Graph Matching Based Privacy-Preserving Scheme in Social Networks

Hongyan Zhang[1,2], Xiaolin Li[1], Jiayu Xu[1], and Li Xu[1(✉)]

[1] College of Computer and Cyber Security, Key Laboratory of Network Security and Cryptology, Fujian Normal University, Fuzhou 350117, Fujian, People's Republic of China
xuli@fjnu.edu.cn

[2] Concord University College Fujian Normal University, Fuzhou 350117, Fujian, People's Republic of China

Abstract. The increasing popularity of social networks has inspired recent research to explore social graphs for data mining. Because social graph data contains sensitive information about users, publishing the graph data directly will cause privacy leakage of users. In this paper, we assume that attackers might re-identify targets with 1-neighborhood graph attacks. To prevent such attacks, we propose a Graph Matching based Privacy-preserving Scheme, named GMPS, to anonymize the social graphs. We utilize Jensen-Shannon Divergence to compute node structure similarity to improve the accuracy of node clustering. And then, utilize the graph modification to achieve k-anonymity and use graph matching to measure the similarity of graphs. The experiment results on HepTh and Facebook show that the proposed approach achieves k-anonymity with low information loss and high data utility.

Keywords: Social networks · 1-neighborhood attack · Jensen-Shannon Divergence · Graph matching · Privacy preserving

1 Introduction

The growing popularity of social networks has prompted recent research to explore social graph data to understand its structure for advertising, data mining and so on. The large amount of personal data that users share on social networks makes them a desirable target for attackers [1]. Releasing social graph data directly will compromise users' privacy, resulting the risk of uses' property and personal safety. Therefore, preserving the privacy of users has become a challenges for social graph data publishing [2].

Social networks use nodes and edges to model social relations with graph structure, where nodes represent users and edges represent relations between users [3]. Normally, data owners may release their data after Navïve anonymization, which just remove nodes' identities before data publishing. However, Navïve

L. Lin et al. (Eds.): SocialSec 2021, CCIS 1495, pp. 110–118, 2021.
https://doi.org/10.1007/978-981-16-7913-1_8

anonymization cannot protect users' privacy sufficiently, while adversaries may have some background knowledge about users, i.e., degree, the 1-neighborhood graphs. Based on background knowledge, there exist re-identification or de-anonymization attacks [4–6] against graph structure, the attacks can be divided into degree attacks [7], neighborhood graph attacks [8], subgraph attacks [9]. To defend de-anonymous attacks, k-anonymity technique has been utilized by many researchers. The k-anonymity technique used in graph data publishing is implemented by adding or deleting nodes and edges to make that there are other $k-1$ 1-neighborhood graphs isomorphic to one node. In this paper, we consider the adversary has the background with the 1-neighborhood graphs of users, because it is more difficult for an attacker to collect the information beyond a one-hop neighborhood [10].

The main contributions of this paper are summarized as follows:

1. To achieve k-anonymity, we divided nodes into several clusters in which the sizes are between $[k, 2k)$. We utilize Jensen-Shannon Divergence to compute node structure similarity to improve the accuracy of node clustering.
2. To measure graph anonymity, we use graph matching algorithm the accuracy to obtain the distance of 1-neighborhood graphs of each pair of nodes.
3. The experiment results on HepTh and Facebook show that the proposed approach achieves k-anonymity with low information loss and high data utility.

The rest of the paper is organized as follows. The notions, terminologies and the problem description are introduced in Sect. 3. The strategies are elaborated in Sect. 4. Section 5 gives the experimental analysis on our scheme respectively. The validation results are presented in this section as well. We conclude this paper in Sect. 6.

2 Related Works

To de-anonymous attacks, Campan and Truta [11] proposed a k-anonymity model, in which each node should be similar to at least $k-1$ nodes based on both structural information and nodes attributes, therefore, the anonymized nodes cannot be re-identified with the probability larger than $1/k$. Due to this reason, k-anonymity has become the most popular method to protect individuals privacy in social network data publishing problem [12]. To achieve k-anonymity, existing approaches can be classified into clustering-based and graph modification approaches [10].

