DOI: 10.3969/j. issn. 1673-5188. 2016. S0. 001
http://www.cnki.net/kcms/detail/34.1294.TN.20160615.1654.002.html, published online June 15, 2016

Attacks and Countermeasures in Social Network Data Publishing

YANG Mengmeng, ZHU Tianqing, ZHOU Wanlei, and XIANG Yang
(School of Information Technology, Deakin University, Burwood, VIC 3125, Australia)

Abstract

With the increasing prevalence of social networks, more and more social network data are published for many applications, such as social network analysis and data mining. However, this brings privacy problems. For example, adversaries can get sensitive information of some individuals easily with little background knowledge. How to publish social network data for analysis purpose while preserving the privacy of individuals has raised many concerns. Many algorithms have been proposed to address this issue. In this paper, we discuss this privacy problem from two aspects: attack models and countermeasures. We analyse privacy concerns, model the background knowledge that adversary may utilize and review the recently developed attack models. We then survey the state-of-the-art privacy preserving methods in two categories: anonymization methods and differential privacy methods. We also provide research directions in this area.

Keywords

social network; data publishing; attack model; privacy preserving

1 Introduction

Social network is a very popular platform where people make new friends and share their interests. A dramatically increasing number of users have joined social networks. Social network service providers hold a large amount of data. To some extent, these data provide a great opportunity to analyse social networks, while at the same time, it brings privacy concern.

Normally, in order to preserving users' privacy, social network data are published without identity information, which is replaced by meaningless numbers or letters. However, Backstrom et al. [1] pointed out that simply removing the identities of vertices could not preserve the privacy. Users can still be identified by attackers based on various background knowledge.

Many privacy preserving methods were proposed to defend against these attacks. Unlike the case in traditional relational datasets, privacy preserving in social network data publishing is a challenging problem:

- All the identities in the social network are connected with each other by edges, so any small modification may cause big changes to other vertices, sub-graph and even the whole network.
- It is very difficult to modify background knowledge because there are so much information can be used as background knowledge to re-identify the identities and breach the privacy.
- It is difficult to quantify information loss and utility. There are many elements in social networks, such as hubs, betweenness and communities, so we cannot simply compare two networks by vertices and edges. Additionally, utility is different based on different applications. We cannot use a unified standard to measure the utility.

Utility and privacy are contradicting elements. Most privacy preserving methods acquire a high level of privacy guarantee at the expense of utility. How to balance utility and privacy is a key problem when designing a privacy-preserving algorithm.

Our contributions in this paper are summarised as follows:

- We model the background knowledge that can be used by adversaries to break users' privacy.
- We classify the possible attack methods into two categories. One is that the adversary attempts to re-identify a specific person, and the other is that the adversary attempts to identify as many individuals as possible.
- We categorise the anonymization methods into two groups: anonymization and differential privacy. We review the privacy preserving models developed in recent 5 years.

The rest of this paper is organised as follows. We summarise the attack models in section 2. In section 3, we review the state-of-the-art privacy preserving methods from two categories: anonymization and differential privacy. Then we conclude the pa-

per and give the research direction in the future in section 4.

2 Attack Models

With the development of social network analysis and mining, privacy becomes an urgent problem that needs to be solved. While simply moving identifier is far from preserving the information, any background knowledge can be used by the adversary to attack the privacy easily.

2.1 Background Knowledge Attacker Utilizes

Background knowledge is the information that is known to adversaries, which can be used to infer privacy information of an individual in the social network. It plays an important role in modeling privacy attacks and developing countermeasures. We explore possible background knowledge that can be used by the adversary.

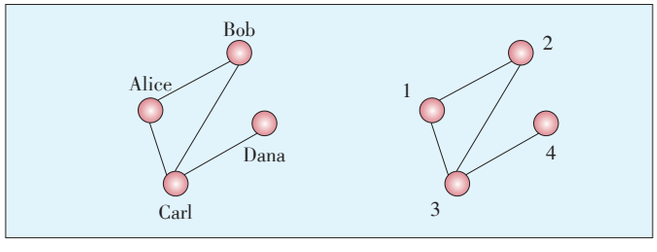
1) Vertices degree

Vertices degree represents how many direct connections between a node and its neighbours. Once the degree of the user is different from others in the graph, the vertex is re-identified. For example, in **Fig. 1**, node 3 and node 4 can be identified directly if the adversary knows Carl has three friends and Danna has only one friend.

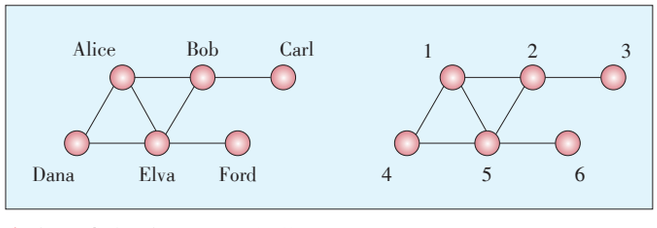
Tai et al. [2] identified a new attack called friendship attack, which is based on degree pair of an edge. They launched both degree and friendship attacks on 20Top-Conf dataset and proved that the friendship attack causes a much more privacy disclosure than the degree attack.

2) Neighbourhood

Neighbourhood refers to the neighbours of an individual who have connections with each other. Attackers make use of this kind of structural information to identify individuals [3], [4]. For example, in **Fig. 2**, if attackers know Bob has three friends and two neighbors and they connected with each other, Bob can be recognized in the anonymized graph.



▲**Figure 1. A degree attack.**



▲**Figure 2. A neighbourhood attack.**

Ninggal et al. [5] proposed another kind of attack called neighbourhood-pair attack, which uses a pair of neighbourhood structural information as background knowledge to identify victims. Such attacks assume attackers know more information than neighbourhood attacks do, so attackers have a higher chance to distinguish users in a dataset.

