 16] developed in the numerical data setting are not applicable in this network data setting. We also present a novel solution to determine the (approximate) optimal rank, a key parameter in our reconstruction algorithm.
- To derive the low rank approximation based reconstruction method for network data, we examine the relationship between graph topological structure and spectral spaces determined by eigen-pairs of the adjacency matrix. In particular, we discover that eigen-pairs of leading positive eigenvalues capture the inner-community connection structure while eigen-pairs of leading negative eigenvalues capture the inter-community connections.
- We explicitly assess effects of perturbation on the accuracy of the reconstructed feature values. Our empirical evaluation results show that accurate feature values can still be recovered from the randomized graphs even with the large magnitude of noise (e.g., $k = 0.8m$).
- One surprising finding is that, for most social networks, the reconstructed networks do not incur further disclosure risks of individual privacy than the released randomized graphs. This is very different from the numerical data setting. Our further investigation shows that only networks with low ranks or a small number of dominant eigenvalues may incur further privacy disclosure due to reconstruction.

1.2 Paper Organization

The rest of this paper is organized as follows. In Section 2, we first discuss topological features used in this paper and revisit those low rank approximation based reconstruction methods on numerical data. In Section 3, we examine the spectra of network data and show the relationship between the positive (negative) eigenvalues and the reconstructed graph structure via low rank approximation. In Section 4, we present our low rank approximation based reconstruction algorithm. We also show our novel method to determine the optimal rank for low rank approximation. We conduct empirical evaluations on three real social networks in terms of both privacy and utility in Section 5. In Section 6, we further examine what type of graphs are sensitive to low rank approximation based reconstruction in terms of privacy protection. Finally we offer our concluding remarks and point out future directions in Section 7.

2 Preliminaries

Table 1: Notations

n, m	number of nodes and edges
k	number of edges added and deleted
r	number of eigen-pairs in low rank approximation
$A(\tilde{A})$	adjacency matrix of graph G ($\tilde{G}$)
A_r ($\tilde{A}_r$)	rank r approximation of A ($\tilde{A}$)
$\hat{A}$	adjacency matrix of the reconstructed graph
$\lambda_i, \boldsymbol{x}_i$	the ith largest eigenvalue in magnitude of A and the corresponding eigenvector
E	difference matrix, $E = \tilde{A} - A$
ε_1	the largest eigenvalue of E in magnitude

2.1 Notation and Features

We use the tilde conventions to denote perturbations and use the hat conventions to denote estimations. The original quantity is denoted by the same symbol without a tilde or hat. Table 1 summarizes our notations used in this paper.

To understand and utilize the information in a network, researches have developed various measures to indicate the structure and characteristics of the network from different perspectives [10]. In this paper, we consider the following topological features of the graph:

- λ_1, the largest eigenvalue of the adjacency matrix A. The eigenvalues of A encode information about the cycles of a network as well as its diameter. The maximum degree, chromatic number, clique number, and extend of branching in a connected graph are all related to λ_1. In [21], the authors studied how a virus propagates in a real work and proved that the epidemic threshold for a network is closely related to λ_1.
- ν_2, the second largest eigenvalue of the normal matrix $N = D^{-1}A$. Let $\nu_1 \geq \nu_2 \geq \cdots \geq \nu_n$ denote the eigenvalues of N, $\nu_1 \equiv 1$. $1 - \nu_2$ is the lower bound of the normal cut of the graph [19]. Therefore, ν_2 is close to 1 if the graph has a clear community structure, and the eigenvectors of ν_2 is a good indicator of the community partition.
- Q, modularity indicates the goodness of the community structure [10]. It is defined as the fraction of all edges that lie within communities minus the expected value of the same quantity in a graph generated from a random model which keeps the expected number of degree for each node. A value $Q = 0$ indicates that the community structure is no stronger than would be expected by random chance and high value other than zero represents large deviations from randomness.
- C, transitivity measure is one type of clustering coefficient measure and characterizes the presence of local loops near a vertex. It is formally defined as $C = 3N_\Delta/N_3$, where N_Δ is the number of triangles and N_3 is the number of connected triples.

Throughout this paper, we use the *polblogs* as an example. The *polblogs* network compiles the 16714 links among 1222 US political blogs, based on incoming and outgoing links and posts during the time of the 2004 presidential election [1].

2.2 Low Rank Approximation based Reconstruction Methods on Numerical Data Revisited The low rank approximation has been well investigated as a point-wise reconstruction method in the numerical setting. In the setting of randomizing numerical data, a data set U with m records of n attributes is perturbed to $\tilde{U}$ by an additive noise data set V with same dimensions as U, i.e., $\tilde{U} = U + V$. A spectral filtering based reconstruction method was first proposed in [16] to reconstruct original data values from the perturbed data. Similar methods (e.g., PCA based reconstruction method [15], SVD based reconstruction method [11]) have also been investigated. All methods exploited spectral properties of the correlated data to remove the noise from the perturbed data set. This is because real-world numerical data is usually highly correlated in a low dimensional space while the randomly added noise is distributed (approximately) equally over all dimensions. Then, more accurate aggregate features can be reconstructed by projecting the randomized data into a proper low dimensional space where the majority information of the original data is preserved.

Spectral Filtering. The objective of the spectral filtering based approach is to derive the estimation $\hat{U}$ of U from the perturbed data $\tilde{U}$ based on random matrix theory. An explicit filtering procedure is shown below.

1. Calculate the covariance matrix of $\tilde{U}$ by $\tilde{\Sigma} = \tilde{U}^T\tilde{U}$ (assume U has mean equal to 0).
2. The covariance matrix $\tilde{\Sigma}$ is symmetric and positive semi-definite, we apply spectral decomposition on $\tilde{\Sigma}$ to get its i-th largest eigenvalue $\tilde{\lambda}_i$ and the corresponding eigenvector $\tilde{\boldsymbol{x}}_i$.
3. Derive the eigenvalues information from the covariance matrix of the noise V and choose a proper number of dimensions, r.
4. Let $\tilde{X}_r = [\tilde{\boldsymbol{x}}_1\, \tilde{\boldsymbol{x}}_2 \cdots \tilde{\boldsymbol{x}}_r]$, and the orthogonal projection on to the subspace spanned by $\tilde{\boldsymbol{x}}_1, \ldots, \tilde{\boldsymbol{x}}_r$ is $P_r = \tilde{X}_r\tilde{X}_r^T$. Obtain the estimated data set using $\hat{U} = \tilde{U}P_r$.

SVD. Singular value decomposition decomposes a matrix $U \in \mathbb{R}^{m\times n}$ (say $m \geq n$) as $U = \sum_{i=1}^{n} \sigma_i \boldsymbol{p}_i \boldsymbol{q}_i^T$, where $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n$ are the *singular values* and $\boldsymbol{p}_i \in \mathbb{R}^m$ and $\boldsymbol{q}_i \in \mathbb{R}^n$ are the left and right *singular vector* of σ_i respectively. Similarly, after perturbation $\tilde{U} = U + V$, we have the SVD of $\tilde{U}$ as $\tilde{U} = \sum_{i=1}^{n} \tilde{\sigma}_i \tilde{\boldsymbol{p}}_i \tilde{\boldsymbol{q}}_i$. The SVD reconstruction method simply reconstructs U approximately as $\hat{U} = \tilde{U}_r = \sum_{i=1}^{r} \tilde{\sigma}_i \tilde{\boldsymbol{p}}_i \tilde{\boldsymbol{q}}_i^T$.