For clustering-based models, similar nodes and edges into groups to form super nodes, a super node represent a subgraph which incorporates certain similar nodes and the edges between them, the edges between super nodes represent the relationship between subgraphs. Since a clustered graph only contains super nodes and super edges, by making the size of each cluster at least k, the probability to reidentify a user can be bounded to be at most $1/k$. Campan and Truta [11] discussed how to implement clustering method when consider the lost of both node labels and structure information. Zheleva and Getoor [13] developed

a clustering method to prevent the sensitive link leakage. Cormode et al. [14] introduced (k, l)-clustering for bipartite graphs and interaction graphs, respectively.

Graph modification aims to alter graph structure to achieve k-anonymity for privacy preservation. Liu and Terzi [7] proposed a approach to make node degree achieve k-anonymity, that is, each node has at least other $k-1$ nodes with the same degree. Zhou and Pei [8] proposed a scheme to against the 1-neighborhood attack. For each vertex v, its 1-neighborhood graph is isomorphic 1-neighborhood graphs to $k-1$ other nodes. Zou, Chen and Ozsu [9] proposed a k-automorphism scheme to preserve privacy. Cheng, Fu and Liu [15] proposed two targets of attacks, NodeInfo and LinkInfo. Then they proposed a scheme to form k pairwise isomorphic subgraphs. Yuan et al. [16] defined a k-degree-l-diversity anonymity model to protect not only the sensitive labels of individuals but also the structural information. Li et al. [17] proposed a graph based framework to preserve privacy in data publication. Based on the features of the graph, they quantified the privacy and utility measurements of the anonymous datasets. Liu et al. [10] first proposed a kind of attack named weighted 1*-neighborhood attack, which assume that attackers have some background knowledge about both individuals' 1-neighborhood graphs and the degrees of its neighbor nodes and edge weights between nodes. They proposed a heuristic indistinguishable group anonymous scheme to achieve k-anonymity. In the anonymous social graph has high graph utility. Huang et al. [18] proposed a privacy preserving approach based on the differential privacy model, which combined clustering and randomization algorithms. Moreover, they also proposed a privacy measure algorithm against graph structure and degree attacks. Ding et al. [12] proposed a new utility measurement with a new information loss matrix, based on which a k-decomposition algorithm and a privacy preserving framework are developed.

In general, existing researches about privacy preserving can protect the privacy of users for social graph data publishing. However, privacy protection needs a trade-off between privacy and data utility. In our approach, we focus on 1-neighborhood graph attack, and we can achieve better data utility meanwhile guarantee k-anonymity.

3 Preliminaries

In this paper, we use an undirected graph $G = (V, E)$ to model the social network, where V is a set of nodes, $E \subseteq V \times V$ is a set of edges. The nodes represent the users, the edges represent the relationships between users such as friendship. The cardinalities of V and E are denoted by $|V|$ and $|E|$ respectively, We assume that $|V| = n, |E| = m$.

Due to the small world phenomenon of social networks,the diameters of social networks are small, it is difficult to collect information of d-hop neighbors. We focus on 1-Neighborhood Graph attack, the adversary have knowledge about one node's edge-neighborhood graph.

Definition 1. *(1-Neighborhood Graph)* [8] $G(v) = <V(v), E(v)>$, *where* $V(v)$ *is the set of neighborhood nodes of* v *and* $V(v) = \{u|(u,v) \in E\} \cup \{v\}$. $E(v)$ *is the set of edges between the nodes in* $V(v)$, *and* $E(v) = \{(u,v)|u,v \in V(v) \wedge (u,v) \in E\}$.

4 The Proposed Approach

4.1 Node Clustering

In this section, we use k-anonymity to preserve the identities privacy of users. In order to achieve k-anonymity, the processing is divided into three steps: (1) Cluster initializing: for given nodes, according to the node degree and local cluster coefficient, we cluster the nodes in graph G into several clusters. (2) Cluster reshaping: for all clusters, compute the degree distribution similarity(DDS) between each two nodes, according to DDS, reshaping the clusters, s.t. every cluster has $[k, 2k)$ nodes. (3) Modify the 1-neighborhood graph of nodes in every cluster, s.t. the 1-neighborhood graph of nodes in the same cluster probabilistic indistinguishability.

