Middle-East Journal of Scientific Research 24 (S2): 144-149, 2016
ISSN 1990-9233

DOI: 10.5829/idosi.mejsr.2016.24.S2.133

Enhancing Privacy in Social Networks Using Anonymization Technique During Publishing of Network Data in the Form of Graphs

U. Harita and Mr. R.J. Poovaraghan

SRM University, Ramapuram Campus,
Chennai, Tamil Nadu, India

Abstract: Social networks have been emerging trend in recent times. Social network analysis gains more importance and aims at discovering social patterns. The organization that runs a social network application needs to make public its data to a third party. Even though the truthful identifiers of individuals are removed from the published data, publication of its network information may lead to exposure of sensitive information about an individual, such as political preference of an individual. This necessitates anonymization of the network data as well. Social networks evolve over time and graphs representing the networks are required to be published frequently. The identity of the participants needs be anonymized in order to safeguard the privacy of the individuals and their relationships (edges) with other members in the social network. Privacy attacks, like the degree-trail attack re-identifies the nodes to focus on participant from a sequence of printed graphs by examination the degree of the nodes within the printed graphs with the degree evolution of a target. The power of this attack is that the adversary influences the degree of the target individual by interacting with the social network. We show that the adversary succeeds to learn about the private information even if published graph is anonymized. In this paper, we focus on *k*-anonymity since *k*-anonymity is the most applicable privacy model, which can be used even when sensitive attributes are undefined. We show that the problem is quite challenging and present a practical solution to deal with neighborhood attacks. One of our discussions is a formal method to assess the privacy risks of such attacks and empirically study the severity on real social network data. We discuss the privacy concerns and methods that can be adopted to safeguard sensitive information. We explore the privacy-preserved data publishing (PPDP) techniques and also discuss various threats that can be imposed by an adversary and present the ways to anonymize a social network.

Key words: Data mining · Privacy-preserving data publishing · Anonymization · Mutual friend attack · k-anonymity · Greedy approach

INTRODUCTION

Social networks are naturally depicted by graphs.Nodes represent the participants within the network and edges (links) represent the relationships between them. Participants of a social network typically want their sensitive information, including their relationships to the other individuals in the network, to remain private from the general public — yet data miners and researchers want to analyze the raw data to discover interesting characteristics about particular social networks. Though all of the identifying attributes of the participants were removed from the nodes, it an dversary was able to easily re-identify the node of a participant by exploiting auxiliary information such as the nodes degree [1]. A compromise is often reached between the data publishers and the data miners, where both the parties agree to some approach that is used to anonymize a snapshot of the live network data prior to its publication. Thus, data publishing in the social networks mainly deal with anonymizing graph data which is more challenging than anonymizing relational data. In this context, the following three challenges are identified: Modeling adversary's background knowledge is quite challenging. Second, measuring the loss of information in an anonymizied social network data is tougher than that in anonymizing relational data. Third, devising an appropriate anonymization technique for social network

Corresponding Author: U. Harita, M. Tech Student, Department of Computer Science & Engineering, SRM University, Ramapuram Campus, Chennai. Tamil Nadu, India.

is harder than anonymizing relational data. To deal with the above stated challenges, several approaches have been proposed. According to [2], anonymization methods on simple graphs, where vertices are un associated with attributes and edges have no labels, can be categorized as follows: edge modification, edge randomization and clustering-based generalization. In this paper, we briefly review some of the very recent studies, with focus on the attack model and privacy model and we focus on anonymizing a social network.

The remainder of this paper is organized as follows: Section II to section V discuss the problem associated with privacy in data mining.

Related Work: Recent developments in information technology necessitates collection and processing enormous amounts of personal data, such as shopping habits and medical history and driving records. Though this information is very useful in many areas, including medical research, law enforcement and national security, there is an increasing public concern regarding an individuals' privacy. Privacy is commonly seen as the right of individuals to control information about them. It has been demonstrated that existing privacy laws and policies are well behind the developments in technology and no longer offer adequate protection. In turn there emerges new privacy threats in Data Mining and knowledge discovery,.KDDM uses vast amounts of Data in order to generate hypotheses and discover general patterns in the mining process.

