A Novel Privacy Protection Protocol for Hiding Sensitive Data ON Social Network

Ravi Kumar Maradana [#1], H Swapna Rekha [*2]

M.Tech Scholar [#1], Associate Professor *[2]

Department of Computer Science & Engineering,
Sri Sivani College of Engineering, Chilkapalem,
Etcherla Mandal, Srikakulam District.532402.

Abstract

Social data analysis is a style of analysis in which people work in a social, collaborative context to make sense of data. The term was introduced by Martin Wattenberg in 2005 [1] and recently also addressed as big social data analysis [2] in relation to big data computing. This project is motivated by the recognition of the need for a finer grain and more personalized privacy in data publication of social networks. We propose a privacy protection scheme that not only prevents the disclosure of identity of users but also the disclosure of selected features in users' profiles. An individual user can select which features of her profiles she wishes to conceal. The social networks are modeled as graphs in which users are nodes and features are labels. Labels are denoted either as sensitive or as non-sensitive. We treat node labels both as background knowledge an adversary may possess, and as sensitive information that has to be protected. We present privacy protection algorithms that allow for graph data to be published in a form such that an adversary who possesses information about a node's neighborhood cannot safely infer its identity and its sensitive labels. We show that our solution is effective, efficient and scalable while offering stronger privacy guarantees than those in previous research.

Keywords

Data Storage, Privacy-Preserving, Social Network Data, Sensitive Knowledge

1. Introduction

Social data analysis is a style of analysis in which people work in a social, collaborative context to make sense of data. The term was introduced by Martin Wattenberg in 2005 and recently also addressed as big social data analysis in relation to big data computing. On a Social Data Analysis system or network, users store data sets and create visual representations. The datasets and visualisations/graphs are accessible to other users of the network or website. Users can create new and interesting visualisations/graphs as well as associated commentary from the same data sets. The discussion mechanisms often use frameworks such as a blogs and wikis to drive this social exploration/Collaborative intelligence.

The publication of social network data entails a privacy threat for their users. 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. Previous research has proposed various privacy models with the corresponding protection mechanisms that prevent both inadvertent private information leakage and attacks by malicious adversaries. 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. The threat definitions and protection mechanism leverage structural properties of the graph. This paper is motivated by the recognition of the need for a finer grain and more personalized privacy. 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. We refer to these details and messages as features in the user's profile. We propose a privacy protection scheme that not only prevents the disclosure of identity of users but also the disclosure of selected features in users' profiles. An individual user can select which features of her profile she wishes to conceal.

The social networks are modeled as graphs in which users are nodes and features are labels. Labels are denoted either as sensitive or as non-sensitive. Each node in the graph represents a user, and the edge between two nodes represents the fact that the two persons are friends. Labels annotated to the nodes show the locations of users. Each letter represents a city name as a label for each node. Some individuals do not mind their residence being known by the others, but some do, for various reasons. In such case, the privacy of their labels should be protected at data release. Therefore the locations are either sensitive or non-sensitive as shown in Figure 1. 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. A more sophisticated approach consists in releasing information about sensitive labels, while ensuring that the identities of users are protected from privacy threats. We consider such threats as neighborhood attack, in which adversary friends out sensitive information based on prior knowledge of the number of neighbors of a target node and the labels of these neighbors.

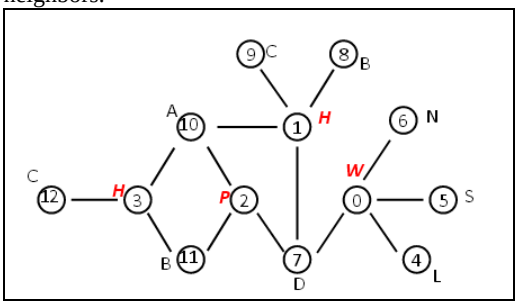

Figure.1. Example of the labeled graph representing a social network

We present privacy protection algorithms that allow for graph data to be published in a form such that an adversary cannot safely infer the identity and sensitive labels of users. We consider the case in which the adversary possesses both structural knowledge and label information.

2. Background Work

In this section we will describe the assumptions and background knowledge that is used for developing the new privacy preserving for mobile social network.

2.1 Main Motivation

The first necessary anonymization technique in both the contexts of micro- and network data consists in removing identification. This nave technique has quickly been recognized as failing to protect privacy. For microdata, Sweeney et al. propose k -anonymity **[17]** to circumvent possible identity disclosure in naively anonymized microdata. l-diversity is proposed in **[13]** in order to further prevent attribute disclosure. Similarly for network data, Backstrom et al., in **[2],** show that naive anonymization is insufficient as the structure of the released graph may reveal the identity of the individuals corresponding to the nodes. Hay et al. **[9]** emphasize this problem and quantify the risk of re-identification by adversaries with external

information that is formalized into structural queries (node refinement queries, sub graph knowledge queries). Recognizing the problem, several works **[5, 11, 18, 20,22, 24, 27, 8, 4, 6**] propose techniques that can be applied to the naive anonymized graph, further modifying the graph in order to provide certain privacy guarantee. Some works are based on graph models other than simple graph **[12, 7, 10, 3].** To our knowledge, Zhou and Pei [**25, 26**] and Yuan et al. **[23]** were the first to consider modeling social networks as labeled graphs, similarly to what we consider in this paper. To prevent re-identification attacks by adversaries with immediate neighborhood structural knowledge, Zhou and Pei **[25]** propose a method that groups nodes and anonymizes the neighborhoods of nodes in the same group by generalizing node labels and adding edges. They enforce a k-anonymity privacy constraint on the graph, each node of which is guaranteed to have the same immediate neighborhood structure with other K – 1 nodes.

