An Approach for Personalized Labelled Privacy Preservation on Social Network Data

Kodali Anupama, Yelisela Rajesh

1Dept. of CSE, Andhra Loyola Institute of Engineering and Technology, Krishna District, AP, India
2Dept. of CSE Andhra Loyola Institute of Engineering and Technology, AP, India

Abstract
Preserving privacy in publishing social network data becomes an important concern. With some local knowledge about individuals in a social network, an adversary may attack the privacy of some victims easily. In this paper, we take an initiative toward preserving privacy in social network data. A problem with many secure/trusted computer systems intended for usage by the Department of Defense is that data often becomes overclassified and is not labeled accurately upon output. Therefore, we are providing security based on the labels assigned for users in the social networks. To this aim, the algorithms transform the original graph into a graph in which nodes are sufficiently indistinguishable. The algorithms designed to do so while losing as little information and while preserving as much utility as possible.

Keywords
Privacy, Social Network Data, Sensitive Labeling, Publishing

I. Introduction
Sensitive information about users of the social networks should be protected. The challenge is to devise methods to publish social network data in a form that affords utility without compromising privacy [1]. Previous research has proposed various privacy models with the corresponding protection mechanisms that prevent both inadvertent private information leakage and attacks by malicious adversaries. Users entrust social networks such as Facebook and LinkedIn with a wealth of personal information such as their age, address, current location or political orientation. The social networks are modeled as graphs in which users are nodes and features are labels. Labels annotated to thenodes show the locations of users. Each letter represents a city name as a label for each node.

The privacy issue arises from the disclosure of sensitive labels. One might suggest that such labels should be simply deleted. Still, such a solution would present an incomplete view of the network and may hide interesting statistical information that does not threaten privacy. These early privacy models are mostly concerned with identity and link disclosure. The social networks are modeled as graphs in which users are nodes and social connections are edges.

II. Review of Literature
Literature survey is the important step to be considered in software developing process. Here, we study some previous literature papers about sensitive labeling.

A. Privacy Attacks Using Published Social Network Data
As more and more rich social media, popular online social networking sites, and various kinds of social network analyzing and mining techniques are available, privacy in social networks becomes serious concern [6,9,2], particularly when social network data is published.

An adversary may intrude privacy of some victims using the published social network data and some background knowledge. Importantly, many of the richest emerging sources of social network data come from settings such as e-mails, instant messages or telephone communication. Users have strong expectations of privacy on such data. When social network data is made public in one way or another, it is far from sufficient to protect privacy by simply replacing the identifying attributes such as name of individuals by meaningless unique identifiers.

B. Challenges in Anonymizing Social Network Data
Privacy preservation on relational data has been studied extensively. A major category identify individuals by joining a published table containing sensitive information with some external tables modeling background knowledge of attackers. To battle the re-identification attacks, the mechanism of k-anonymity was proposed [3]. Specifically, a data set is said to be k-anonymous 

Privacy preservation on relational data has been studied extensively. A major category identify individuals by joining a published table containing sensitive information with some external tables modeling background knowledge of attackers. To battle the re-identification attacks, the mechanism of k-anonymity was proposed [3]. Specifically, a data set is said to be k-anonymous 

Machanava- jjhala et al [8] showed that a k-anonymous table may still have some subtle but severe privacy problems due to the lack of diversity in the sensitive attributes. In particular, they showed that, the degree of privacy protection does not really depend on the size of the equivalence classes on quasi-identifier attributes which contain tuples that are identical on those attributes. Instead, it is determined by the number and distribution of distinct sensitive values associated with each equivalence class. To overcome the weakness in k-anonymity, they propose the notion of l-diversity [8].

Xiao and Tao [11] prove that l-diversity always guarantees stronger privacy preservation than k-anonymity. Though several important models and many algorithms have been proposed to preserve privacy in relational data, most of the existing studies can deal with relational data only. Those methods cannot be applied to social network data straightforwardly. Anonymizing social network data is much more challenging than anonymizing relational data. First, it is much more challenging to model background knowledge of adversaries and attacks. Second, it is much more challenging to measure the information loss in anonymizing social network data than that in anonymizing relational data. Typically, the information loss in an anonymized table can be measured using the sum of information loss in individual tuples.