Privacy Risk in Graph Stream Publishing for Social Network Data: To understand how social networks evolve over time, graphs that represent the networks need to be published periodically. The identity of the participants should not be disclosed in public, hence the identity of a participant that is depicted by a node needs to be anonymized in order to safe guard the individuals privacy and their relationships (edges) with the other people in the social network. A new form of privacy attack, commonly known as the degree-trail attack is hereby focussed. This attack re-identifies the nodes belonging to a participant who is the victim from a sequence of published graphs by comparing the degree of the nodes in the published graphs with the degree evolution of a target. Using this attack the adversary can actively influence the degree of the target individual by merely interacting with the social network. Hence the adversary can succeed with a high probability even if published graphs are anonymized by strongest known privacy preserving methodologies in the literature. Moreover, the success rate does not depend on the distinctiveness of the target nodes and neither distinguishes the adversary from a normal participant. One of our contributions is a formal method to assess the privacy risk of such attacks and empirically study the severity on real social network data.

Structural Diversity for Resisting Community Identification in Published Social Networks: As the number of social networking data to be published and shared for commercial and research purposes,is increasing in rapid quantities, privacy issues about an individuals in the social networks have become a major concern. Vertex identification is one of the most rigorous problems associated with social networks has been addressed., which identifies a specific user from the network based upon the background knowledge such as vertex degree., In reality each individual in a social network is not only associated with a single vertex identity but also with community identity, which defaultly represents the personal privacy information openly to the public. In this regard a new privacy issue known as community identification is identified by showing that the community identity of a victim can be sensibly tracked even though the social network is protected by existing anonymity schemes. To deal with the above mentioned issue, *structural diversity* provides + the anonymity for various community identities. The k-Structural Diversity Anonymization (k-SDA) ensures sufficient vertices with the same vertex degree in at least k communities in a social network.. The performance studies on real data sets from various perspectives demonstrate the practical utility of the proposed privacy scheme and our anonymization approaches.

Problem Formulation: In order to preserve privacy to publish social network data, we need to first spot the privacy information to be preserved. Next we need to model the background knowledge that an adversary may use to attack the privacy. Finally we need to specify how far published social network data will be utilized so that an anonymization method retains the maximum utility as far as possible while there is no compromise in the informations privacy. Different formulations of the above issues leads to different versions of privacy preservation in social networks. Here, we propose a version which we believe has utility in many applications.

Privacy in Social Networks and Anonymization: In this paper, we are focussed in preserving the privacy of individuals which are represented as vertices in a social network. To be specific, how small subsets of vertices are connected in a social network is considered as the privacy of those vertices. Consider a social network $G = (V;E; L;L)$ and the anonymization $G0 = (V\ 0;E0;L0;L0)$ for publishing. We assume that during the anonymization, no fake vertices are added. That is, there is a bijection function A: $V\ !\ V\ 0$. This assumption is quite desirable in applications since introducing fake vertices may often change the global structure of a social network. Moreover, we assume that for $(u; v)\ 2\ E,(A(u);A(v))\ 2\ E0$.

That is, the connections between vertices in G are retained in $G0$. For a vertex $u\ 2\ V$, if an adversary can identify a vertex $u0\ 2\ V\ 0$ such that how u connects to other vertices in G is very similar to how $u0$ connects to other vertices in $G0$ and is substantially different from how any other vertices connect to others, then the privacy of u is leaked. Therefore, privacy preservation in publishing social network data is to prevent any vertex $u\ 2\ V$ (G) from being reidentified in $G0$ with high confidence. Technically, given a positive integer k, $G0$ preserves the privacy in G if every vertex $u\ 2\ V$ (G) cannot be re-identified in $G0$ with a confidence larger than $1\ k$.