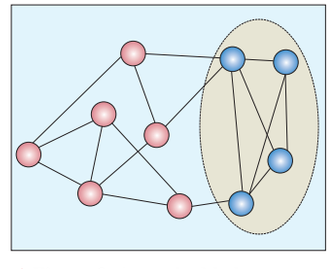
3) Embedded sub-graph

Sub-graph refers to a subset of the whole graph. Some adversaries create few fake nodes and build links using a specific way before the data is published, and then match the target graph with reference graph based on the sub-graph which has been planted. In **Fig. 3**, the grey part is the original graph, the black part is the sub-graph embedded by the adversary. Normally, the embedded sub-graph is unique and easy for attackers to identify after the dataset is released.

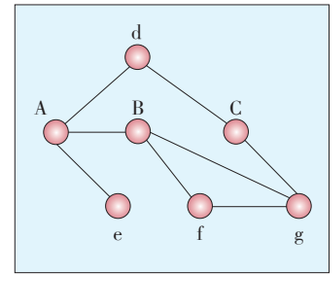
4) Link relationship

The relationship between two vertices also can be acquired by an adversary. Wang et al. [6] considered that the public users' identities are public and not sensitive. They utilized the connection between victims and public users to perform attack. For example, in **Fig. 4**, A, B, and C are public users, such as BBC and Michael Jackson. Their identities are publicity, and if attackers know vertex d has one hop to A and C and two hops to B, d can be identified.

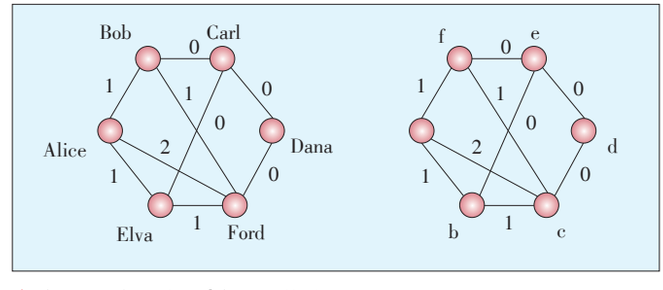
Sun et al. [7] committed a mutual friend attack. Their algorithm identifies a pair of users who connect to each other based on the number of mutual friends. For example, in **Fig. 5**, the numbers on the edge represent the number of mutual friends between two nodes. If the adversary knows Alice and Ford have two mutual friends, then she/he can identify a and c combined with other reference information (e.g. degree).



▲**Figure 3. An embedded sub-graph attack.**



▲**Figure 4. A fingerprint attack.**



▲**Figure 5. A mutual friends attack.**

5) Attributes of vertices

Attributes of individuals in a social network are represented as labels to vertices. Attackers may get the attributes of victims, such as age, sex and occupation. Such information can be helpful for adversaries to compromise users' privacy. For example, in Fig. 5, if the attacker knows Alice is a girl, Ford is a boy, he can identify them specifically by checking labeled sex information.

2.2 Attack Methods

There are two types of attacks. The first one is that the adversary tries to identify a specific person in the released dataset [5]–[7]. The other one is that the adversary attempts to identify as many individuals as possible. Most studies focus on the second one.

2.2.1 Structural Based Attacks

The first structural attack for re-identifying the vertices mutually present in two real and large social networks was proposed by Narayanan et al. [8]. They seeded 4-cliques and identified them by degrees and common neighbour counts. The propagation phase is based on the similarity score between the identified node and candidate node who has at least one mapped neighbour with the identified node. Their algorithm correctly identified 30.8% vertices just with 12.1% error. In their later work, Narayannan et al. [9] improved the seed identification process and formulated it as a combinatorial optimization problem, which improves the robustness compared with previous works.

Competing with Narayanan et al. [8], Peng et al. [10] proposed a two-stage de-anonymization algorithm. The attacker first plants a small specially designed sub-graph Gf, then propagate it based on two dissimilarity metrics. The experiment results showed their algorithm had a better efficiency even when the seed set was very small than Narayanan's algorithm. Besides, the incorrect identification number grows slowly when the initial seeds are growing. However the algorithm proposed by Peng et al. performed not well for large perturbation percentage. Besides, the test dataset used was too small with just hundreds of vertices.

Simon et al. [11] presented a Grasshopper algorithm. They selected the top degree nodes as seeds and introduced a weighting scheme based on the number of mappings in the nodes' neighbourhood and set convergence criteria. Their algorithm achieved a better result compared to [8] when the attacker has a rather noisy knowledge (lower similarity between two graphs), but does not always have a good result on different datasets.

Ji et al. [12] defined two vectors in terms of many structural attributes. They used these vectors to map nodes between two networks. Only 5–10 seed nodes are enough to de-anonymize. However the complexity and computational cost of this algorithm are very high.

There is another research group [13]–[16] maping the vertices between two similar graphs based on the graph matching algorithm. For example, Yartseva [13] introduced a simple percolation-based graph matching algorithm. This algorithm starts from a pre-matched seed set, and then maps the nodes with at least r neighbours that have been matched. In order to improve the precision, Korula et al. [15] matches the vertices from high degree to low degree according to the number of similarity witnesses. Chiasserini et al. [16] extended Yartseva's work [13] to a more realistic case that considering the power-low degree distribution. It makes the seed set as small as n.

2.2.2 Other Attack Methods

Most attacks are based on structure of the graph. However, there are other methods used to disclose users' privacy. Faresi et al. [17] and Goga et al. [18] used labeled attributes to correlate identical accounts on different social sites. Nilizadeh et al. [19] proposed a community-based de-anonymize method. They partitioned the graph into many communities. The algorithm identifies the seed communities first, and then maps the communities by creating a network of communities. Sharad et al. [20] proposed an automated de-anonymization method, which formulates the de-anonymizaiton problem as a learning task. The related research [21]–[27] focuses on predicting the link relationship, which can be used to disclose users' privacy as well.

3 Countermeasures

It is widely recognized that simply moving users' identity cannot guarantee their privacy. Many researchers pay much attention to this problem. We categorise the state-of-art privacy preserving methods into two categories: anonymization and differential privacy. **Table 1** shows the privacy models corresponding to attack models.

3.1 Anonymization

Anoymization is a popular method for preserving privacy, which is extensively used for social network data publishing.

▼Table 1. Privacy models

Privacy model	Attack model				
	Degree	Friendship	Neighbourhood	Sub-graph	Mutual friends
k-degree [29]	√				
structural diversity [34]	√				
k^2-degree-anonymity [2]		√			
k-neighbor [3]			√		
k-NMF-anonymity [7]					√
k-isomorphism [68]				√	
k-automorphism [69]	√		√	√	
differential Privacy [52]	√	√	√	√	√

We review these privacy preserving methods in this section. **Table 2** summarises recently developed anonymization methods from three aspects of privacy breach.

3.1.1 Preserving Vertices Identity

Most studies in recent years focus on preserving users' identity, preventing adversary re-identifying vertices in the graph. The main anonymization methods are based on k-anonymity [28], which means there are at least k nodes have the same structure with each other in the graph. It is realized by changing the structure of the original graph.

1) Graph Modification

Graph modification is a way that makes the graph satisfy k-anonymity by adding or deleting edges or nodes. State-of-art graph modification methods are summarised as follows.

In 2008, Liu and Terzi [29] first answered the question "how to minimally modify the graph to protect the identity of each individual involved?" They studied the vertices re-identity problem based on the degree background knowledge and provided a dynamic-programming algorithm for generating the k-anonymous graph based on the desired degree sequence.

Liu and Li [30] pointed out that the algorithm proposed in [29] had uncertainties. For example, if the anonymous graph construction process is random, the results will be totally different from the original graph. They developed two degree sequence partition algorithms. Those algorithms partition the degree sequence according to the partition cost calculated by the sum of difference of max neighbor vertices to their target degree. The nodes with smallest degree and distance are considered for constructing the graph satisfying k-degree anonymization. Noisy nodes are added when adding edges only cannot satisfy the constraint.

In order to guarantee the utility of anonymized graph, Wang et al. [31], [32] defined a measure standard Hierarchical Community Entropy (HCE) based on the hierarchical random graph (HRG) model to represent the information embedded in the graph community structure. They proposed an algorithm modifying edges that change the original graph to the nearest k-anonymizaiton graph.

Ninggal and Abawajy [33] introduced a utility-awared graph anonymizaiton algorithm. They use two metrics, Shortest Path Length (SPL) and Neighbourhood Overlap (NO) to quantify the utility. Compared to the scheme in [29], the algorithm in [33] introduces less distortion and improves utility preservation. However, this algorithm was only tested on four small datasets with hundreds of vertices. Besides, the computational cost of the algorithm is expensive.

Tai et al. [34] introduced a new privacy problem for protecting the community identity of vertices in the social network against degree attack. Even the graph satisfies k-degree anonymity, community identity still can be breached if the sets of nodes with the same degree belong to the same community. The authors proposed a structural diversity model by adding edges to make sure that the nodes with the same degree are distributed to at least k communities.

Tai et al. [2] introduced a Degree Sequence Anonymization algorithm to defend the friendship attack. The algorithm clusters vertices with similar degree, constructs at least k edges between two clusters by adding and deleting, and then adjusts the edges to k-anonimization under some conditions.

Zhou and Pei [3] provided a practical solution to defend neighborhood attack. They proposed a coding technique based on the depth-first search tree to represent neighborhood components. The algorithm tries to modify similar vertices as much as possible by adding edges to the vertices with the smallest degree, making sure every neighborhood sub-graph is isomorphic to at least k-1 other sub-graphs. But it does not consider the graph metric that may destroy the utility of the graph. In order to solve this problem, Okada et al. [35] extended the node selection function. They selected the closest node with the smallest degree and most similar label to suppress the changes

▼**Table 2. Characters of anonymization algorithms**

Anonymization algorithms	Operation	Information loss (anonymization cost)	Usability evaluation	Privacy disclosure		
				Vertices	attributes	links
[29]	EA, ESW	BD	GGP	√		
[30]	EA, VA	BD	GGP	√		
[31]	EA, ED, ESH	HCE	GGP	√		
[33]	EA, ED	UPM	GGP	√		
[34]	EA, VSP	BD	GGP	√		
[2]	EA, ED	BD	GGP	√		
[3]	EA, LG	BD	ANQ	√		
[35]	EA, LG	BD	GGP	√		
[7]	EA, ED	BD	GGP	√		
[36]	LG, EA, NA	BD	GGP, ANQ	√		
[4]	EA	BD	GGP	√	√	
[44]	EA, ED, VA	BD	GGP	√	√	
[45]	EA, VA	BD	GGP	√	√	
[46]	LG	BD	GGP	√	√	
[48]	REA	BD	GGP			√
[49]	ESH	–	LRP, RE, GGP			√
[50]	ESH	–	GGP			√
[40]	CL	SIL&DIL [51]	GGP	√		√
[38]	CL	SIL	SIL	√		√

ANQ: aggregate network queries
BD: based on distance
CL: clustering
DIL: descriptive information loss
EA: edge addition
ED: edge deletion
ESH: edge shifting
ESW: edge swapping
GGP: general graph properties
HCE: the change of Hierarchical Community Entropy value
LG: label generalization
LRP: link retention probability
PESW: possibility edge swapping
RE: reconstruction error
REA: random edge addition
SIL: structural information loss
UPM: The Utility Preserving Magnitude based on shortest path difference metric and neighborhood-overlap metric
VA: vertices addition
VD: vertices deletion
VSP: vertices splitting

of the distance of nodes. If the distance exceeds the threshold, the algorithm adds a noise node to suppress the changes of distance.

Wang et al. [36] considered the situation that attackers explore the sensitive information with labeled neighbourhood information as background knowledge. Their algorithm groups closest nodes according to the label sequence similarity. With different labels, each group contains at least one node. The authors modified the graphs in each group by label generalization, edge insertion and node insertion to make them isomorphic.

Sun et al. [7] proposed k-NMF anonymity to defend mutual friends attacks. This algorithm ensures that there exist at least k-1 other friend pairs in the graph that share the same number of mutual friends. The algorithm puts the edges into several groups and anonymizes each edge in the group one by one by adding edge. The algorithm chooses the candidate node that has maximum mutual friends with the vertex that need to be anonymized to ensure the utility of the graph, because the more mutual friends between the two vertices, the less impact the edge addition will have on the utility of the graph.

2) Clustering Methods

Clustering-based methods group the closest nodes together and show the original graph with super vertices and super edges. It shrinks the graph considerably, so it is not suitable for analysing local structure [37]. However it is a good method for answering aggressive queries.

Sihag [38] used clustering methods to make the vertices satisfy k-anonymization. They modeled the clustering process as an optimization problem and applied the genetic algorithm for choosing the best solution. The structural information loss proposed in [39] is used as the fitness function. A better solution is generated in each iteration until the terminating condition is satisfied.

Tassa and Cohen [40] introduced a sequential clustering algorithm to anonymise social network data. All nodes are clustered randomly to N/K groups (N is the vertices number, and K represents the cluster size). If Co is the cluster that node v belongs to, the information loss is calculated when moving v from Co to other clusters Ct. Node v is moved to the cluster that fits it best. This process is repeated until no vertices need to be moved to another cluster. This algorithm performs better in terms of reducing information loss and maintaining graph attributes than other clustering algorithms in [39] and [41]. In addition, the authors first applied the privacy preserving algorithm to a distributed social network.

3.1.2 Preserving Sensitive Attributes

The main method for preserving attributes in the social network is l-diversity [42], [43]. As an extension of the k-anonymity model, the l-diversity model reduces the granularity of data representation by using such techniques as generalization and suppression.

Paper [4], [44]–[46] all used l-diversity to protect the sensitive labels. Motivated by the observation that the nodes with high degree are usually famous people who have a relatively low privacy requirement, Jiao et al. [45] classified the nodes into three categories: High privacy requirement, middle and low, provided a personalized k-degree-l-diversity (PKDLD) model to protect nodes' degree and sensitive labels. Chen et al. [46] protected sensitive attributes in a weighted social network using l-diversity technology.

Rajaei et al. [47] provided ($\alpha, \beta, \gamma, \delta$) Social Network Privacy (SNP) to protect directed social network data with attributes against four types of privacy disclosure: presence, sensitive attribute, degree and relationship privacy. They grouped nodes with a high number of different properties and partition attributes to few tables and connected them by group IDs. This algorithm answers aggregate queries with high accuracy and maintains more data utility because the exact value is published and degree distribution is not changed. However, some false individuals would be generated during the process of anonymization, which may cause some errors.

3.1.3 Preserving Link

The basic technology for preserving link privacy is random perturbation. The main strategy is edge addition, deletion and switch.

Mittal et al. [48] proposed an algorithm preserving link privacy based on random walk. Their algorithm introduces fake edges with specific probability and defines a parameter t to control the noise that they want to add to the original graph.

Fard et al. [49] proposed a sub-graph-wise perturbation algorithm to limit link disclosure. They modeled the social network as a directed graph and partitioned vertices into some sub-graphs according to the closeness of nodes. The destination nodes are replaced by the nodes randomly selected from all destination nodes in the sub-graph with a certain probability. The algorithm preserves more graph structures compared with selecting from the whole graph. However, with the increasing of the number of sub-graphs, each sub-graph becomes very small, which increases the threats of identifying the link. In order to solve this drawback, neighbourhood randomization [50] was proposed. Selecting the destination nodes from the neighbourhood of the source node can avoid partition graph.

Ying and Wu [51] theoretically analysed how well the edge randomization approach protected the privacy of sensitive links, while Tassa and Cohen [40] believed that it is elusive and high non-uniform. Ying and Wu pointed out that some hub nodes with high degree are still distinguishable even when the algorithm has a high perturbation parameter. In addition, the random perturbation fails to provide a lower anonymization level (when k is small).

3.2 Differential Privacy

The main methods for protecting users' privacy are to modi-

fy the graph structure. Generally, these methods can only defend one specific kind of attacks and have no ability to resist the newly developed approaches. However, differential privacy [52] has been proved performing well in this direction.

Differential privacy is a mechanism that makes little difference to the results of the query with the addition or deletion of any tuple by adding random noise on the output. It works well on the tabular dataset preserving privacy. Some researchers also apply it to social networks [53]–[62], because it does not need to model background knowledge that is still a challenge for traditional anonymization methods. Besides, differential privacy is based on mathematics, which provides a quantitative assessment method andmakes the level of privacy protection comparable. We introduce it from two sides: node privacy and edge privacy.

3.2.1 Edge Privacy

Edge privacy makes negligible difference to the result of the query by adding or deleting a single edge between two individuals in the graph. The privacy dK-graph model [63] was used to enforce edge differential privacy [56]–[58]. The dK-series is used as the query function, but controllable noise is added based on the sensitivity parameter. In order to reduce the noise added to the dK-series, Sala et al. [57] provided a Divide randomize and Coonquer (DRC) algorithm, partitioning the data of dK-series into clusters with similar degree. It significantly reduces the sensitivity for each sub-series.