It has been shown that the spectral filtering method is equivalent to the SVD reconstruction method [11]. We can observe that all spectral based methods reconstruct the original data by projecting the perturbed data onto the projection subspaces that are determined by the first r eigenvectors for the spectral filtering method or by the first r singular vectors for the SVD method. The original spectral filtering algorithm [16] suggested using $r = \max\{i|\tilde{\lambda}_i \geq \varepsilon_1\}$ to determine the first r eigen components, where ε_1 is the largest eigenvalue of the noise covariance matrix $\mathrm{Cov}(V)$. The authors of [11] further proved that using $r = \max\{i|\tilde{\lambda}_i \geq 2\varepsilon_1\}$ can achieve approximately optimal reconstruction for i.i.d. noise. This is because that it only includes the i-th eigen component when the benefit due to inclusion of the i-th component is greater than the loss due to the noise projected on the i-th component, i.e., $\tilde{\lambda}_i \geq 2\varepsilon_1$.

3 Low Rank Approximation on Graph Data

The adjacency matrix A discussed here is different from the numerical data set U and the covariance matrix Σ in the following perspectives. First, A is a symmetric 0-1 matrix whereas U is a numerical matrix and the covariance matrix Σ is a semi-definite one. Second, for numerical data, all the eigenvalues of Σ are real and non-negative. For graph data A, the covariance matrix is not properly defined. We can see that in AA^T, the non-zero entry at row i column j means j is 2 steps away from i. When we directly apply eigen-decomposition on the adjacency matrix A, the eigen-decomposition of A contains negative eigenvalues.

In Section 3.1, we study the low rank approximation on graph data. In Section 3.2, we examine the spectra of graph data and show the relationship between the topological graph structure and the significant eigen-pairs that may involve both positive and negative eigenvalues.

3.1 Low Rank Approximation Let λ_i be A's i-th largest eigenvalue in magnitude: $|\lambda_1| \geq |\lambda_2| \geq \cdots \geq |\lambda_n|$, and $\boldsymbol{x}_i$ denotes the eigenvector of λ_i. The rank r approximations of A via the eigen-decomposition are given by:

$$A_r = \sum_{i=1}^{r} \lambda_i \boldsymbol{x}_i \boldsymbol{x}_i^T. \tag{3.1}$$

Among all the matrix with rank no larger than r, the low rank approximation A_r shown in (3.1) is the matrix closest to A in term of the Frobenius norm [20]:

$$\|A_r - A\|_F^2 = \min_{\mathrm{rank}(B)\leq r} \|B - A\|_F^2.$$

The key difference between our low rank approximation on graph data and those low rank approximation methods on numerical data is that we rank eigenvalues based on their absolute values and also include those significant negative eigenvalues in the low rank approximation. In Section 3.2, we will illustrate the relationship between the graph topology and significant positive and negative eigenvalues.

Because A_r is a real matrix, we need to derive a symmetric 0-1 matrix $\hat{A}$ that is close to A_r. Our strategy is to find the $2m$ largest off-diagonal entries in A_r (note that A and $\hat{A}$ are symmetric) and set the corresponding entries in $\hat{A}$ as 1 and others as 0, i.e.,

$$\hat{A}(i,j) = \begin{cases} 1, & \text{if } A_r(i,j) \text{ is one of the } 2m \\ & \text{largest off-diagonal entries,} \\ 0, & \text{otherwise.} \end{cases} \tag{3.2}$$

By using (3.2), we have the following property.

PROPERTY 1. *If $\hat{A}$ is obtained by* (3.2)*, $\hat{A}$ is the closest adjacency matrix to A_r in term of the Frobenius norm, i.e.,*

$$\|\hat{A} - A_r\|_F^2 = \min_{B\in\mathcal{A}_n^m} \|B - A_r\|_F^2,$$

where $\mathcal{A}_n^m$ denotes the set of all symmetric $n\times n$ 0-1 matrices with $2m$ off-diagonal 1's and 0 else where.

The following theory states that the difference between the spectrum of $\hat{A}$ and that of A_r is upper bounded by $\|\hat{A} - A_r\|_F^2$.

THEOREM 1. *[20] Given two $n \times n$ symmetric matrices A and E with eigenvalues $\lambda_1 \geq \cdots \geq \lambda_n$ and $\varepsilon_1 \geq \cdots \geq \varepsilon_n$ respectively. Let $\tilde{\lambda}_1 \geq \cdots \geq \tilde{\lambda}_n$ be the eigenvalues of $\tilde{A} = A + E$. Then we have*

$$\lambda_i + \varepsilon_n \leq \tilde{\lambda}_i \leq \lambda_i + \varepsilon_1, \tag{3.3}$$

$$\textstyle\sum_i (\tilde{\lambda}_i - \lambda_i)^2 \leq \|E\|_F^2. \tag{3.4}$$

By minimizing this upper bound, we expect the eigenvalues and eigenvectors of $\hat{A}$ is close to those of A. In fact, many spectral properties, such as eigenvectors, the sum of several eigenvalues, and spectral subspace, are stable when the magnitude of the difference matrix is moderate. For varies spectrum bounds and more details, please refer to [20]. Since the graph topology is closely related with eigenvalues and eigenvectors of the graph, we expect that $\hat{A}$ can preserve the major topological information of the original graph.

3.2 Leading Eigen-pairs vs. Graph Topology In this section, we study the relationship between eigen-pairs and graph topology. In particular, we examine the role of positive and negative eigenvalues in graph topology.

Without loss of generality, we partition the node set V into two groups $V_1 = \{1, \ldots, n_1\}$ and $V_2 = \{n_1+1, \ldots, n\}$. Then the adjacency matrix can be partitioned as

$$A = A_{\text{inner}} + A_{\text{inter}} = \begin{pmatrix} A_{11} & \mathbf{0} \\ \mathbf{0} & A_{22} \end{pmatrix} + \begin{pmatrix} \mathbf{0} & A_{12} \\ A_{12}^T & \mathbf{0} \end{pmatrix}, \tag{3.5}$$

where A_{11} and A_{22} represent the edges within V_1 and V_2 respectively, and A_{12} represents the edges between V_1 and V_2.