Adversary Background Knowledge: In order to attack the privacy of a target individual in the original network, an adversary requires some background knowledge. When equipped with different background knowledge, an adversary conducts different types of attacks against privacy. Therefore, the assumptions of adversaries' background knowledge play a major role in modeling privacy attacks on social networks and developing anonymization strategies to protect privacy in social network data. In this paper, we assume that an adversary may have the background knowledge about the neighborhood of some target individuals. This assumption is realistic in many applications. Among many types of information about a target victim that an adversary may collect to attack the victim's privacy, one essential piece of information easy to be collected is the neighborhood, i.e., who the neighbors of the victim are and how the neighbors are connected.. Generally, we will think about the d-neighbors of the target vertex, i.e., the vertices inside distance d to the target vertex within thenetwork wherever d could be a positive whole number. However, once d is giant, aggregation data concerning the d-neighbors of a target vertex might usually be impractical for associate opposer since the opposer might usually have a restricted access to an oversized social network. Moreover, as found in several social networks, the network diameter is usually tiny.In alternative words,once d>1, associate opposer might needto collect data concerning several vertices. Therefore, we tend to confine our discussionduring this paper to the essential case wherever solely the immediate neighbors, i.e., vertices in NeighborG(u), square measurethought-about. The case of d > one is fascinating for future work.An opposer might attack the privacy exploitation the neighborhoods.For a social network G, suppose associate opposer is aware of NeighborG(u) for a vertex u a pair of V (G). If NeighborG(u) has k instances in G0 wherever G0 is associate anonymization of G, then u are often re-identified in G0 confidently 1k.Similar to the philosophy within the k-anonymity model [3], to shield the privacy of vertices sufficiently, we would like to stay the re-identification confidence under a threshold. Let k be a positive whole number. For a vertex u a pair of V (G), u is k-anonymous in anonymization G0 if there square measure a minimum of (k ¡ 1) alternative vertices v1;::: ; vk¡1 a pair of V (G) such NeighborG0 (A(u)),NeighborG0 (A(v1)),:::, NeighborG0 (A(vk¡1)) square measure similarity.G0 is k-anonymous if each vertex in G is k- anonymous in $G0$. Analogous to the correctness of k-anonymity model on relational data, we have the following claim.*Theorem 1 (K-anonymity):* Let G be a social network and $G0$ an anonymization of G. If $G0$ is k-anonymous, then with the neighborhood background knowledge, any vertex in G cannot be re-identified in $G0$ with confidence larger than $1k$.An adversary knowing the neighborhood of a target vertex is a strong assumption. Provided privacy is preserved under this assumption, privacy is also preserved when an adversary knows only part of the neighborhood (i.e., only some neighbors and some connections among neighbors) of a target vertex.

Usage of Anonymized Social Networks: An important characteristic of anonymizing social network data is how the anonymized networks are expected to be utilized. Different applications may have different outlooks. In this paper, we focus on using anonymized social networks to answer aggregate network queries. An aggregate network query computes the aggregate on some paths or subgraphs satisfying some given conditions.Aggregate network queries are useful in many applications, such as customer relationship management. While many types of queries on social networks are interesting, we are particularly interested in aggregate network queries in this paper since typically detail data is needed to answer such

queries accurately. Using aggregate network queries we can inspect the effectiveness of social network anonymization in a meaningful way.

Proposed Solution: Privacy becomes a crucial concern in many applications. The development of techniques that incorporate privacy concerns has become a fruitful direction for database and data mining research. One of the major issue is publishing micro data publicly [4]. In this paper, we focus on k-anonymity since *k*-anonymity is the most essential and widely applicable privacy model, which can be used even when sensitive attributes are not well defined. In this section, we introduce a practical approach to anonymize a social network and that satisfies the k-anonymity requirement. The method is in two steps. First, we extract the neighborhoods of all vertices in the network. That helps to facilitate the comparisons among neighborhoods of different vertices including the isomorphism tests which is conducted frequently in anonymization.

In the second step, we greedily organize vertices into groups and anonymize the neighborhoods of vertices in the same group. Due to the well-recognized power law distribution of the degrees of vertices in large social networks, we start with those vertices of high degrees. One major challenge in anonymizing a social network is that changing labels of available vertices and adding edges may affect the neighborhoods of other vertices and the network properties. The following properties help us in designing anonymization methods. Property 1: vertex degree in power law distribution. The degrees of vertices in a large social network frequently follow the power law distribution. Such degree distributions have been applied in various social networks including Internet and biological networks. Property 2: the "small-world phenomenon" [5]. Also stated as "six degrees of separation", which states that large social networks in practice often have small average diameters.

Our social network anonymization method processes vertices in the degree of descending order, and utilizes the above two properties of large social networks in practice. The *k*-anonymity requires that each vertex *u 2 V* (*G*) is grouped with at least (*k ¡* 1) other vertices such that their anonymized neighborhoods are isomorphic. For a group *S* of vertices having the isomorphic anonymized neighborhoods, all vertices in *S* have the same degree. Since the degrees of vertices in a large social network follow a power law distribution, only a few vertices have a high degree. Processing those vertices of high degrees first can keep the information loss about those vertices low. There are often many vertices of a low degree. It is relatively easy to anonymize those low degree vertices and retain high quality. Moreover, as will be shown soon, low degree vertices can be used to anonymize those high degree vertices and do not affect the diameters of the network too much.