3. Proposed Sensitive Label Preserving Algorithm

This section presents the detailed explanation of sensitive label of social network data and its privacy preserving during publication process.

The main objective of the algorithms that we propose is to make suitable grouping of nodes, and appropriate modification of neighbors' labels of nodes of each group to satisfy the l-sensitive-label-diversity requirement. We want to group nodes with as 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 an 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. Algorithm DNN and INN, which we devise first, sort nodes by degree and compare neighborhood information of nodes with similar degree. Details about algorithm DNN and INN please refer to **[15].**

3.1 Algorithm GINN

The algorithm starts out with group formation, during which all nodes that have not 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. Their neighbor labels are modified to be the same immediately so that nodes in one group always have the same neighbor labels.

For two nodes, v1 with neighborhood label set (LSv_1), and v2 with neighborhood label set (LSv_2), we calculate neighborhood label similarity (NLS) as follows:

$$NLS(v_1, v_2) = \frac{|LS_{v_1} \cap LS_{v_2}|}{|LS_{v_1} \cup LS_{v_2}|}$$

Larger value indicates larger similarity of the two neighborhoods. 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. Thereafter, the algorithm proceeds to create the next group. If fewer than *l*- Nodes are left after the last group's formation, these remainder nodes are clustered into existing groups according to the similarities between nodes and groups.

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, so that labels of nodes can be accordingly updated immediately for the next grouping operation. This modification process ensures that all nodes in a group have the same neighborhood information.The objective is achieved by a series of modification operations. To modify graph with as low information loss as possible, we devise three modification operations:

1) label union,
2) edge insertion and
3) noise node addition

Label union and edge insertion among nearby nodes are preferred to node addition, as they incur less alteration to the overall graph structure.

Edge insertion is to complement for both a missing label and insufficient degree value. A node is linked to an existing nearby (two-hop away) node with that label. Label union adds the missing label values by creating super-values shared among labels of nodes. 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 node is included among the values of its label super-value. After such edge insertion and label union operations, if there are nodes in a group still having different neighborhood information, noise nodes with non-sensitive labels are added into the graph so as to render the nodes in group indistinguishable in terms of their neighbors' labels which is clearly shown in algorithm 1.

Algorithm 1: Global-Similarity-based Indirect Noisy Node Algorithm

Input: graph $G(V, E, L, L^s)$, parameter l;
Result: Modified Graph G'

```
while V_left > 0 do
    if |V_left| ≥ l then
        compute pairwise node similarities;
        group G ← v_1, v_2 with Max_similarity;
        Modify neighbors of G;
        while |G| < l do
            dissimilarity(V_left, G);
            group G ← v with Max_similarity;
            Modify neighbors of G without actually adding noisy nodes ;
    else if |V_left| < l then
        for each v ∈ V_left do
            similarity(v, Gs);
            G_Max_similarity ← v;
        Modify neighbors of G_Max_similarity without actually adding noisy
        nodes;
Add expected noisy nodes;
Return G'(V', E', L');
```

We consider the unification of two nodes' neighborhood labels as an example. One node may need a noisy node to be added as its immediate neighbor since it does not have a neighbor with certain label that the other node has; such a label on the other node may not be modifiable, as its is already connected to another sensitive node, which prevents the re-modification on existing modified groups.

In this algorithm, noise node addition operation that is expected to make the nodes inside each group satisfy -sensitive-label-diversity are recorded, but not performed right away. Only after all the preliminary grouping operations are performed, the algorithm proceeds to process the expected node addition operation at the final step. Then, if two nodes are expected to have the same labels of neighbors and are within two hops (having common neighbors), only one node is added. In other words, we merge some noisy nodes with the same label, thus resulting in fewer noisy nodes.

4. Implementation Modules

Implementation is the stage where the theoretical design is automatically converted into practically by dividing this into various modules. We have implemented the current application in Java Programming language with JEE as the main interface for developing the proposed application with Front End as HTML, JSP Pages and Back end as MY SQL data base for storing and retrieving the records. Our proposed application is divided into following 3 modules. They are as follows:

- **a)** User Authentication Module
- **b)** Information Loss Module
- **c)** Sensitive Label Privacy protection Module.

a) User Authentication Module

In this module, Users are having authentication and security to access the detail which is presented in the ontology system. Before accessing or searching the details user should have the account in that otherwise they should register first.If user is registered only then he can enter into the site with his login crediantials,if not it will tell that he will fail.

b) Information Loss Module

We aim to keep information loss low. Information loss in this case contains both structure information loss and label information loss. There are some non-sensitive data's are Loss due to Privacy making so we can't send out full information to the public.

c) Sensitive Label Privacy protection Module

There are who post the image to the online social network if allow the people for showing the image it will display to his requesters it make as the sensitive to that user. Thesis is very useful to make sensitive data for the public **Batch Auditing for Multi-client Data**

5. Conclusion

In this paper, we have investigated 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 suggested a model for attaining privacy while publishing the data, in which node labels are both part of adversaries' background knowledge and sensitive information that has to be protected. We accompany our model with algorithms that transform a network graph before publication, so as to limit adversaries' confidence about sensitive label data. Our experiments on both real and synthetic data sets confirm the effectiveness, efficiency and scalability of our approach in maintaining critical graph properties while providing a comprehensible privacy guarantee.

6. References

[1]. L. A. Adamic and N. Glance. The political blogosphere and the 2004 U.S. election: divided they blog. In LinkKDD, 2005.

[2]. L. Backstrom, C. Dwork, and J. M. Kleinberg. Wherefore art thou R3579X?: anonymized social networks, hidden patterns, and structural steganography.Commun. ACM , 54(12), 2011.

[3]. S. Bhagat, G. Cormode, B. Krishnamurthy, and D. S. and. Class-based graph anonymization for social network data.PVLDB , 2(1), 2009.

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

[5]. J. Cheng, A. W.-C. Fu, and J. Liu. K - isomorphism: privacy-preserving network publication against structural attacks. In SIGMOD
, 2010.

[6]. G. Cormode, D. Srivastava, T. Yu, and Q. Zhang. Anonymizing bipartite graph data using safe groupings. PVLDB , 19(1), 2010.

[7]. S. Das,O. Egecioglu, and A. E. Abbadi. Anonymizing weighted social network graphs. In ICDE , 2010.

[8]. A. G. Francesco Bonchi and T. Tassa. Identity obfuscation in graphs through the information theoretic lens. In ICDE, 2011.

[9]. M. Hay, G. Miklau, D. Jensen, D. Towsley, and P. Weis. Resisting structural re-identification in anonymized social networks. PVLDB , 1(1), 2008.

[10]. Y. Li and H. Shen. Anonymizing graphs against weight-based attacks. In ICDM Workshops , 2010.

[11]. K. Liu and E. Terzi. Towards identity anonymization on graphs. In SIGMOD ,2008.

[12]. L. Liu, J. Wang, J. Liu, and J. Zhang. Privacy preserving in social networks against sensitive edge disclosure. In SIAM International Conference on Data Mining , 2009.

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

[14]. MPI. http://socialnetworks.mpi-sws.org/.

[15]. Y. Song, P. Karras, Q. Xiao, and S. Bressan. Sensitive label privacy protection on social network data. Technical report TRD3/12, 2012.

[16]. Y. Song, S. Nobari, X. Lu, P. Karras, and S. Bressan. On the privacy and utility of anonymized social networks. In iiWAS , pages 246{253, 2011.

[17]. L. Sweeney.K -anonymity: a model for protecting privacy. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems , 10(5), 2002.

[18]. C.-H. Tai, P. S. Yu, D.-N. Yang, and M.-S. Chen. Privacy-preserving social network publication against friendship attacks. In SIGKDD , 2011.

[19]. O. Tore, A. Filip, and S. John. Node centrality in weighted networks: generalizing degree and shortest paths. Social Networks , 32(3), 2010.

[20.] W. Wu, Y. Xiao, W. Wang, Z. He, and Z. Wang. K -symmetry model for identity anonymization in social networks. In EDBT, 2010.

[21]. X. Ying and X. Wu. Randomizing social networks: a spectrum perserving approach. In SDM , 2008.

[22]. X. Ying and X. Wu. On link privacy in randomizing social networks. In PAKDD2009.

[23]. M. Yuan, L. Chen, and P. S. Yu. Personalized privacy protection in social networks.PVLDB , 4(2), 2010.

[24]. L. Zhang and W. Zhang. Edge anonymity in social network graphs. In CSE , 2009.

[25] B. Zhou and J. Pei. Preserving privacy in social networks against neighborhood attacks. In ICDE , 2008.

7. About the Authors

Ravi Kumar Maradana is currently pursuing his 2 Years M.Tech (CSE) in Computer Science and Engineering at Sri Sivani College of Engineering, Chilkapalem, Etcherla Mandal, Srikakulam District. His area of interests includes Networks Security.

H.Swapna Rekha is currently working as an Associate Professor and Head of Department with Dept. of CSE at Sri Sivani College of Engineering, Chilkapalem, Etcherla Mandal, Srikakulam District. She has qualified in SET Examination which was conducted by AP State Board. Her research interests include network Security and Data Mining.