Wang and Wu [58] pointed out that Sala's approach was based on local sensitivity that may reveal information of the dataset (the example in [64]). Therefore, this approach could not achieve rigorous differential privacy. The authors in [58] used smooth sensitivity to calibrate the noise and achieved a strict differential privacy guarantee with smaller noise.

Xiao et al. [59] provided a novel sanitization solution that hides users' connection to others through differential privacy. They used the hierarchical random graph model (HRG) to infer the social network structure and record connection probabilities between all pair of vertices in the graph. In order to reduce the sensitivity, the Markov Chain Monte Carlo (MCMC) method is designed to sample a good HRG from the whole space. The sanitized graph is generated based on the identified HRG. This algorithm achieves a desirable utility due to smaller sensitivity compared with state-of-the-art works and effectively preserves some structural properties.

Edge privacy is a weaker guarantee than node privacy. Adversaries may still learn some general information. For example, high-degree nodes may have an identifiable effect on the query results [65]. However, it is practically strong enough in many applications, such as answering queries about individual relationship.

3.2.2 Node Privacy

Node privacy means adversaries do not have the ability to learn any information of an individual. It is very difficult to achieve node privacy while to guarantee the accurate query result, because the sensitivity is a very big result from adding or deleting nodes and connected edges. The query results would be too noisy to be applied in real life [66], [67], but it was proved a strong guarantee in some cases [65]. Some studies [54], [55] contributed to reduce sensitivity and returned accurate answers. However, existing algorithms cannot provide a good utility for practical applications. It is still an open problem.

4 Conclusions and Future Direction

In this paper, we first summarised and analysed the adversaries' attack methods to provide a good reference for researchers to design privacy preserving algorithms. Then we surveyed recently developed privacy preserving methods in two categories, anonymization and differential privacy. Though the privacy preserving methods are developed very well in the relational dataset, it is still in its infancy in social network datasets. For traditional method, there are few open problem need to be solved. First, define the information loss. The great majority of preserving methods do not have a specific definition of information loss. The number of edge and node addition and deletion is used to judge anonymizaiton cost, which is unreasonable. Second, defend against attacks with multiple types of background knowledge. If we want to develop traditional anonymization methods for privacy preserving, we need to consider that adversaries have various background knowledge, which is very practical in real life. Differential privacy can overcome some disadvantages of the traditional methods. For example, it does not based on any background knowledge and can quantify the level of privacy preserving as well. However, we cannot apply it directly, because the sensitivity of social networks is very high. How to reduce the sensitivity with less noise is a key research problem in the future.

References

[1] L. Backstrom, C. Dwork, and J. Kleinberg, "Wherefore art thou r3579x?: anonymized social networks, hidden patterns, and structural steganography," in *Proc. 16th International Conference on World Wide Web*, New York, USA, 2007, pp.181–190. doi: 10.1145/2043174.2043199.

[2] C.-H. Tai, P. S. Yu, D.-N. Yang, and M.-S. Chen, "Privacy-preserving social network publication against friendship attacks," in *Proc. 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, San Diego, USA, 2011, pp. 1262–1270. doi: 10.1145/2020408.2020599.

[3] B. Zhou and J. Pei, "Preserving privacy in social networks against neighborhood attacks," in *IEEE 24th International Conference on Data Engineering (ICDE)*, Toronto, Canada, 2008, pp. 506–515. doi: 10.1007/s10115-010-0311-2.

[4] B. Zhang and J. Pei, "The k-anonymity and l-diversity approaches for privacy preservation in social networks against neighborhood attacks," *Knowledge and Information Systems*, vol. 28, no. 1, pp. 47–77, 2011. doi: 10.1007/s10115-010-0311-2.

[5] M. I. H. Ninggal and J. H. Abawajy, "Neighbourhood-pair attack in social network data publishing," in *Mobile and Ubiquitous Systems: Computing, Networking, and Services*, London, England, 2014, pp. 726–731. doi: 10.1007/978-3-319-11569-6_61.

[6] Y.Wang and B. Zheng, "Preserving privacy in social networks against connection fingerprint attacks," in *IEEE 31st International Conference on Data Engi-*

neering (ICDE), Seoul, Korea, 2015, pp. 54– 65. doi: 10.1109/ICDE.2015.7113272.
[7] C. Sun, P. S. Yu, X. Kong, and Y. Fu, "Privacy preserving social network publication against mutual friend attacks," in *IEEE 13th International Conference on Data Mining Workshops (ICDMW)*, Dallas, USA, 2013, pp. 883– 890. doi: 10.1145/1217299.1217302.
[8] A. Narayanan and V. Shmatikov, "De-anonymizing social networks," in *30th IEEE Symposium on Security and Privacy*, Oakland, USA, 2009, pp. 173–187. doi: 10.1109/SP.2009.22.
[9] A. Narayanan, E. Shi, and B. I. Rubinstein, "Link prediction by de-anonymization: How we won the kaggle social network challenge," in *International Joint Conference on Neural Networks (IJCNN)*, San Jose, USA, 2011, pp. 1825–1834. doi: 10.1109/IJCNN.2011.6033446.
[10] W. Peng, F. Li, X. Zou, and J. Wu, "A two stage deanonymization attack against anonymized social networks," *IEEE Transactions on Computers*, vol. 63, no. 2, pp. 290–303, 2014. doi: 10.1109/TC. 2012.202.
[11] B. Simon, G. G. Guly_as, and S. Imre, "Analysis of grasshopper, a novel social network de-anonymization algorithm," *Periodica Polytechnica Electrical Engineering and Computer Science*, vol. 58, no. 4, pp. 161–173, 2014. doi: 10.3311/PPee.7878.
[12] S. Ji, W. Li, M. Srivatsa, J. S. He, and R. Beyah, "Structure based data de-anonymization of social networks and mobility traces," in *17th International Conference on Information Security*, Hong Kong, China, 2014, pp. 237– 254. doi: 10.1007/978-3-319-13257-0_14.
[13] L. Yartseva and M. Gross glauser, "On the performance of percolation graph matching," in *Proc. First ACM Conference on Online Social Networks*, Boston, USA, 2013, pp.119–130. doi: 10.1145/2512938.2512952.
[14] P. Pedarsani and M. Grossglauser, "On the privacy of anonymized networks," in *Proc. 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, San Diego, USA, 2011, pp. 1235– 1243. doi: 10.1145/2020408.2020596.
[15] N. Korula and S. Lattanzi, "An efficient reconciliation algorithm for social networks," *Proc. VLDB Endowment*, vol. 7, no. 5, pp. 377– 388, 2014. doi: 10.14778/2732269.2732274.
[16] C. Chiasserini, M. Garetto, and E. Leonardi, "De-anonymizing scale-free social networks by percolation graph matching," in *INFORCOM*, Chicago, USA, 2015, pp. 1571–1579. doi: 10.1109/INFOCOM.2015.7218536.
[17] A. A. Faresi, A. Alazzawe, and A. Alazzawe, "Privacy leakage in health social networks," *Computational Intelligence*, vol. 30, no. 3, pp. 514–534, 2014. doi: 10.1111/coin.12005.
[18] O. Goga, H. Lei, S. H. K. Parthasarathi, G. Friedland, R. Sommer, and R. Teixeira, "Exploiting innocuous activity for correlating users across sites," in *Proc. 22nd International Conference on World Wide Web*, Rio de Janeiro, Brazil, 2013, pp. 447–458. doi: 10.1145/2488388.2488428.
[19] S. Nilizadeh, A. Kapadia, and Y.-Y. Ahn, "Community-enhanced de-anonymization of online social networks," in *Proc. ACM SIGSAC Conference on Computer and Communications Security*, Scottsdale, USA, 2014, pp. 537–548. doi: 10.1145/2660267.2660324.
[20] K. Sharad and G. Danezis, "An automated social graph de-anonymization technique," in *Proc. 13th Workshop on Privacy in the Electronic Society*, Scottsdale, USA, 2014, pp. 47–58. doi: 10.1145/2665943.2665960.
[21] L. Dong, Y. Li, H. Yin, H. Le, and M. Rui, "The algorithm of link prediction on social network," *Mathematical Problems in Engineering*, vol. 2013, article ID 125123, 2013. doi: 10.1155/2013/125123.
[22] N. Gupta and A. Singh, "A novel strategy for link prediction in social networks," in *Proc. 2014 CoNEXT on Student Workshop*, Sydney, Australia, 2014, pp. 12–14. doi: 10.1145/2680821.2680839.
[23] P. Sarkar, D. Chakrabarti, and M. Jordan. (2012). *Nonparametric link prediction in dynamic networks* [Online]. Available: http://arxiv.org/abs/1206.6394
[24] V. Malviya and G. P. Gupta, "Performance evaluation of similarity-based link prediction schemes for social network," in *1st International Conference on Next Generation Computing Technologies (NGCT)*, Dehradun, India, 2015, pp. 654–659. doi: 10.1109/NGCT. 2015.7375202.
[25] L. Duan, C. Aggarwal, S. Ma, R. Hu, and J. Huai, "Scaling up link prediction with ensembles," in *Proc. Ninth ACM International Conference on Web Search and Data Mining*, California, USA, 2016, pp. 367–376. doi: 10.1002/asi.v58:7.
[26] M. Al Hasan and M. J. Zaki, "A survey of link prediction in social networks," in *Social Network Data Analytics*, C. C. Aggarwal, Ed. Minneapolis, USA: Springer US, 2011, pp. 243–275. doi: 10.1007/978-1-4419-8462-3_9.
[27] Y. Dhote, N. Mishra, and S. Sharma, "Survey and analysis of temporal link prediction in online social networks," in *International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, Mysore, India, 2013, pp. 1178–1183. doi: 10.1109/ICACCI. 2013.6637344.
[28] P. Samarati and L. Sweeney, "Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression," SRI International Technical Report, Menlo Park, USA, Tech. Rep., 1998.
[29] K. Liu and E. Terzi, "Towards identity anonymizationon graphs," in *Proc. ACMSIGMOD International Conference on Management of Data*, Vancouver, Canada, 2008, pp. 93–106. doi: 10.1145/1117454.1117456.
[30] P. Liu and X. Li, "An improved privacy preserving algorithm for publishing social network data," in *IEEE International Conference on High Performance Computing and Communications & Embedded and Ubiquitous Computing*, Zhangjiajie, China, 2013, pp. 888 – 895. doi: 10.1109/HPCC.and.EUC. 2013.127.
[31] Y. Wang, L.Xie, B. Zheng, and K. C. Lee, "High utility k-anonymization for social network publishing," *Knowledge and Information Systems*, vol. 41, no. 3, pp. 697–725, 2014. doi: 10.1007/s10115-013-0674-2.
[32] Y. Wang, L. Xie, B. Zheng, and K. C. Lee, "Utility-oriented k-anonymization on social networks," in *16th International Conference on Database Systems for Advanced Applications*, Hong Kong, China, 2011, pp. 78–92.
[33] M. I. H. Ninggal and J. H. Abawajy, "Utility-aware social network graph anonymization," *Journal of Network and Computer Applications*, vol. 56, pp. 137–148, 2015. doi: 10.1016/j.jnca.2015.05.013.
[34] C.-H. Tai, S. Y. Philip, D.-N. Yang, and M.-S. Chen, "Structural diversity for privacy in publishing social networks," in *SIAM International Conference on Data Mining*, Mesa, USA, 2011, pp. 35–46. doi: 10.1137/1.9781611972818.4.
[35] R. Okada, C. Watanabe, and H. Kitagawa, "A k-anonymization algorithm on social network data that reduces distances between nodes," in *IEEE 33rd International Symposium on Reliable Distributed Systems Workshops (SRDSW)*, Nara, Japan, 2014, pp. 76–81. doi: 10.1109/SRDSW.2014.19.
[36] Y. Wang, F. Qiu, F. Wu, and G. Chen, "Resisting label-neighborhood attacks in outsourced social networks," in *Performance Computing and Communications Conference (IPCCC)*, Austin, USA, 2014, pp. 1–8. doi: 10.1109/PCCC. 2014.7017106.
[37] B. Zhou, J. Pei, and W. Luk, "A brief survey on anonymization techniques for privacy preserving publishing of social network data," *ACM Sigkdd Explorations Newsletter*, vol. 10, no. 2, pp. 12– 22, 2008. doi: 10.1145/1540276.1540279.
[38] V. K. Sihag, "A clustering approach for structural k-anonymity in social networks using genetic algorithm," in *Proc. CUBE International Information Technology Conference*, Pune, India, 2012, pp. 701– 706. doi: 10.1145/2381716.2381850.
[39] A. Campan and T. M. Truta, "Data and structural k-anonymity in social networks," in *Second ACM SIGKDD International Workshop PinKDD*, Las Vegas, USA, 2009, pp. 33–54. doi: 10.1007/978-3-642-01718-6_4.
[40] T. Tassa and D. J. Cohen, "Anonymization of centralized and distributed social networks by sequential clustering," *IEEE Transactions on Knowledge and Data Engineering*, vol. 25, no. 2, pp. 311–324, 2013. doi: 10.1109/TKDE.2011.232.
[41] E. Zheleva and L. Getoor, "Preserving the privacy of sensitive relationships in graph data," in *1st ACM SIGKDD International Conference on Privacy, Security, And Trust in KDD*, San Jose, USA, 2008, pp. 153– 171. doi: 10.1145/1117454.1117456.
[42] A. Machanavajjhala, J. Gehrke, D. Kifer, and M. Venkitasubramaniam, "l-diversity: Privacy beyond k-anonymity," in *Proc. 22nd IEEE International Conference on Data Engineering (ICDE)*, Washington, USA, 2006, pp. 24–24. doi: 10.1109/2006. doi: 10.1109/ICDE.2006.1.
[43] A. Machanavajjhala, J. Gehrke, D. Kifer, and M. Venkitasubramaniam, "l-diversity: Privacy beyond k-anonymity," *ACM Transactions on Knowledge Discovery from Data*, vol. 1, no. 1, 2007. doi: 10.1145/1217299.1217302.
[44] M. Yuan, L. Chen, P. S. Yu, and T. Yu, "Protecting sensitive labels in social network data anonymization," *IEEE Transactions on Knowledge and Data Engineering*, vol. 25, no. 3, pp. 633–647, 2013. doi: 10.1109/TKDE.2011.259.
[45] J. Jiao, P. Liu, and X. Li, "A personalized privacy preserving method for publishing social network data," in *11th Annual Conference on Theory and Applications of Models of Computation*, Chennai, India, 2014, pp. 141– 157. doi: 10.1007/978-3-319-06089-7_10.
[46] K. Chen, H. Zhang, B. Wang, and X. Yang, "Protecting sensitive labels in weighted social networks," in *Web Information System and Application Conference (WISA)*, Yangzhou, China, 2013, pp. 221– 226. doi: 10.1109/WISA.2013.50.
[47] M. Rajaei, M. S. Haghjoo, and E. K. Miyaneh, "Ambiguity in social network data for presence, sensitive attribute, degree and relationship privacy protection," *PLOS ONE*, vol. 10, no. 6, 2015. doi: 10.1371/journal.pone.0130693.
[48] P. Mittal, C. Papamanthou, and D. Song. (2012). *Preserving link privacy in so-*

cial network based systems [Online]. Available: http://arxiv.org/abs/1208.6189
[49] A. M. Fard, K. Wang, and P. S. Yu, "Limiting link disclosure in social network analysis through sub graph-wise perturbation," in *Proc. 15th International Conference on Extending Database Technology*, Berlin, Germany, 2012, pp. 109–119. doi: 10.1109/ICDE.2011.5767905.
[50] A. M. Fard and K. Wang, "Neighborhood randomization for link privacy in social network analysis," *World Wide Web*, vol. 18, no. 1, pp. 9–32, 2015. doi: 10.1007/s11280-013-0240-6.
[51] X. Ying and X. Wu, "On link privacy in randomizing social networks," *Knowledge and Information Systems*, vol. 28, no. 3, pp. 645–663, 2011.
[52] Dwork, "Differential privacy," in *International Colloquium on Automata, Languages and Programming*, Venice, Italy, 2006, pp. 1–12.
[53] J. Blocki, A. Blum, A. Datta, and O. Shefet, "Differentially private data analysis of social networks via restricted sensitivity," in *Proc. 4th Conference on Innovations in Theoretical Computer Science*, Berkeley, USA, 2013, pp. 87–96. doi: 10.1145/2422436.2422449.
[54] S. Chen and S. Zhou, "Recursive mechanism: towards node differential privacy and unrestricted joins," in *Proc. ACM SIGMOD International Conference on Management of Data*, New York, USA, 2013, pp. 653–664. doi: 10.1145/2463676.2465304.
[55] S. P. Kasiviswanathan, K. Nissim, S. Raskhodnikova, and A. Smith, "Analyzing graphs with node differential privacy," in *Theory of Cryptography*, Tokyo, Japan, 2013, pp. 457–476. doi: 10.1007/978-3-642-36594-2_26.
[56] D. Proserpio, S. Goldberg, and F. McSherry, "A work flow for differentially-private graph synthesis," in *Proc. ACM Workshop on Online Social Networks*, Helsinki, Finland, 2012, pp. 13–18. doi: 10.1145/2342549.2342553.
[57] A. Sala, X. Zhao, C. Wilson, H. Zheng, and B. Y. Zhao, "Sharing graphs using differentially private graph models," in *Proc. ACM SIGCOMM Conference on Internet Measurement Conference*, Berlin, Germany, 2011, pp. 81–98. doi: 10.1007/s00778-006-0039-5.
[58] Y. Wang and X. Wu, "Preserving differential privacy in degree-correlation based graph generation," *Transactions on Data Privacy*, vol. 6, no. 2, pp. 127–145, Aug. 2013. doi: 10.1145/1866739.1866758.
[59] Q. Xiao, R. Chen, and K.-L. Tan, "Differentially private network data release via structural inference," in *Proc. 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, New York, USA, 2014, pp. 911–920. doi: 10.1007/s00778-013-0344-8.
[60] M. Kapralov and K. Talwar, "On differentially private low rank approximation," in *Proc. 24h Annual ACM-SIAM Symposium on Discrete Algorithms*, New Orleans, USA, 2013, pp. 1395–1414. doi: 10.1137/1.9781611973105.101.
[61] Y. Wang, X. Wu, and L. Wu, "Differential privacy preserving spectral graph analysis," in *17th Pacific-Asia Conference on Knowledge Discovery and Data Mining*, Gold Coast, Australia, 2013, pp. 329–340. doi: 10.1007/978-3-642-37456-2_28.
[62] F. Ahmed, R. Jin, and A. X. Liu. (2013). *A random matrix approach to differential privacy and structure preserved social network graph publishing* [Online]. Available: http://arxiv.org/abs/1307.0475
[63] P. Mahadevan, D. Krioukov, K. Fall, and A. Vahdat, "Systematic topology analysis and generation using degree correlations," *ACM SIGCOMM Computer Communication Review*, vol. 36, no. 4, pp. 135–146, 2006. doi: 10.1145/1159913.1159930.
[64] K. Nissim, S. Raskhodnikova, and A. Smith, "Smooth sensitivity and sampling in private data analysis," in *Proc. Thirty-Ninth Annual ACM Symposium on Theory of Computing*, San Diego, USA, 2007, pp. 75–84. doi: 10.1145/1250790.1250803.
[65] C. Task and C. Clifton, "What should we protect? defining differential privacy for social network analysis," in *State of the Art Applications of Social Network Analysis*, Springer, 2014, pp. 139–161. doi: 10.1007/978-3-319-05912-9_7.
[66] M. Hay, C. Li, G. Miklau, and D. Jensen, "Accurate estimation of the degree distribution of private networks," in *Ninth IEEE International Conference on Data Mining (ICDM)*, Miami, USA, 2009, pp. 169–178. doi: 10.1109/ICDM.2009.11.
[67] D. Kifer and A. Machanavajjhala, "No free lunch in data privacy," in *Proc. ACM SIGMOD International Conference on Management of Data*, Bangalore, India, 2011, pp. 193–204. doi: 10.1145/1217299.1217301.
[68] J. Cheng, A. W.-C. Fu, and J. Liu, "K-isomorphism: privacy preserving network publication against structural attacks," in *Proc. ACMSIGMOD International Conference on Management of Data*, Indianapolis, USA, 2010, pp. 459–470. doi: 10.1145/1807167.1807218.
[69] L. Zou, L. Chen, and M. T.Ozsu, "K-automorphism: a general framework for privacy preserving network publication," *Proceedings of the VLDB Endowment*, vol. 2, no. 1, pp. 946–957, 2009. doi: 10.14778/1687627.1687734.

Manuscript received: 2016-02-17

Biographies

YANG Mengmeng (ymengm@deakin.edu.au) received her BE from Qingdao Agricultural University, China in 2007, and M.Eng from Shenyang Normal University, China in 2014. She is currently a PhD candidate in the School of Information Technology, Deakin University, Australia. Her research interests include privacy preserving, network security and machine learning.

ZHU Tianqing (t.zhu@deakin.edu.au) received her BE and ME degrees fromWuhan University, China, in 2000 and 2004, respectively, and a PhD degree from Deakin University in Computer Science, Australia, in 2014. She is currently a continuing teaching scholar in the School of Information Technology, Deakin University, Australia. Her research interests include privacy preserving, data mining and network security. She has won the best student paper award in PAKDD 2014.

ZHOU Wanlei (wanlei.zhou@deakin.edu.au) received his BE and ME degrees from Harbin Institute of Technology, China in 1982 and 1984, respectively, and a PhD degree from The Australian National University, Australia, in 1991, all in Computer Science and Engineering. He also received a DSc degree from Deakin University in 2002. He is currently the Alfred Deakin Professor and Chair Professor in Information Technology, School of Information Technology, Deakin University. His research interests include distributed systems, network security, bioinformatics, and e-learning. Professor Zhou has published more than 300 papers in refereed international journals and refereed international conferences proceedings, including over 30 articles in IEEE journal in the last 5 years.

XIANG Yang (yang.xiang@deakin.edu.au) received his PhD in Computer Science from Deakin University, Australia. He is the Director of Centre for Cyber Security Research, Deakin University. He is the Chief Investigator of several projects in network and system security, funded by the Australian Research Council (ARC). His research interests include network and system security, data analytics, distributed systems, and networking. He has published more than 200 research papers in many international journals and conferences. Two of his papers were selected as the featured articles in the April 2009 and the July 2013 issues of *IEEE Transactions on Parallel and Distributed Systems*. Two of his papers were selected as the featured articles in the Jul/Aug 2014 and the Nov/Dec 2014 issues of *IEEE Transactions on Dependable and Secure Computing*.