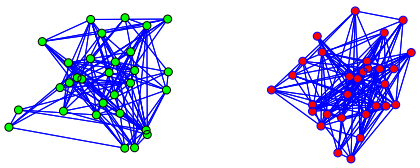
Disconnected communities In an ideal graph with two disconnected communities, A_{11} and A_{22} are dense matrices of comparable size, and $A_{12} = \mathbf{0}$. Then, all the eigenvalues of A_{11} and A_{22} are eigenvalues of A. Let μ_1 and η_1 be the largest eigenvalue in magnitude of A_{11} and A_{22} with eigenvector $\boldsymbol{y}_1$ and $\boldsymbol{z}_1$ respectively. μ_1 and η_1 are two eigenvalues of A with eigenvectors $\binom{\boldsymbol{y}_1}{\mathbf{0}}$ and $\binom{\mathbf{0}}{\boldsymbol{z}_1}$. Note that, by the Perron-Frobenius theorem [9], μ_1 and η_1 must be positive and all entries in $\boldsymbol{y}_1$ and $\boldsymbol{z}_1$ must be positive. Assume $\mu_1 \geq \eta_1$, then

$$A_1 = \mu_1 \begin{pmatrix} \boldsymbol{y}_1 \\ \mathbf{0} \end{pmatrix} (\boldsymbol{y}_1^T, \mathbf{0}) = \begin{pmatrix} \mu_1 \boldsymbol{y}_1 \boldsymbol{y}_1^T & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix}. \tag{3.6}$$

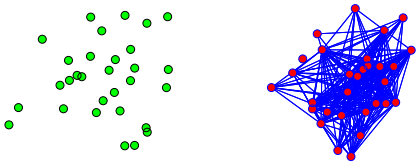
We can see all large entries only appear among the nodes in V_1. Similarly, the rank 2 approximation of A is given by

$$A_2 = \begin{pmatrix} \mu_1 \boldsymbol{y}_1 \boldsymbol{y}_1^T & \mathbf{0} \\ \mathbf{0} & \eta_1 \boldsymbol{z}_1 \boldsymbol{z}_1^T \end{pmatrix}, \tag{3.7}$$

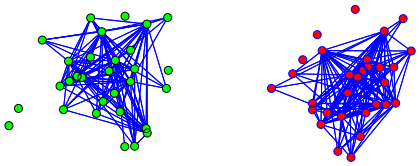
and large entries appear both within V_1 and V_2. Figure 2 shows a synthetic network with 60 nodes and 280 edges. This network contains two disconnected 30-node communities generated via ER model with inner-community probability 0.5. The derived graphs $\hat{A}$ by discretizing A_1 and A_2 via (3.2) are shown in Figure 2(b) and 2(c). For the graph derived from A_1, all the edges appear in only one of the communities. After adding one more eigen-pair in the low rank approximation, the derived graph shown in Figure 2(c) reveals two very clear communities.



(a) Original graph



(b) $r = 1$



(c) $r = 2$

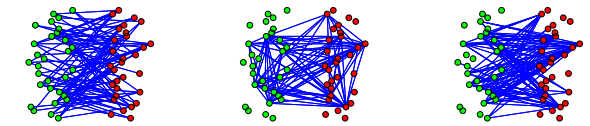
Figure 2: Synthetic random graph with two disconnected communities

Bipartite graph The negative eigenvalues are closely related to the bipartite structure of the graph. A bipartite graph is a graph containing two types of nodes, and edges only exist between two nodes of different types. For a bipartite graph, A_{11} and A_{22} in (3.5) are both zero matrix. The spectrum of A is then fully determined by A_{12}. Let $\sigma \geq 0$ be the largest singular value of A_{12} (note A_{12} is generally a non-square matrix) with right-singular value $\boldsymbol{u}$ and left-singular value $\boldsymbol{v}$. If G is a connected graph, all the entries of $\boldsymbol{u}$ and $\boldsymbol{v}$ are positive. It is easy to verify that σ and $-\sigma$ are both the eigenvalues of A with eigenvalue $\begin{pmatrix} \boldsymbol{u} \\ \boldsymbol{v} \end{pmatrix}$ and $\begin{pmatrix} -\boldsymbol{u} \\ \boldsymbol{v} \end{pmatrix}$ respectively. Similar as (3.6) and (3.7), we can have

$$A_1 = \begin{pmatrix} \sigma \boldsymbol{u}\boldsymbol{u}^T & \sigma \boldsymbol{u}\boldsymbol{v}^T \\ \sigma \boldsymbol{v}\boldsymbol{u}^T & \sigma \boldsymbol{v}\boldsymbol{v}^T \end{pmatrix}, \; A_2 = \begin{pmatrix} \mathbf{0} & 2\sigma \boldsymbol{u}\boldsymbol{v}^T \\ 2\sigma \boldsymbol{v}\boldsymbol{u}^T & \mathbf{0} \end{pmatrix}.$$

We can see that entries within V_1 and V_2 in A_1 are non-zero, which is significantly different from A. However, as we introduce the leading negative eigenvalue, non-zero entries in A_2 only appear in those entries across two type of nodes.

Figure 3(a) shows a synthetic bipartite graph with 60 nodes and 94 edges. Any two nodes of different colors have probability 0.1 to be connected, and nodes of the same color do not connect to each other. The first two eigenvalues are 4.27 and -4.27 respectively. The derived graphs ($\hat{A}$) from A_1 and A_2 are shown in Figure 3(b) and 3(c) respectively. We can see that, when only the positive eigenvalue and its eigenvector are involved, many edges connecting two nodes of the same type are falsely introduced in $\hat{A}$; and as the negative eigenvalue and its eigenvectors are included, $\hat{A}$ derived from A_2 shown in Figure 3(c) correctly reveals the bipartite structure.



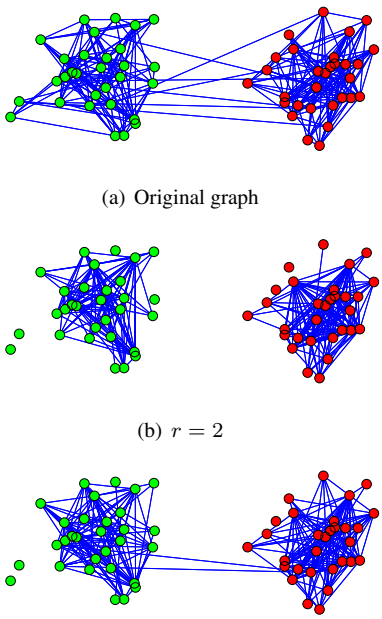
(a) Original graph (b) $r = 1$ (c) $r = 2$

Figure 3: Synthetic random quasi-bipartite graph

Social Networks Social networks usually have clear connected community structures. In other words, there are few non-zero entries in A_{inter}, i.e., $\|A_{\text{inter}}\|_F^2$ is small. By Theory 1, the eigenvalues and eigenvectors of A are close to A_{inner}, and similar to (3.6) and (3.7), the upper right and lower left parts of A_1 and A_2 are close to $\mathbf{0}$.

Figure 4(a) shows a synthetic network with 2 clear but connected communities. It is generated by adding inter-community edges with probability 0.05 to the synthetic graph in Figure 2(a). The first four eigenvalues are $\lambda_1 = 10.30, \lambda_2 = 9.05, \lambda_3 = -4.82$, and $\lambda_4 = -4.79$. The $2m$ largest entries in A_2 and A_4 are shown in Figure 4(b) and 4(c) respectively. Similar as Figure 2(c), large entries of A_2 appear in both of the two communities, and no inter-community entries have large values. As two negative eigenvalues λ_3 and λ_4 are included in A_4, inter-community edges emerge. $\hat{A}$ is closer to the original graph A.

For graphs containing c large communities, the c largest positive eigenvalues corresponds to the communities. If node j and node k belong to the i-th community C_i, the j-th and k-th entry of $\boldsymbol{x}_i$ (x_{ji} and x_{ki}) tend to be large, which



(a) Original graph

(b) $r = 2$

(c) $r = 4$

Figure 4: Synthetic random graph with two clear but connected communities

matches the finding by Ying and Wu [25] that eigenvectors corresponding to the large positive eigenvalues of A are good indices of the community partition. Then, the (j, k) entry of matrix $(\lambda_i \boldsymbol{x}_i \boldsymbol{x}_i^T)$, which equals to $\lambda_i x_{ji} x_{ki}$, tends to be large. Therefore, large entries in the low rank approximation matrix $A_c = \sum_{i=1}^{c} \lambda_i \boldsymbol{x}_i \boldsymbol{x}_i^T$ would reflect edges within these communities.

Similarly, for a quasi-bipartite graph, A_{inner} has few non-zero entries ($\|A_{\text{inner}}\|_F^2$ is small). Hence, the spectrum of A would have the similar pattern of A_{inter} and some of the leading eigenvalues of A are negative. Besides quasi-bipartite graphs, bowtie graphs [6] or graphs with very skewed degree distribution also have their adjacency matrices close to bipartite graphs (in term of the Frobenius norm). In such graphs, a large number of nodes do not connect to each other directly but through a small number of core nodes and core nodes are well connected to each other. Suppose node set V_1 represents the core nodes, then A_{11} represents the edges among core nodes, and A_{22} represents the edges among non-core nodes. $\|A_{\text{inner}}\|_F^2$ is small because there are few edges in A_{22} and the size of A_{11} is small. By the perturbation theory, the spectrum of a bowtie graph is similar to that of a bipartite graph and has significant negative eigenvalues.

4 Reconstruction from Randomized Graph via Low Rank Approximation

Recall that in the edge randomization process, we randomly add k false edges followed by deleting k true edges. The perturbation can be expressed as a perturbation matrix E where $e_{ij} = e_{ji} = 1$ if edge (i, j) is added, $e_{ij} = e_{ji} = -1$ if edge (i, j) is deleted, and $e_{ij} = 0$ otherwise. The process of randomization and the randomization parameter k are assumed to be published along with the released graph.

In Section 4.1, we present our low rank approximation based reconstruction algorithm and show why the algorithm (given an optimal rank r) can reconstruct topological features accurately. In Section 4.2, we conduct theoretical analysis and give our procedure to determine the optimal r.

4.1 Algorithm Let $\tilde{\lambda}_i$ be $\tilde{A}$'s i-th 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 $\tilde{\lambda}_i$. The rank r approximation of $\tilde{A}$ is $\tilde{A}_r = \sum_{i=1}^{r} \tilde{\lambda}_i \tilde{\boldsymbol{x}}_i \tilde{\boldsymbol{x}}_i^T$.

The topology of the randomized graph $\tilde{A}$ may be significantly different from that of the original graph A when the magnitude of perturbation is medium or large. However, by choosing an appropriate r, $\tilde{A}_r$ can preserve major topological structures. This is because that $\tilde{A}_r$ only includes those significant eigen-pairs and filters out all noises added in the rest dimensions. Recall that the leading eigen-pairs reflect the dominant structure of the graph, e.g., those eigen-pairs with large positive eigenvalues capture the inner structure of those significant communities and those eigen-pairs with negative eigenvalues capture the inter-community connections. Since $\tilde{A}$ is obtained by randomly adding and deleting edges on A, both strong inner- and inter-community connections are less affected by the randomization. Therefore, $\tilde{A}_r$ consisting of the leading eigen-pairs can still capture the major topological structures of the original graph.

After low rank approximation, $\tilde{A}_r$ is a real matrix. Similarly we adopt the following strategy to obtain a 0-1 matrix $\hat{A}$ as the reconstructed graph.

$$\hat{A}(i,j) = \begin{cases} 1, & \text{if } \tilde{A}_r(i,j) \text{ is one of the } 2m \\ & \text{largest off-diagonal entries,} \\ 0, & \text{otherwise.} \end{cases} \tag{4.8}$$

We show our graph reconstruction algorithm in Algorithm 1.

4.2 Determine r in Low Rank Approximation based Graph Reconstruction In the low rank approximation, the different choices of r can significantly affect the accuracy of reconstruction. When r is very small, the topological structure of the reconstructed $\hat{A}$ may be significantly different from that of the original graph A. This is because too few eigen-pairs are included in reconstruction and not all major structures are captured during the reconstruction. On the

Algorithm 1 Graph Reconstruction Algorithm

Input: randomized graph $\tilde{A}$, randomization parameter k
Output: reconstructed graph $\hat{A}$

1: Calculate $\tilde{\lambda}_i$ and $\tilde{\boldsymbol{x}}_i$, $|\tilde{\lambda}_1| \geq \cdots \geq |\tilde{\lambda}_n|$.
2: Calculate λ_1^* using (4.9);
3: $r = 1$;
4: **repeat**
5: Construct $\hat{A}$ from $\tilde{A}_r = \sum_{i=1}^{r} \tilde{\lambda}_i \tilde{\boldsymbol{x}}_i \tilde{\boldsymbol{x}}_i^T$ by (4.8).
6: $\hat{\lambda}_1 =$ the largest eigenvalue of $\hat{A}$ in magnitude;
7: $r = r + 1$;
8: **until** $|\hat{\lambda}_1 - \lambda_1^*|$ increases

other hand, the reconstruction with a large r may introduce too much noise. As a result, the benefit due to the inclusion of major structures is decreased by the loss due to the added noise. Figure 5 shows the reconstructed feature values, along with the original and randomized values, for *polblogs* network as the choice of r varies ($k = 0.4m$). When r is very small, the reconstructed feature values are significantly different from the original value, indicating that the topology of $\hat{A}$ is very different from the original graph. As r increases, $f(\hat{A})$ approaches the original value, and for some r, the reconstructed value approximately equals to original value. Further increasing r makes the reconstructed feature values approach to the randomized one, indicating that too much noise is included in $\hat{A}$. We can see that choosing a proper r is critical in reconstructing graphs.

We would emphasize again that the strategies of determining r in reconstructing numerical data (via comparing $\tilde{\lambda}_i$ with ε_1) is not applicable here. This is because the entries of E can only be 0, 1 and -1, and the magnitude of E can be very large while k is actually moderate. For example, when we randomly add and delete $k = 0.4m$ edges on *polblogs* network, we can get $\varepsilon_1 = 28.6$, which is greater than almost all $\tilde{\lambda}_i$ except $\tilde{\lambda}_1$ and $\tilde{\lambda}_2$. The strategies of determining r by $r = \max\{i|\tilde{\lambda}_i \geq \varepsilon_1\}$ [15] would choose $r = 2$. However, as shown in Figure 5, when $r = 2$, the feature values of the reconstructed graph are significantly different from the original value.

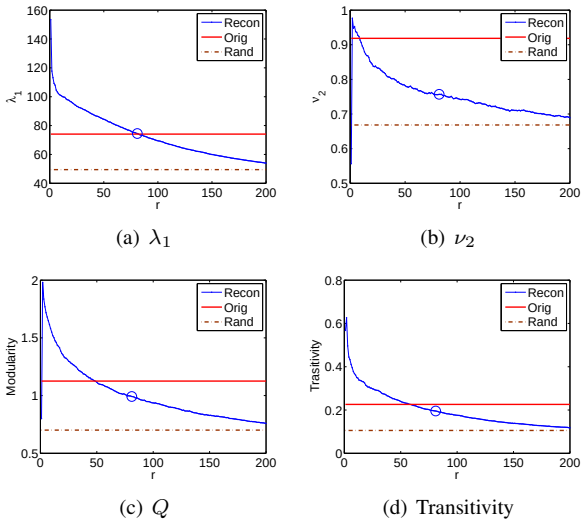
One natural idea is to determine r such that $f(\hat{A})$ is approximately equal to $f(A)$ for some feature f. One problem is that feature values of the original graph may not be available to data miners. In general, it is difficult, if not impossible, to derive the accurate estimates of real space feature values (e.g., cluster coefficient, transitivity) from the randomized graph using the statistics of randomization. However, for the spectral feature λ_1, we can derive the moment estimate of the original values, as shown in our next result.