Given one tuple in the original table and the corresponding anonymized tuple in the released table, we can calculate the distance between the two tuples to measure the information loss at the tuple level. However, a social network consists of a set of vertices and a set of edges. It is hard to compare two social networks by comparing the vertices and edges individually. Two social networks having the same number of vertices and the same number of edges may have very different network-wise properties such as connectivity, betweenness, and diameter. Thus,
there can be many different ways to assess information loss and anonymization quality.

C. Randomized Spectrum Preserving Method
Ying and Wu [14] tackled the problem as [5] by considering randomly adding/deleting edges or randomly switching edges. Instead of designing specific randomization algorithms, Ying and Wu [14] analyzed the effect of randomization in protecting attacks. The spectrum of a graph is defined as the set of eigen values of the adjacency matrix of the graph. The eigen values of a network are connected to important topological properties such as diameter, presence of cohesive clusters, long paths and bottlenecks, and randomness of the graph. Ying and Wu showed that the spectrum property has close relation with many graph characteristics and can provide global measures for some network properties. Furthermore, Ying and Wu investigated [14] the spectrum of networks. A natural idea for graph anonymization is to consider whether a graph can be perturbed without significantly changing one or some particular eigenvalues. If so, the approach is probable to better preserve structural characteristics. By considering the change of spectrum in the randomization process, the proposed spectrum preserving [14] approach can outperform the simple edge randomization methods. The algorithm can determine which edges should be added, removed or switched so that the change of the eigen values can be under control.

III. Related Work
The first necessary anonymization technique in both the contexts of micro and network data consists in removing identification. This naive technique has quickly been recognized as failing to protect privacy. They proposed a method that group nodes and anonymizes the neighborhoods of nodes in the same group by generalizing node labels and adding edges.

Modules:
1. Grouping of nodes by k-means algorithm. Traffic analysis of data sent over the network.
2. Removing the nodes by adding actual noisy node with different labels and evaluating the overlapping.

Definition 1: The neighborhood information of node v comprises the degree of v and the labels of v’s neighbors.

Definition 2: (l-sensitive-label-diversity) for each node v that associates with a sensitive label, there must be at least l-1 other nodes with the same neighborhood information, but attached with direct sensitive labels.

IV. Proposed System
The current trend in the Social Network is its not giving the privacy about user profile views. The method of data sharing or (Posting) is taking more time, not under the certain condition of displaying sensitive and non-sensitive data. Here, we extend the existing definitions of modules and we introduced the sensitive and non-sensitive label concept in our project.

We want to group nodes with similar neighborhood information as possible so that we can change as few labels as possible and add as few noisy nodes as possible. We propose algorithm, Global-similarity-based Indirect Noise Node (GINN) that does not attempt to heuristically prune the similarity computation as the other two algorithms, Direct Noisy Node Algorithm (DNN) and Indirect Noisy Node Algorithm(INN) do.

A. Algorithm GINN
The algorithm Global-similarity-based Indirect Noise Node starts out with group formation, during which all nodes that havenot yet been grouped are taken into consideration in clustering like fashion. In the first run, two nodes with the maximum similarity of their neighborhood labels are grouped together.

Then nodes having the maximum similarity with any node in the group are clustered into the group till the group has l nodes with different sensitive labels. Therefore the algorithm proceeds to create the next group. After having formed these groups, we need to ensure that each group’s members are indistinguishable in terms of neighborhood information. Thus, neighborhood labels are modified after every grouping operation.

Edge insertion is to complement both a missing label and insufficient degree value. A node is linked to an existing nearby node with that label. The labels of two or more nodes coalesce their values to a single super-label value, being the union of their values. This approach maintains data integrity, in the sense that the true label of a node is included among the values of its label super-value. After such edge insertion and label fusion operations.