Cluster Initializing. For a given graph $G = <V, E>$, we initially cluster nodes $v \in V$ by the following metrics: $d(v)$, $lc(v)$. Here, $d(v)$, $lc(v)$ denote the degree of the node v and the local clustering coefficient, respectively.

Definition 2. *Local clustering coefficient* $lc_v = \mu_G(v)/\omega_G(v)$, *where* $\mu_G(v)$ *and* $\omega_G(v)$ *are the numbers of triangles and triples in* $G(v)$, *respectively.*

We group nodes into a cluster if $|d(v_i) - d(v_j)| < \delta_1$ and $|lc(v_i) - lc(v_j)| < \delta_2$, δ_1, δ_2 are two pre-defined parameters, and then we obtain several clusters $C_1^1, C_2^1 \ldots C_{M_1}^1$.

Although after the above processing, the nodes are grouped into several clusters, not all the sizes of the clusters are greater than or equal to k. In reality, during the empirical study, the size of clusters follow the power-law distribution, that is, the cardinalities of most clusters are small, a small number of the clusters have thousands of members. Therefore, we execute a cluster combination process to make sure that the sizes of all clusters will be larger than k.

First, we sort all the clusters in descending order of the cardinality of the clusters, $C_1^2, C_2^2 \ldots C_{M_1}^2$. For each cluster of a size smaller than k, we incorporate the nodes in C_{i+1}^2 into C_i^2, the processing continues until every cluster has a size larger than k. Algorithm 1 shows the processing of Cluster Initializing.

Cluster Reshaping

Definition 3 *(JensenCShannon Divergence* [19]*). Suppose that we have two sets of discrete values* x_i *and* y_i *with the corresponding probability distribution,* $p(x_i)$ *and* $p(y_j)$. *The relative entropy between these two distributions is defined as* $R[p(x)||p(y)] = \sum_{i=1}^{n} p(x_i)\frac{p(x_i)}{p(y_i)}$.

Suppose that there are M_1 clusters after cluster initializing, we sort the clusters in the descending order of the maximal node degree of the members in the clusters, the sorted clusters are denoted as $C_1^{'}, C_2^{'}, \ldots C_{M_1}^{'}$, $|C_i^{'}| = n_i^{'}$. For each cluster $C_i^{'}$ which size is larger than $2k$, we perform cluster splitting to enable the size of each cluster to be $[k, 2k)$. We utilize the degree distribution similarity as the metric to slip the clusters. In order to obtain the degree distribution similarity of all the other nodes in the same cluster, first, construct the 1-neighborhood graph of nodes in the same cluster. Then, for each node $v \in C_i^{'}$, obtain the degree distribution $P(v)$ of its 1-neighborhood graph. Compute the degree distribution similarity of the these nodes, for node $v \in C_i^{'}$, compute the degree distribution similarity with all the other nodes u in $C_i^{'}$, $sim_{uv} = 1 - \frac{R_{uv}}{R_{max}}$. Then, select the $k-1$ most similar nodes of node v, add these nodes and v into the same cluster.

In order to compute the degree distribution similarity of two nodes in social networks, the processing is divided into three steps: (1) compute the nodes' degree distribution. (2) compute the relative entropy of two nodes. (3) compute the similarity of two nodes.

(1) Compute the nodes' degree distribution
Graph $G = <V, E>$, where $V = \{v_1, v_2, \cdots, v_n\}$, $E = \{e_1, e_2, \cdots, e_m\}$. $G(v_i) = <V(v_i, E(v_i))>$ is the 1-neighborhood graph of node v_i, let $N_i = |V(v_i)|$. $D_{v_i} = \{d_{i1}, d_{i2}, \ldots, d_{iN_i}\}$, D_{v_i} represent the degree sequences of nodes in $G(v_i)$, including v_i. $P_i = \{p_{i1}, p_{i2}, \ldots, p_{iN_i}\}$, where $p_{ij} = \frac{d_{ij}}{D_i}$, $D_i = \sum_{i=1}^{N_i} d_{ij}$.