Anonymization Quality Measure: In our social network anonymization model, there are two methods to anonymize the neighborhoods of vertices: generalizing vertex labels and adding edges. Each of these methods are prone to some information loss. The information loss due to generalization of vertex labels is measured by the normalized certainty penalty [6]. To illustrate the same,consider a vertex *u* of label *l*1, where *l*1 is at the leaf level of the label hierarchy, i.e., without any descendants. If *l*1 is generalized to *l*2 for *u* where *l*2 *A l*1. Let *size*(*l*2) be the number of descendants of *l*2 that are leafs in the label hierarchy and *size*(*) be the total number of leafs in the label hierarchy. Then, the *normalized certainty penalty* of *l*2 is *NCP*(*l*2) = *size*(*l*2)/*size*(*).The information loss due to addition of edges is be measured by the total number of edges added and the number of vertices that aren't in the neighborhood of the target vertex and are linked to the anonymized neighborhood for the purpose of anonymization. Consider two vertices *u*1*; u*2 *2 V* (*G*) where *G* is a social network. If *NeighborG*(*u*1) and *NeighborG*(*u*2) are generalized to *NeighborG0* (*A*(*u*1)) and *NeighborG0* (*A*(*u*2)) such that *NeighborG0* (*A*(*u*1)) and *NeighborG0* (*A*(*u*2)) are isomorphic. Let *H* = *NeighborG*(*u*1) ⋃ *NeighborG*(*u*2) and *H0* = *NeighborG0* (*A*(*u*1)) ⋃ *NeighborG0* (*A*(*u*2)). The *anonymization cost* is

$$Cost(u,v) = \alpha \cdot \sum_{v' \in H'} NCP(v') + \beta \cdot |\{(v_1,v_2)|(v_1,v_2) \notin E(H), (\mathcal{A}(v_1),\mathcal{A}(v_2)) \in E(H')\}| + \gamma \cdot (|V(H')| - |V(H)|)$$

where α, β and υ are the weights specified by users., The cost consists of three parts. first part is the normalized certainty penalty that measures the information loss of generalizing labels of vertices. The second part measures the information loss due to addition of edges. The last part counts the number of vertices linked to the anonymized neighborhoods in order to achieve *k*-anonymity. The anonymization cost of two vertices *u* and *v* measures the similarity between *NeighborG*(*u*) and *NeighborG*(*v*). The smaller the anonymization cost, the similar are the two neighborhoods. Another approach is the greedy method

to anonymize two neighborhoods *NeighborG*(*u*) and *NeighborG*(*v*).We first find all possible perfect matches of neighborhood components in *NeighborG*(*u*) and *NeighborG*(*v*). Two components are said to perfectly match with each other if they have identical minimum DFS code. Those perfect matches are titled as "matched" and pass over for further consideration. For example, consider two vertices *u* and *v* whose neighborhoods are shown in Figure 2. Each vertex is shown in the form of (*id; label*). The neighborhood component *C*2(*u*) *2NeighborG*(*u*) perfectly matches *C*3(*v*) *2 NeighborG*(*v*).For those unmatched components, the anonymization algorithm tries to pair similar components and anonymize them.. To calculate the similarity between two components, we try to match vertices that are similar in the two components as far as possible.The above mentioned similarity search problem.has been proved NP-hard [7]. Instead of calculating the optimal matching, we conduct a greedy match in which we first try to find two vertices with the same degree and the same label in the two components to be matched. If there are recurring matching vertex pairs, the pair that has the highest vertex degree is chosen.

If there is no such a pair of matching vertices, the matching requirement is relaxed (vertex degree and label),we now calculate the difference of degrees and the normalized certainty penalty of generalizing the labels in the label hierarchy and select the one with the minimum anonymization cost. Now a breadth-first search is conducted to match vertices one by one, until all possible vertex matchings are done. The anonymization cost is computed according to the matching and is used to measure the similarity of the two components.

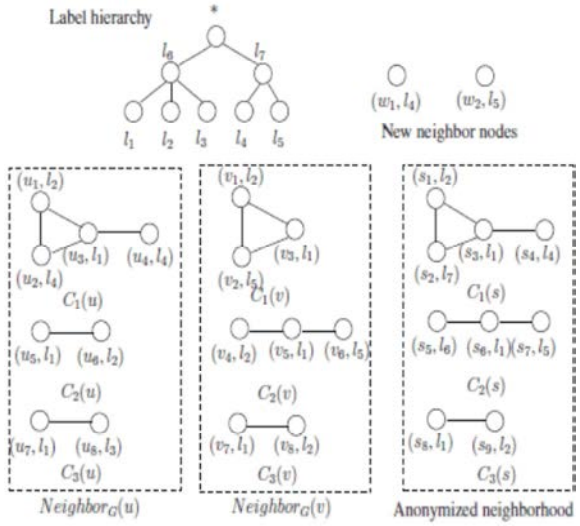

Fig. 2:

Consider components *C*1(*u*) and *C*1(*v*) in Figure 2 Vertices *u*1 and *v*1 match. We start from these two vertices and perfom a breadth-first search. Vertex *v*2 partially matches vertex *u*2. Vertex *v*3 partially matches vertex *u*3. The vertex matching stops since all possible vertex matchings are found. However, vertex *u*4 does not find any vertex matching in *C*1(*v*). Thus we have to find a vertex *w*1 *2 V* (*G*) that is neither in *C*1(*v*) nor in *C*1(*u*) and add it into *C*1(*v*), so that *C*1(*u*) and *C*1(*v*) can be anonymized to the same. When a vertex has to be introduced into the neighborhood for the sake of anonymization, the following rules are used: we first consider those vertices in *V* (*G*) that are unanonymized. The vertex with smallest degree has the highest priority. If there are more than one candidate with the same smallest degree, we choose the one having the closest label in terms of normalized certainty penalty. If we cannot find any other vertex that is unanonymized, we select one anonymized vertex *w* with the smallest degree and satisfying the label requirement and mark *w* and its (*k ¡* 1) other vertices anonymized in the same group as "unanonymized". In our example, suppose we can find an unanonymized vertex (*w*1*; l*4) to be added to *C*1(*u*), the anonymization cost of *C*1(*u*) and *C*1(*v*) is *α*.Based on the component similarity, we can pair similarcomponents. We start with the component with the largest number of vertices. This component is paired with the most similar component in the other neighborhood. The two paired components are anonymized to the same, marked "matched",and removed from consideration. The matching continues until all components in one neighborhood are marked "matched".If there are some components left in the other neighborhood say *NeighborG*(*u*), we use some other vertices in *V* (*G*) that are not in *NeighborG*(*u*) to construct a component and add it to *NeighborG*(*u*) to construct the matching and anonymization. The vertices are selected using the same criteria as selecting vertices to match two components.We anonymize each pair of matched neighborhood components to the same. The two neighborhoods then are anonymized. For example, in Figure 4, the algorithm matches components *C*1(*u*) and *C*1(*v*) and *C*2(*v*) and *C*3(*u*) in turn.As a result, two vertices *w*1 and *w*2 from *V* (*G*) have to be added into components *C*1(*v*) and *C*3(*u*), respectively.

CONCLUSION

In this paper, we have tackled an important issue of preserving privacy in social network data and took an initiative to combat neighborhood attacks. We modeled

the problem in a systematic approach and developed a practically feasible approach. As social network data is much more complex than relational data, privacy preserving in social networks is much more difficult and gives rise to many serious efforts in the future. To be specific, modeling attacks by the adversary and developing privacy preservation strategies are critical. For future work, we believe that the following types of attacks should be addressed systematically. We have only handled 1-neighborhoods in this paper. It could be desirable in certain applications that *d*-neighborhoods ($d > 1$) are safe guarded though this may introduce a serious challenge during computation.As d increases the neighborhood size increases exponentially. The anonymization of large neighborhoods are become too challenging. A *k*-anonymous social network still may face the threat of leakage of privacy. If an adversary can still recognize a victim in a group of anonymized vertices in a group, but all are associated with some sensitive information, then the adversary still can know that sensitive attribute of the victim. Some mechanism analogous to *l*-diversity [8] should be introduced.

REFERENCES

1. Brankovic, L. and V. Estivill-Castro, 1999. Privacy issues in knowledge discovery and data mining, in Proceeding of Australian Institute Computer Ethics Conferences, pp: 88-99.
2. Medforth, N. and K. Wang, 2011. Privacy risk in graph stream publishing for Social network data, in Proceeding IEEE 11th International Conference Data Mining (ICDM), pp: 437-446.
3. Tai, C.H., P.S. Yu, D.N. Yang and M.S. Chen, 2013. Structural Diversity for resisting community identification in published social Networks, IEEE Transaction Knowledge of Data Enggineering, 26(1): 235-252.
4. Wernke, M., P. Skvortsov, F. Dürr and K. Rothermel, A classification of location privacy attacks and approaches, Person Ubiquitous Computing, 18(1): 163-175.
5. Narayanan, A. and V. Shmatikov, 2008. Robust de-anonymization of large sparse datasets, in Proceeding of IEEE Symposium. Secure Privacy (SP), pp: 111-125.
6. Adamic, L. and E. Adar, 2005. How to search a social network, Social Networks, 27(3): 187-203.
7. Kossinets, G. and D.J. Watts, 2006. Empirical analysis of an evolving social network, Science, 311(5757): 88-90.
8. Backstrom, L. *et al.*, 0000. Group formation in large social networks: membership,growth and evolution, in *KDD'06*.