GINN ALGORITHM
Input: graph G(V,E,L,L'), Parameter l;
Result : Modified Graph G'

While \( V_{\text{left}} > 0 \) do
  if \( V_{\text{left}} > l \) then
    Compute pairwise node similarities.
    Group g ← v1,v2 with Max_similarity;
    Modify neighbors of G;
  else if \( |G| < l \) do
    Dissimilarities(V_{\text{left}}, G);
    Group g ← v with MaxSimilarities;
    Modify neighbors of G without actually adding noisy nodes;
  else if \( |V_{\text{left}}| < l \) then
    for each v ∈ V_{\text{left}} do
      similarities(v,Gs);
    Gmax_similarity ← V;
    Modify neighbors of Gmax_similarities without actually adding noisy nodes;
    Add Expected noisy nodes;
    Return G'

If two nodes are expected to have the samelabels of neighbors and are within two hops onlyone node is added. In other words, we merge some noisy nodes with the samelabel, thus resulting in fewer noisy nodes.

B. Page Rank Algorithm
In order to give ranking to sensitive and non-sensitive labels, we used page ranking algorithm. Here initially count of all posts is set to zero. If any user clicks on any post then present count of the post gets incremented by 1. Such that ranking is given to sensitive and non-sensitive posts.

C. Information Loss
In view of utility of released data, we aim to keep information loss low. Information Loss in this case contains both structure information loss and label information loss.

Information Loss = (No. of Non sensitive posts/No. of Posts) + 1
V. Experimental Evaluation
We evaluate our approaches using both synthetic and real data sets. All of the approaches have been implemented in Python. We use three data sets. The first dataset is a network of hyperlinks between weblogs on US politics. The second data set that we use is generated from the Facebook dataset proposed in. The third data set that we use is a family of synthetic graphs with varying number of nodes.

A. Data Utility
We compare the data utilities we preserve from the original graphs, in view of measurements on degree distribution, label distribution, degree centrality clustering coefficient, average path length, graph density and radius.

B. Algorithm Scalability
We measure the running time of the methods for a series of synthetic graphs with varying number of nodes in our third dataset. Fig. 2 presents the running time of each algorithm as the number of nodes increases. Algorithm DNN is faster than the other two algorithms, showing good scalability at the cost of large noisynodes added.

C. Information Loss
We calculated the information loss based on the uploaded posts. It is graphically represented as follows. Here X-axis represents the social network data. And Y-axis represents number of users.

VI. Conclusion
In this paper we investigate the protection of private label information in social network data publication. We consider graphs with rich label information which are categorized to be either sensitive or non-sensitive. We assume that adversaries possess prior knowledge about a node’s degree and the labels of its neighbors and can use that to infer the sensitive labels of targets. We suggest a model for attaining privacy while publishing the data in which node labels are both part of adversaries background knowledge and sensitive information that is to be protected. We accompany our model with algorithms that transform network graph before published, so as to limit adversaries’ confidence about sensitive label data.

References
[1] Yi Song, Panagiotis Karras, Qian Xiao, Stephane Breassan, "Sensitive Label Privacy Protection on Social Network Data", IEEE Transactions on Knowledge and Data Engineering, 2013


Kodali Anupama is pursuing Master of Technology (M.Tech) in Computer Science and Engineering (CSE) from Andhra Loyola Institute of Engineering and Technology, Vijayawada, in Andhra Pradesh, India. This institute is affiliated to JNTUK, Kakinada. Apart from the academic excellence, she has been involving into the various research activities too.

Yelisela Rajesh, received his M.Tech from Koneru Lakshmaiah University (KLU), a deemed university, in Green Fields, Vaddeswaram, Guntur District, Andhra Pradesh, India. He is working in the department of Computer Science and Engineering as Assistant Professor in Andhra Loyola Institute of Engineering and Technology, Vijayawada, affiliated to JNTUK University, Kakinada, India. He has 3 years of teaching experience. He is very interested in research activities. He has good number of publications in the reputed international journals.