(2) Compute the relative entropy of two nodes
Suppose two nodes v_i and v_j, the degree distribution of the two nodes are $P_i = \{p_{i1}, p_{i2}, \ldots, p_{iN_i}\}$, $P_j = \{p_{j1}, p_{j2}, \cdots, p_{jN_j}\}$. Even if $N_i \neq N_j$, without loss of generality, let $N_i > N_j$, we could add $|N_i - N_j|$ 0s into P_j. The relative entropy between v_i and v_j is

$$r_{ij}[P_i||P_j] = \sum_{k=1}^{N_i} p_{ik} \frac{p_{ik}}{p_{jk}}.$$

(3) Compute the similarity of two nodes
Due to the asymmetry of the relative entropy, we also need to compute $r_{ji}[P_j||P_i] = \sum_{k=1}^{N_i} p_{jk} \frac{p_{jk}}{p_{ik}}$. Let $R_{ij} = r_{ij} + r_{ji}$, $R_{max} = max(R_{ij})$, $1 \leq i, j \leq n$, the similarity of each pair of nodes can be defined as follows:

$$sim_{ij} = 1 - \frac{R_{ij}}{R_{max}}.$$

4.2 Graph Modification

After node clustering, we need to modify the graph to achieve $k-$anonymous. Suppose that there are M clusters, $C_1, C_2, \ldots C_M$, the size of each cluster is

about $[k, 2k)$. We sort the clusters in descending order of the maximal node degree, and the new ordered clusters are denoted as $\widehat{C_1}, \widehat{C_2}, \ldots \widehat{C_M}$. Then, for each pair of nodes u and v in the same cluster, we compute distance between the l-neighborhood graphs of each pair of nodes to determine whether they are structure similar. If the 1-neighborhood of all nodes in the same cluster are inexact matching, then, we achieve k-anonymity.

We use the Munkres's algorithm [20] to find the minimum cost $cost(G(u), G(v))$. If $cost(G(u), G(v)) \geq \alpha$, Find the optimal edit path $p = p_1, p_2, \ldots p_{n+m}$ of the integers $1, 2, \ldots, n+m$ which minimizes $\sum_{i=1}^{n+m} C_{ip_i}$. Then, modify the graph structure to achieve structure similarity. The process is given as Algorithm 1.

Algorithm 1. Graph Modification

Input: Graph G, α
Output: Anonymity Graph $\widetilde{G}$

```
Sort C_i, i = 1, 2, ... M with descending order of maximal node degree
obtain Ĉ_i, i = 1, 2, ... M
for each Ĉi do
    choose the first node suppose u as the seed
    for each node v in Ĉ_i − {u} do
        construct 1-neighborhood graphs of G(u) and G(v)
        compute Cost(G(u), G(v))
        if Cost(G(u), G(v)) ≥ α then
            find the optimal edit path p = p_1, p_2, ... p_{n+m}
            modify G(v) similar to G(v)
return G̃
```

5 Validation Experiment

In this section, All the experiments were conducted in Python on a server running the Ubuntu 20.04.1 LTS operating system. To demonstrate the effectiveness of our scheme, we validate the performance of the proposed scheme on two real datasets: CA-HepTh and Facebook. Details of graph characteristics are shown in Table 1.

Table 1. Details of Social Networks

Dataset	Nodes	Edges	AVD	ACC	APL
HepTh	9877	25998	5.3	0.4714	7.4
Facebook	4039	88234	44	0.605	4.7

To explore the utility of the anonymized graph $\widetilde{G}$, we test the following two metrics:

1. *Average degree(AVD)*: The AVD of G can be calculated as $\sum_{v \in V} d_v/|V|$;
2. *Average clustering coefficient(ACC)*: The ACC of G can be calculated as $\sum_{v \in V} C_v/|V|$, where C_v is the local clustering coefficient of v;

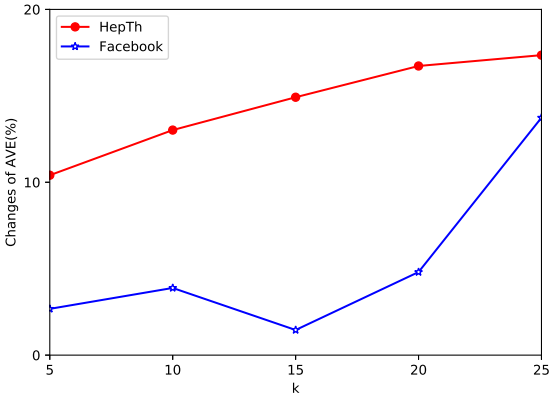

Fig. 1. AVE

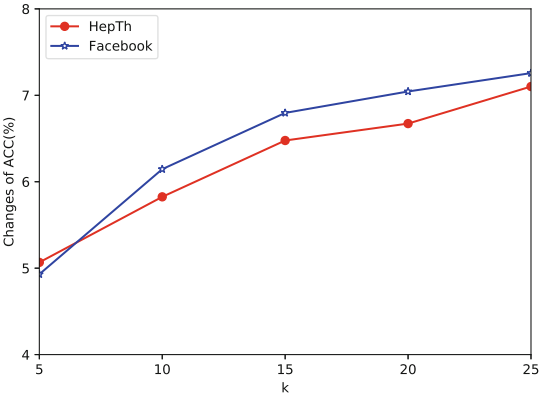

Fig. 2. ACC

Average Degree (AVE). The AVE of G can be calculated as $\sum_{v \in V} d_v/|V|$. Figure 1 shows the change of average degree of HepTh and Facebook. We can see that as k increase, change of AVE increases. The AVE change in the Facebook

is small, while HepTh is the larger. As k increase from 5 to 25, the percentage of AVE changes from 0.112 to 0.176 in Hepth, while from 0.026 to 0.137 in Facebook. The result may be caused by the ACC, the larger the ACC, the smaller the percentage of AVE changes.

Average Clustering Coefficient (ACC). Clustering coefficient is the degree of clustering between the nodes of a graph. The ACC of G can be calculated as $\sum_{v \in V} C_v/|V|$, where C_v is the local clustering coefficient of v. Figure 2 is the ACC of datasets, as k increase changes of ACC increase in both dataset. In Hepth, the change is from 5.12 to 6.93, from 4.98 to 7.05 in Facebook. Therefore, the ACC changes nearly in both datasets.

6 Conclusions

Although graph anonymization can reduce the risk of privacy disclosure, malicious attackers might have background knowledge about 1-neighborhood graph of targets, and they may re-identity the users in anonymous social graphs. In this paper, we propose a Graph Matching based Privacy-preserving Scheme, named GMPS, in Social Networks to realize social graph anonymization. We utilize JensenCShannon Divergence to compute node structure similarity to improve the accuracy of node clustering. And then, utilize the graph modification to achieve k-anonymity and use graph matching to measure the similarity of graphs. The experiment results on HepTh and Facebook show that the proposed approach achieves k-anonymity with low information loss and high data utility.

Acknowledgments. The authors would like to thank the National Science Foundation of China (Nos. U1905211, 61771140, 61702100, 61702103), Fok Ying Tung Education Foundation (No. 171061), Natural Science Foundation of Fujian Province (No. 2020J01167), Educational Research Project for Young and Middle-aged Teachers in Fujian Province (Science and Technology) (No. JAT200968).

References

1. Rathore, S., Sharma, P.K., Loia, V., Jeong, Y.-S., Park, J.H.: Social network security: issues, challenges, threats, and solutions. Inf. Sci. **421**, 43–69 (2017)
2. Kleinberg, J.M.: Challenges in mining social network data: processes, privacy and paradoxes. In: Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Jose, California, USA, pp. 4–5. ACM (2007)
3. Wang, Q., Zhang, Y., Lu, X., Wang, Z., Qin, Z., Ren, K.: Real-time and spatio-temporal crowd-sourced social network data publishing with differential privacy. IEEE Trans. Dependable Secure Comput. **15**(4), 591–606 (2018)
4. Narayanan, A., Shmatikov, V.: De-anonymizing social networks. In: 30th IEEE Symposium on Security and Privacy, Oakland, California, USA, pp. 173–187. IEEE (2009)

5. Ji, S., Mittal, P., Beyah, R.: Graph data anonymization, de-anonymization attacks, and de-anonymizability quantification: a survey. IEEE Commun. Surv. Tutor. **19**(2), 1305–1326 (2017)
6. Li, H., Chen, Q., Zhu, H., Ma, D., Wen, H., