Data Anonymization via Sensitive Labels in Social Networks

Siddeni Reshmi¹, Bhaludra Raveendranadh Singh², S.Sunanda³, Moligi Sangeetha⁴

¹ pursing M.Tech (CSE) ² Principal 3Professor (CSE) ⁴Professor & HOD (CSE)

Visvesvaraya College of Engineering and Technology (VCET), M.P.Patelguda, Ibrahimpatnam (M), Ranga Reddy (D)-501510, India

Abstract— Privacy is one of the major concerns when publishing or sharing social network data for social science research and business analysis. Recently, researchers have developed privacy models similar to k-anonymity to prevent node re-identification through structure information. However, even when these privacy models are enforced, an attacker may still be able to infer one’s private information if a group of nodes largely share the same sensitive labels (i.e., attributes). In other words, the label-node relationship is not well protected by pure structure anonymization methods. Furthermore, existing approaches, which rely on edge editing or node clustering, may significantly alter key graph properties. In this paper, we define a k-degree-l-diversity anonymity model that considers the protection of structural information as well as sensitive labels of individuals. We further propose a novel anonymization methodology based on adding noise nodes. We develop a new algorithm by adding noise nodes into the original graph with the consideration of introducing the least distortion to graph properties. Most importantly, we provide a rigorous analysis of the theoretical bounds on the number of noise nodes added and their impacts on an important graph property. We conduct extensive experiments to evaluate the effectiveness of the proposed technique.

1. INTRODUCTION

Now a days there are tremendous growth in social medias like Facebook, Linkedin, mane one thinks that these social sites are very useful to get important information share their his private information with their friends and keep in touch with their friends and relatives. But due to this people are getting addicted to use these sites regularly. Since they are uploading their private information and they don’t want to reveal their private information hence to provide security to their private and sensitive information and at the same time prevent the utility of social network data becomes a very challenging topic. In this paper we are considering a graph model where each vertex in that graph is considered as a sensitive label.

A structural attack is an attack which uses the structure information like the degree and the sub-graph of a node and to identify the nodes. To prevent such attacks the published graph should be satisfy k-anonymity algorithm. The goal is to publish a such type of social graph, which always has minimum k candidates in different attack scenarios in order to protect privacy. A graph is k-degree anonymous if and only if for any node in this graph, there exist at least k -1 other nodes with the same degree. If the attacker knows that one person has three friends in the graph, he can quickly know that node 2 is that person and the related attributes of node 2 are disclosed. k-degree anonymity can be used to prevent such structure attacks. However, in most of the application, a social network is the application where each node has sensitive attributes should be published. For example, a graph might contain the user salaries which are sensitive. In this case, k-degree alone is not sufficient to prevent the inference of sensitive attribute of individuals.

Current approaches for protecting graph privacy can be classified into two categories: clustering and edge editing. Clustering is to merge a sub-graph to one super node, which is somewhat unsuitable for sensitive labeled graphs, since when a group of nodes are combined into one super node, the node-label relationship have been lost. Edge-editing methods keep the node in the original graph unchanged and only add or delete or swap the edges.

To overcome this issue we are proposing a novel approach to preserve important graph properties. In such a way that the distances between nodes to add certain “Noise” nodes into a graph. Most of the social network satisfies power law distribution. It means there exist a large no of low degree vertices in the graph which can be used to hide added noise nodes from being re-identified.

Our privacy preserving goal is to prevent an attacker from re-identifying a user and finding the fact that a certain user has a specific sensitive value. To get this scheme, we define a k-degree-l-diversity (KDLD) model for safely publishing a labeled graph, and then develop the corresponding graph anonymization algorithms with the least distortion to the properties of the original graph, such as degree and distance between nodes.

2. KDLD SEQUENCE GENERATION

To generate KDLD sequence, the triples in P should be divided into groups. All the respective nodes in the same group will be adjusted to have the same degree. We propose two algorithms for this problem Algorithm K-L-BASED and
Algorithm L-K-BASED. The algorithms tend to put the nodes with similar degrees into the same group to reduce the degree changes. Let us consider the degree sequence is \( P \) of the original graph, algorithm K-L-Based selects the first \( k \) elements in \( P \) as a group. Then we keep on merging the next element into the current group until the \( l \)-diversity constraint is satisfied.

Different from Algorithm K-L-BASED which checks the k-degree first, the Algorithm L-K-BASED tries to satisfy the \( l \)-diversity first. If we use ‘t’ to represents the position number of each element in \( P \). Each time Algorithm L-K-BASED picks \( l \) non grouped \( P[t] \) in \( P \) that do not share any common labels with the minimum \( C \). Minimum \( C \) guarantees the algorithm select \( l \) nodes with most similar degrees in the non-grouped nodes. The algorithm then continues merging \( P[t] \) with minimum \( t \) in the remaining elements until we get \( k \) items. Then similar to Algorithm K-L-BASED, we use the similar cost function to determine if the next element should be merged or not. We designed a heuristic algorithm for 1D \( l \)-diversity models, which do not have \( k \)-anonymous requirement. For distinct \( l \)-diversity with \( k = 1 \), our LK-based algorithm follows the same logic as their heuristic algorithm.

Every nodes of the same group will be adjusted to have the same degree in the next graph construction phase. We can take the mean degree of a group as its target degree. This target degree is also used to find the actual cost in Algorithm K-LBASED and Algorithm L-K-BASED.

3. COMPLEX \( l \)-DIVERSITY MODEL

Except distinct \( l \)-diversity model, the Machanavajjhala proposed two other \( l \)-diversity models: entropy \( l \)-diversity request and recursive \( (c, l) \)-diversity. Entropy \( l \)-diversity requests tighter privacy constraints. However, it is too much restrictive for the practical purpose. Recursive \( l \)-diversity has a more relaxed condition.

If the generated KDLD sequence satisfies recursive \( (c, l) \)-diversity and all the same degree groups also satisfy \( (c, l) \)-diversity after adding noise nodes, our algorithm works for recursive \( (c, l) \)-diversity. So, the key points are to modify the KDLD sequence generation algorithm and graph construction.

Algorithm 1.

For each same degree group \( C \) do:

Group \( C_0 \) = grop of original nodes in \( C \); Group \( C_n \) = grop of noise nodes in \( C \);

Array \( N_0 \) = sensitive labels on the nodes in \( C_0 \);

Array \( N_n \) = 10’s occurrence numbers in \( C_0 \); Array \( N_n \) = an array with size \( |l_0| \);

for \( i=0;i<l_0; i++ \) do;

While \( N_n+N_0 \) satisfies recursive \( (e_i) \)-diversity) do

Compute \( \min \) where \( N_n[\min]+N_0[\min] \) is the minimum value in \( N_n+N_0 \);

3.1 Recursive \( (c, l) \)-DIVERSITY

We have used algorithm-1 to assign sensitive labels to noise nodes. For a same degree group the basic way to assign labels to new nodes added into the group is : 1. Assign labels to new nodes follows the label distribution of the original nodes in this group; 2. If the \( (c, l) \)-diversity is not satisfied, and assign the least frequent label in current group of a noise node who is assigned with the most frequent label (By doing this, the most quick label’s occurrence number is decreased by 1 and the least frequent label’s occurrence number is increased by 1. We continuously do this until the new group satisfies the recursive \( (c, l) \)-diversity.

3.2 KDLD SEQUENCE GENERATION FOR RECURSIVE \( (c, l) \)-DIVERSITY

We design algorithm 2 to generate the KDLD sequence for recursive \( (c, l) \)-diversity. The basic idea to put the nodes with similar degree together to reduce the cost. Note that whenever we use the pure edge-editing method to construct the published graph, the Safety Grouping Condition is only condition which contains the first two constraints. Then, we recursively check the next element. If this element has...
degree d, we directly add it into the current group C. Otherwise, we check whether its label is a label that appears top L - 1 times in C. If not, we add this element into C. After adding one element, we remove this element out of P and start check from the head of P since the label appearance numbers might be changed. We can use this method to increase the size of C until C satisfies the Safety Grouping Condition. After getting a safety group, we copy it into $P_\text{new}$ and start to again construct the next group.

4. EXPERIMENTS

In this session we are testing our algorithm on the distinct l-diversity. We can also conduct corresponding experiment for recursive (c, l)-diversity model. The testing results show the effectiveness of our algorithm.

4.1 More Utilities

We have to first test that how well the published graph represents the actual graph. In order to measure the changes in the actual graph, except APL, we examine another two utilities : Average Change of Sensitive Label Path Length (ACSPL) and Remaining ratio of top influential users (RRTI). ACSPL: In order to measure the connections between any two sensitive labels (including the same label), we can define the average path length between any two path length is defined as:where M is the number of peak unique sensitive attribute labels.

1. RRTI: One important data mining task from a graph is to find the top influential users (experts) in it. We can test the remain ratio of top influential users to show how the published graph preserves to this utility.

The larger RRTI is, the better the published graph preserves the information in the original graph. We use the PageRank algorithm to compute the users’ effective values.

5. RELATED WORK

Simply deleting the sensitive labels from the social network is not guaranty of privacy. The unique patterns, such as node degree or sub-graph to special nodes, can be used to re-identify the nodes. The attack which uses certain background knowledge to re-identify the nodes/links in the published graph is called as Passive attack. There we are proposing two models to publish a privacy preserved graph: edge-editing-based model and clustering-based model. The edge editing-based model is to add or delete edges to make the graph satisfy certain properties according to the privacy requirements. Clustering-based model is to cluster “similar” nodes together to form super nodes. Every super node represents several nodes which are also called a “cluster.” Then, the links between nodes are represented as the edges between super nodes which is called “super edges.” Every super edge might represents more than one edge in the original graph. So we call the graph that only contains super nodes and super edges as a clustered graph. Except “passive attack” there is one another type of attack in social network which is called as “active attack”. It is actively embed special sub-graphs into a social network when this social network is collecting users data. The attacker can attack the users who are connected with the embedded sub-graphs by re-identifying these special sub-graphs in the published graph. One method to prevent the active attack is to find the fake nodes added by attackers and remove them before publishing the data. Shrivastava et al. proposed an algorithm that can identify fake nodes based on the triangle probability difference between normal nodes and fake nodes. Ying et al. proposed another method, which are using spectrum analysis to find the fake nodes. To publish a new graph that is potentially changed by Random Link Attack, hence the publisher can use a two step mechanism. In first step, the graph is filtered by the methods introduced by Backstrom et al. or Shrivastava et al. Then, he can generate the published graph using our model from the filtered graph.

6. CONCLUSION

Here we proposed a k-degree-l-diversity algorithm for privacy preserving social network data publishing. We have implemented both distinct l-diversity and recursive (c, l)-diversity. To achieve our requirement of k-degree-l-diversity we have designed a noise node adding algorithm to create a new graph from the original graph with the constraints of introducing fewer distortion from the original graph. We give a thorough analysis of the theoretical bounds on the number of noise nodes added and their impacts on an important graph property. Our extensive experimental results demonstrates that the noise node adding algorithms can get a better result than the previous work using edge editing only. So it is an interesting direction to study clever algorithm which can reduce the number of noise nodes if the noise nodes contribute to both anonymization and diversity. Another interesting notion is to consider how to implement this protection model in a distributed environment, in which different publishers publish their data independently and their data are overlapping.

7. REFERENCES


AUTHOR PROFILE

Siddenki Reshmi is currently pursuing M.Tech in the Department of Computer Science & Engineering, Visvesvaraya College of Engineering and Technology, M.P Patelguda, Ibrahimipatnam (M), Ranga Reddy(D), India.

Sri Dr. Bhalandra Raveendranadadh Singh working as Associate Professor & Principal in Visvesvaraya College of Engineering and Technology. He obtained M.Tech, Ph.D(CSE), is a young, decent, dynamic Renowned Educationist and Eminent Academician, has overall 20 years of teaching experience in different capacities. He is a life member of CSI, ISTE and also a member of IEEE (USA).

Ms. Sandi Sunanda working as Asst. Professor (CSE) in Visvesvaraya College of Engineering and Technology, M.P Patelguda, Ibrahimipatnam (M), Ranga Reddy(D), India.

Ms’s. Sangeetha M working as Assoc. Professor & HOD (CSE). She has completed bachelor of technology from Swamy Ramananda Theertha Institute of Science & Technology and Post-graduation from Jawaharlal Nehru Technological University, Kakinada campus and is having 12 years of teaching experience.