RESULT 1. *Let $N = \binom{n}{2} - m$, and $\tilde{\lambda}_0 = \tilde{\boldsymbol{x}}_1^T(\mathbf{1} - I - \tilde{A})\tilde{\boldsymbol{x}}_1$, where $\mathbf{1}$ is a $n \times n$ all 1 matrix and I is the identity matrix. Let λ_1^* denote the moment estimator of λ_1. If $\tilde{A}$ is obtained by adding k false edges and deleting k true edges, λ_1^* is given by*

$$\lambda_1^* = \frac{(mk - mN)\tilde{\lambda}_1 + mk\tilde{\lambda}_0}{kN - mN + mk} \tag{4.9}$$

Refer to appendix for the proof.

This result is significant since λ_1 is closely related with many real space topological features, such as the maximum degree, chromatic number, clique number, and extend of branching of the graph [9]. Therefore, our algorithm determines r such that the difference between the reconstructed value $\hat{\lambda}_1$ and the estimated value λ_1^* is minimized. We expect that by preserving λ_1 in the reconstructed graph, many other features can also be well reconstructed.



(a) λ_1 (b) ν_2 (c) Q (d) Transitivity

Figure 5: Original, randomized and reconstructed features for polblog network, r varies from 1 to 200, $k = 0.4m$.

The circled points in Figure 5 plot the r value chosen by our method and the corresponding values for the four features. For λ_1 shown in Figure 5(a), the reconstructed value is close to original value, indicating that the estimator shown in (4.9) accurately matches the original λ_1. For other features, the chosen r value may not be the optimal for those features. However, the reconstructed features are closer to the original value than the randomized one when r is chosen by our method.

5 Empirical Evaluation

In addition to *polblogs* network, we use two network data sets (*polbooks, Enron*) in our evaluation. The *polbooks* network contains 105 nodes and 441 edges. Nodes represent books about US politics sold by the online bookseller Amazon.com, and edges represent frequent co-purchasing of

books by the same buyers.[2] The *Enron* network was built from an email corpus of a real organization over the course covering a 3 years period. We used a pre-processed version of the dataset provided by [18]. This data set contains 252,759 emails from 151 Enron employees, mainly senior managers. We regard there is an edge between node i and j if there are at least 5 emails sent between i and j, which results in 869 edges. The numbers of nodes and edges for three networks are shown in the first row of Table 2.

5.1 Feature Reconstruction We focus on four topological features (λ_1, ν_2, Q, and C) in our evaluation. For each network data set, we first calculate feature values of the original graph and show them in Table 2. We randomize each network data with noise level $\frac{k}{m} = 0.4$. We then apply our low rank approximation based reconstruction algorithm on each randomized graph and calculate the reconstructed feature values from the reconstructed graph. The randomization and reconstruction process repeats 10 times. We report the average results of these 10 rounds in Table 2.

We can observe that perturbation with noise level $\frac{k}{m} = 0.4$ significantly changes the feature values in the randomized graphs. It indicates that edge randomization in general cannot well preserve the graph topological structure. However, for all four features on three network data sets, our reconstructed feature values are much closer to the original ones.

To evaluate accuracy of feature reconstruction, we use the following measure.

DEFINITION 5.1. *For a graph feature f, define reconstruction quality*

$$S_f = 1 - \frac{|f(\hat{A}) - f(A)|}{|f(\tilde{A}) - f(A)|}.$$

$S_f \in (0, 1]$ indicates that the reconstructed feature is closer to the original feature value than the feature value directly calculated from the randomized graph. The larger S_f is, the better the feature is reconstructed. $S_f = 1$ if and only if $f(\hat{A}) = f(A)$, and S_f is close to 1 if $f(\hat{A}) \approx f(A)$.

Table 2 shows the reconstruction quality S_f for these four features on three networks ($k = 0.4m$). We can see that all S_f values are above 0.22 and some S_f values are even close to 1, indicating that the majority of topological structure of the original graph has been reconstructed. We also notice that λ_1 is better reconstructed than the other three features. This is because we use the estimate of λ_1 as our target function when we determine r.

Effect of Noise Level In this experiment, we evaluate how the reconstruction accuracy of features is affected by the magnitude of noise. We set noise level $\frac{k}{m} = 0.2, 0.4, 0.6, 0.8$. We report the feature values of the original data sets ($f(A)$), the randomized feature values under different noise levels ($f(\tilde{A})$), and the reconstructed feature values using our algorithm ($f(\hat{A})$) in Table 3.

For all features, the difference between $f(\tilde{A})$ and $f(A)$ increases as the magnitude of noise increases. For example, $\tilde{\lambda}_1$ is reduced approximately by half from the original value when $k = 0.6m$ for the *polblogs* network. After reconstruction, all reconstructed feature values are much more accurate than those feature values calculated from randomized graphs. For example, even under noise $\frac{k}{m} = 0.6$, our reconstructed transitivity value (C) is 0.15, which is much closer to the original transitivity value (0.23) than the randomized transitivity value (0.06). This result shows that our low rank approximation based reconstruction method can effectively filter out the noise and preserve the topological structure. We can also observe that the difference between $f(\hat{A})$ and $f(A)$ increases when the magnitude of noise increases, indicating that larger noise causes more loss of feature reconstruction quality. For example, the reconstructed transitivity value decreases to 0.09 under noise level $\frac{k}{m} = 0.8$, but it is still better than the randomized transitivity value (0.03).

Table 3: Reconstruction quality for *polblogs* network at different noise levels

	λ_1 (74.08)		ν_2 (0.92)		Q (1.13)		C (0.23)	
$\frac{k}{m}$	rand	recon	rand	recon	rand	recon	rand	recon
0.2	61.43	75.83	0.77	0.84	0.90	1.11	0.16	0.22
0.4	49.38	74.28	0.66	0.75	0.69	0.98	0.10	0.19
0.6	38.39	71.35	0.54	0.62	0.47	0.78	0.06	0.15
0.8	30.56	60.74	0.40	0.48	0.27	0.50	0.03	0.09

5.2 Privacy One question here is that whether attackers can exploit the reconstructed graph $\hat{A}$ to breach the link privacy. If $\hat{A}$ is similar to A at the entry level, attackers may simply use the value of $\hat{a}_{ij}$ as a guess of the original value a_{ij} (the sensitive link between node i and j). If $\hat{A}$ well matches A at the individual entry level, attackers have high confidence about the existence of the true link between node i and j based on the reconstructed $\hat{a}_{ij}$.

To measure the average disclosure risk of all link entries, we use the normalized Frobenius distance defined as

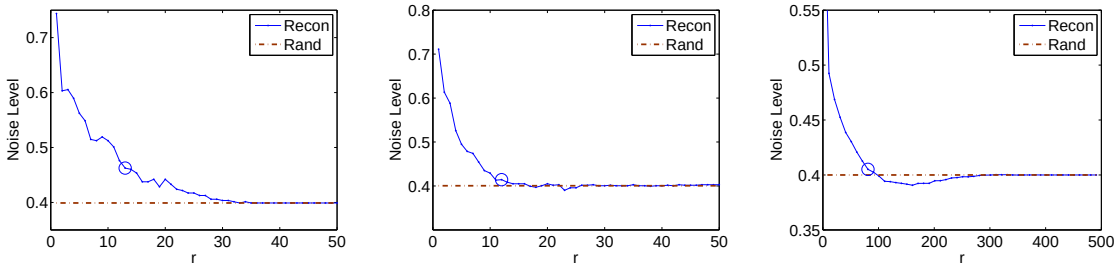
$$d(\hat{A}, A) = \frac{\|\hat{A} - A\|_F^2}{4m}.$$

It is easy to verify that $1 - d(\hat{A}, A) = |\hat{E} \cap E|/|\hat{E}|$. In other words, the larger $d(\hat{A}, A)$ is, the lower the disclosure risk is in the reconstructed graph. $d(\hat{A}, A) = 1$ if and only if no edge from the original graph appears in the reconstructed graph. Similarly, we can measure the disclosure risk of the randomized graph as $d(\tilde{A}, A) \equiv \frac{k}{m}$.

[2] *polbooks* and *polblogs* are available at `http://www-personal.umich.edu/~mejn/netdata/`.

Table 2: The reconstructed features for three data sets ($k = 0.4m$).

	polbooks $(105, 441)$				*Enron* $(151, 869)$				*polblogs* $(1222, 16714)$			
	orig	rand	recon	S_f	orig	rand	recon	S_f	orig	rand	recon	S_f
λ_1	11.9	9.95	12.62	0.65	17.8	14.3	18.3	0.87	74.1	49.5	74.5	0.98
ν_2	0.96	0.72	0.77	0.22	0.89	0.65	0.80	0.63	0.92	0.67	0.76	0.35
Q	0.70	0.45	0.56	0.45	0.56	0.38	0.56	1.00	1.13	0.70	0.99	0.69
C	0.35	0.15	0.20	0.27	0.34	0.15	0.28	0.65	0.23	0.11	0.20	0.75



(a) *polbooks* (b) *Enron* (c) *polblogs*

Figure 6: $d(\hat{A}, A)$ for the three networks, as r varies ($k = 0.4m$).

Figure 6 shows how $d(\hat{A}, A)$ for the three networks changes for different choices of r. We randomize each network data by the noise $k = 0.4m$. For each r, we derive the reconstructed graph $\hat{A}$ by discretizing $\tilde{A}_r$ and calculate the normalized distance $d(\hat{A}, A)$. The circled points plots the $d(\hat{A}, A)$ value when r is chosen by our method. As r approaches n, $d(\hat{A}, A)$ thus converges to k/m because $\hat{A}$ approaches $\tilde{A}$.

One surprising observation is that for both *polbooks* and *Enron* the normalized distance of the reconstructed graph $(d(\hat{A}, A))$ is always above that of the randomized graph $(d(\tilde{A}, A))$ no matter how we choose r as shown in Figure 6(a) and 6(b). This indicates that the reconstructed graph does not incur any further privacy disclosure than the randomized graph. In Section 5.1, we know that the features can be well reconstructed. This is because the risk of privacy disclosure depends on the extent to which $\hat{A}$ matches A individually, while most topological features are overall measures of the graph. It is possible that two graphs have close topological structures but are very different at the individual level, and an accurate reconstruction of features does not necessarily lead to an accurate reconstruction of Frobenius distance. Note that $\hat{A}$ is reconstructed such that it preserves the leading eigenvalue and eigenvectors of A. Therefore, strong structure, which is reflected by the leading eigen-pairs, is preserved in the reconstructed features; and weaker structure indicated by remaining eigen-pairs are neglected along with the noise. Therefore, the reconstruction method can approximate many original topological features at the global level. However, at the individual level, the neglected eigenvalues and eigenvalues can cause many false edges, and the Frobenius norm distance, which accumulates the difference of each entry, can be very large.

However, for *polblogs*, as shown in Figure 6(c), we can observe the normalized distance of the reconstructed graph $(d(\hat{A}, A))$ is a little below that of the randomized graph $(d(\tilde{A}, A))$ for some choices of r. In other words, the reconstructed graph can incur some additional privacy disclosure risks. In the next section, we further investigate what type of graphs may incur additional privacy disclosure risks due to reconstruction.

6 Reconstruction Accuracy on Low Rank Graphs

The phenomenon shown in Section 5.2 is very different from that in the numerical setting. More accurate individual data can be recovered from the randomized numerical data using those point-wise data reconstruction methods based on low rank approximation [11, 15], which jeopardizes data privacy at the individual level.

Our intuition is that there usually exist strong correlations among attributes in the numerical data and the number of attributes is much smaller than the number of tuples. Hence the numerical data U (or its covariance $\mathrm{Cov}(U)$) has a low rank. On the contrary, for most real social networks, their adjacency matrices have very high ranks. For example, all three networks used in our paper have almost full ranks. Our conjecture is that for social networks with low ranks or with a small number of dominant eigenvalues the reconstructed graph can also be close to the original one at the individual entry level.

The difference between the reconstructed graph and the

original graph can be divided into three components:

$$\|A - \hat{A}\|_F = \|(A - A_r) + (A_r - \tilde{A}_r) + (\tilde{A}_r - \hat{A})\|_F$$
$$\leq \|A - A_r\|_F + \|A_r - \tilde{A}_r\|_F + \|\tilde{A}_r - \hat{A}\|_F. \quad (6.10)$$

$\|A - A_r\|_F$ denotes the low rank approximation error that is determined by those excluded non-significant eigen-pairs; $\|A_r - \tilde{A}_r\|_F$ denotes the randomization error that is determined by the noise added in the subspace spanned by the first r eigenvectors; and $\|\tilde{A}_r - \hat{A}\|_F$ denotes the discretization error when we convert the real matrix $\tilde{A}_r$ to the 0-1 matrix $\hat{A}$. To decrease $\|A - A_r\|_F$, we tend to choose a large r value. However, a large r value introduces more noise in the projected spectral space, increasing the randomization error $\|A_r - \tilde{A}_r\|_F$.

Hence, if a graph A can be well approximated by A_r with a small r value, both the low rank approximation error ($\|A - A_r\|_F$) and the randomization error ($\|A_r - \tilde{A}_r\|_F$) could be small. In this case, $\tilde{A}_r \approx A_r \approx A$, and $\tilde{A}_r$ is already close to a 0-1 matrix, which then further reduces the discretization error $\|\hat{A} - \tilde{A}\|_F$.

For three network data sets used in our paper, we can derive their minimum r values such that $\frac{\|A-A_r\|_F^2}{\|A\|_F^2} \leq \tau$. When $\tau = 0.05$, we have $r = 54$ $(0.51n)$ for *polbooks*, $r = 64$ $(0.42n)$ for *Enron*, and $r = 348$ $(0.28n)$ for *polblogs* network. Since all r values are large, the difference between the reconstructed graph and the original graph at the individual level ($\|A - \hat{A}\|_F$) is still significant, indicating the individual privacy is well protected in the reconstructed graph. However, the feature values can still be well reconstructed. This is because those non-significant eigen-pairs do not contribute much to the global topological structure although they may significantly affect the Frobenius distance.

To verify our proposition, we construct a series of synthetic graphs H_t $(t = 2, 5, 10, 50, 100, 200)$ from the *polblog* network. We first calculate $A_t = \sum_{i=1}^{t} \lambda_i \boldsymbol{x}_i \boldsymbol{x}_i^T$ and regard its discretized version (using (3.2)) as H_t. We expect that these synthetic graphs H_t have a small number of dominant eigen-pairs. When $\tau = 0.05$, their minimum r values are listed in Table 4. For example, for graph H_2, the number of dominant eigen-pairs is 16, which is much less than that of the original graph A. As a result, when we apply our low rank approximation based reconstruction algorithm on H_2, the normalized distance is only 0.05, indicating that 95% of original edges are recovered in the reconstructed graph. We can also observe that as t increases, the number of dominant eigen-pairs also increases, and the reconstruction accuracy at the individual entry level decreases. For example, when $t = 200$, the normalized distance is 0.39, which is approximately equal to that of the randomized graph.

Figure 7 shows the normalized distance between the reconstructed graph $\hat{H}_t$ and the original graph H_t for different choices of r. The circled points represent the distance values when r is chosen via our method. We can see that, for graphs H_5, H_50 and H_100, the normalized distance values ($d(\hat{H}_t, H_t)$) are smaller than that of the randomized graph (k/m) for the majority r values. In particular, the normalized distance values on H_5 could reach as low as 0.1. As t increases, the curve of the normalized distance values on H_t approaches the curve of the original graph A, as shown in Figure 7. This phenomenon supports our conjecture: for those graphs with a small number of dominant eigen-pairs, reconstruction can accurately recover the original individual entries, which may seriously jeopardize data privacy.

Table 4: Normalized Frobenius distance of reconstruction for the synthetic graphs from *polblogs* ($k = 0.4m$)

	H_2	H_5	H_{10}	H_{50}	H_{100}	H_{200}	A
min r	16	54	95	179	231	299	348
$d(\hat{H}_t, H_t)$	0.05	0.10	0.15	0.25	0.32	0.39	0.40

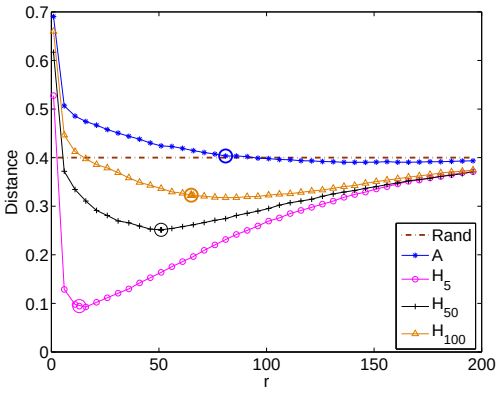


Figure 7: The normalized Frobenius distance of the synthetic graphs

We also calculate the feature values of the reconstructed graphs ($f(\hat{H}_t)$) and compare them with the original feature values ($f(H_t)$). Table 5 shows the feature reconstruction qualities (S_f) for different features. For all features, S_f values are also higher for those synthetic graphs constructed using small t values. This is because we can have an accurate reconstruction on the individual entry level for those graphs, and hence many global features can be accurately reconstructed. Note that, as shown for the three real networks, the inverse direction is generally not guaranteed. In summary, the dominance of the leading eigen-pairs of a graph plays an important role in reconstructing individual entries as well as global features.

Table 5: Feature reconstruction quality of the synthetic graphs from *polblogs* ($k = 0.4m$)

	H_2	H_5	H_{10}	H_{50}	H_{100}	H_{200}	A
S_{λ_1}	1.00	1.00	0.99	0.99	0.98	1.00	1.00
S_{ν_2}	0.99	0.96	0.89	0.67	0.54	0.43	0.35
S_Q	1.00	0.98	0.95	0.83	0.73	0.69	0.66
S_C	0.98	1.00	0.98	0.95	0.83	0.78	0.73

7 Conclusion and Future Work

In this paper, we have presented a low rank approximation based reconstruction algorithm, which can well recover feature values from the randomized network data. We have shown the close relationship between graph topological structure and spectral spaces determined by eigen-pairs of the adjacency matrix. We have also presented a novel solution to determine the optimal rank r in reconstruction. Our empirical evaluation results showed that accurate feature values can still be recovered from the randomized graphs even with the large magnitude of noise. One surprising finding is that, for most social networks, the reconstructed networks do not incur further disclosure risks of individual privacy than the released randomized graphs. Our investigation showed that only networks with low ranks or a small number of dominant eigenvalues may incur further privacy disclosure due to reconstruction.

There are some other aspects of this work that merit further research. Since how to preserve utility is an important issue in privacy-preserving social network analysis, evaluations using more large social networks and more topological features (as well as workload-aware metrics) are needed. It is important to develop approaches that adequately quantify levels of information loss of graph data due to randomization.

We are interested in comparing with other various edge based randomization strategies. In particular, we will study whether a similar low rank approximation based reconstruction method can be derived for the *Random Switch* strategy. It is also our conjecture that it is very hard, if not impossible, to figure out reconstruction methods on the released randomized data using K-anonymity schemes. This is because in K-anonymity based modification schemes, modified edge entries are not randomly chosen. For example, the K-degree scheme [17] examines the degree sequence of nodes and chooses a subset of nodes (that violates the K-degree anonymity property) for edge modification. We will compare various randomization strategies in terms of the tradeoff between privacy and utility.

In the setting of numerical data, distributions of U can be approximately reconstructed from the perturbed data $\tilde{U}$ using distribution reconstruction approaches (e.g., [2, 3]) when some a-priori knowledge (e.g., distribution, statistics etc.) about the noise V is available. Specifically, Agrawal and Aggawal [2] provided an expectation-maximization (EM) algorithm for reconstructing the distribution of the original data from perturbed observations. However, it is unclear whether similar distribution reconstruction methods can be derived for network data. This is because 1) it is hard to define distribution for network data; and 2) the randomization mechanism for network data is based on the positions of randomly chosen edges rather than the independent random additive values for all entries for numerical data. We would emphasize that both distribution reconstruction methods and point-wise reconstruction methods on purely randomized graphs need further investigations. So more accurate analysis can be conducted on reconstructed graphs where individual privacy can still be preserved.

We will investigate how well randomization protects privacy (identity, link privacy, and attribute privacy) in general social networks when adversaries exploit various complex background knowledge in their attacks. How to model various background knowledge and quantify disclosures when complex attacks are used needs to be investigated. Finally, the scalability issue needs to be studied and empirical evaluations need to be conducted on large social networks.

Acknowledgments

This work was supported in part by U.S. National Science Foundation IIS-0546027 and CNS-0831204.

References

[1] L. Adamic and N. Glance. The political blogosphere and the 2004 us election: divided they blog. In *Proceedings of the WWW-2005 Workshop on the Weblogging Ecosystem*, 2005.

[2] D. Agrawal and C. Agrawal. On the design and quantification of privacy preserving data mining algorithms. In *Proceedings of the 20th Symposium on Principles of Database Systems*, 2001.

[3] R. Agrawal and R. Srikant. Privacy-preserving data mining. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pages 439–450. Dallas, Texas, May 2000.

[4] L. Backstrom, C. Dwork, and J. Kleinberg. 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*, pages 181–190, New York, NY, USA, 2007. ACM Press.

[5] S. Bhagat, G. Cormode, B. Krishnamurthy, and D. Srivastava. Class-based graph anaonymization for social network data. In *Proc. of 35th International Conference on Very Large Data Base*, 2009.

[6] A. Broder, R. Kumar, F. Maghoul, P. Raghavan, S. Rajagopalan, R. Stata, A. Tomkins, and J. Wiener. Graph structure in the web: experiments and models. In *International World Wide Web Conference*. ACM Press, New York, NY., 2000.

[7] A. Campan and T. M. Truta. A clustering approach for data and structural anonymity in social networks. In *PinKDD*, 2008.

[8] G. Cormode, D. Srivastava, T. Yu, and Q. Zhang. Anonymizing bipartite graph data using safe groupings. In *Proc. of VLDB08*, pages 833–844, 2008.

[9] D. Cvetkovic, P. Rowlinson, and S. Simic. *Eigenspaces of Graphs*. Cambridge University Press, 1997.

[10] L. da F. Costa, F. A. Rodrigues, G. Travieso, and P. R. V. Boas. Characterization of complex networks: A survey of measurements. *Advances In Physics*, 56:167, 2007.

[11] S. Guo, X. Wu, and Y. Li. Determining error bounds for spectral filtering based reconstruction methods in privacy preserving data mining. *Knowl. Inf. Syst.*, 17(2):217–240, 2008.

[12] S. Hanhijarvi, G. C. Garriga, and K. Puolamaki. Randomization techniques for graphs. In *Proc. of the 9th SIAM Conference on Data Mining*, 2009.

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

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

[15] Z. Huang, W. Du, and B. Chen. Deriving private information from randomized data. In *Proceedings of the ACM SIGMOD Conference on Management of Data*. Baltimore, MA, 2005.

[16] H. Kargupta, S. Datta, Q. Wang, and K. Sivakumar. On the privacy preserving properties of random data perturbation techniques. In *Proc. of the 3rd Int'l Conf. on Data Mining*, pages 99–106, 2003.

[17] K. Liu and E. Terzi. Towards identity anonymization on graphs. In *Proceedings of the ACM SIGMOD Conference*, Vancouver, Canada, 2008. ACM Press.

[18] J. Shetty and J. Adibi. The Enron email dataset database schema and brief statistical report. *Information Sciences Institute Technical Report, University of Southern California*, 2004.

[19] J. Shi and J. Malik. Normalized cuts and image segmentation. In *CVPR '97: Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition (CVPR '97)*, page 731, Washington, DC, USA, 1997. IEEE Computer Society.

[20] G. W. Stewart and J. guang Sun. *Matrix Perturbation Theory*. Academic Press, 1990.

[21] Y. Wang, D. Chakrabarti, C. Wang, and C. Faloutsos. Epidemic spreading in real networks: An eigenvalue viewpoint. *Proceedings of the 22nd International Symposium on Reliable Distributed Systems*, 2003.

[22] X. Ying and X. Wu. Randomizing social networks: a spectrum preserving approach. In *Proc. of the 8th SIAM Conference on Data Mining*, April 2008.

[23] X. Ying and X. Wu. Graph generation with prescribed feature constraints. In *Proc. of the 9th SIAM Conference on Data Mining*, 2009.

[24] X. Ying and X. Wu. On link privacy in randomizing social networks. In *PAKDD*, 2009.

[25] X. Ying and X. Wu. On randomness measures for social networks. In *SDM*, 2009.

[26] E. Zheleva and L. Getoor. Preserving the privacy of sensitive relationships in graph data. In *PinKDD*, pages 153–171, 2007.

[27] B. Zhou and J. Pei. Preserving privacy in social networks against neighborhood attacks. In *Proceedings of the 24th International Conference on Data Engineering (ICDE'08)*, 2008.

[28] L. Zou, L. Chen, and M. T. Özsu. k-automorphism: A general framework for privacy preserving network publication. In *Proc. of 35th International Conference on Very Large Data Base*, 2009.

A Proof of Result 1

Define $\lambda_0 = \boldsymbol{x}_1^T(\mathbf{1} - I - A)\boldsymbol{x}_1$ and $\tilde{\lambda}_0 = \tilde{\boldsymbol{x}}_1^T(\mathbf{1} - I - \tilde{A})\tilde{\boldsymbol{x}}_1$. Since $\tilde{\lambda}_1 = \tilde{\boldsymbol{x}}_1^T A \tilde{\boldsymbol{x}}_1$, we have

$$\mathbf{E}(\tilde{\lambda}_1) = \mathbf{E}(\tilde{\boldsymbol{x}}_1^T \tilde{A} \tilde{\boldsymbol{x}}_1) \approx \boldsymbol{x}_1^T \, \mathbf{E}(\tilde{A}) \boldsymbol{x}_1. \tag{1.11}$$

We adopt the assumption that $\tilde{\boldsymbol{x}}_1 \approx \boldsymbol{x}_1$ in establishing the second equality of (1.11). Since in *Rand Add/Del* every existing (non-existing) edge of A has the same probability to be add (deleted), we have $\mathbf{E}(\tilde{a}_{ij}) = \frac{m-k}{m}$ if $a_{ij} = 1$, and $\mathbf{E}(\tilde{a}_{ij}) = \frac{k}{N}$ if $a_{ij} = 0$ and $i \neq j$, where $N = \binom{n}{2} - m$, i.e.,

$$\mathbf{E}(\tilde{A}) = \frac{m-k}{m} A + \frac{k}{N}(\mathbf{1} - I - A).$$

Continue with (1.11), we have

$$\begin{aligned} \mathbf{E}(\tilde{\lambda}_1) &= \frac{m-k}{m} \boldsymbol{x}_1^T A \boldsymbol{x}_1 + \frac{k}{N} \boldsymbol{x}_1^T (\mathbf{1} - I - A) \boldsymbol{x}_1 \\ &= (1 - \frac{k}{m})\lambda_1 + \frac{k}{N}\lambda_0. \end{aligned}$$

Similarly, we can calculate $\mathbf{E}(\tilde{\lambda}_0)$ and have

$$\begin{aligned} \mathbf{E}(\tilde{\lambda}_1) &= (1 - \tfrac{k}{m})\lambda_1 + \tfrac{k}{N}\lambda_0, \\ \mathbf{E}(\tilde{\lambda}_0) &= (1 - \tfrac{k}{N})\lambda_1 + \tfrac{k}{m}\lambda_0. \end{aligned} \tag{1.12}$$

In estimating λ_1, we substitute $\mathbf{E}(\tilde{\lambda}_1)$ and $\mathbf{E}(\tilde{\lambda}_0)$ with observed $\tilde{\lambda}_1$ and $\tilde{\lambda}_0$, and solving (1.12) for λ_0 and λ_1, we can get the moment estimator of λ_1 is given by:

$$\lambda_1^* = \frac{(mk - mN)\tilde{\lambda}_1 + mk\tilde{\lambda}_0}{kN - mN + mk}.$$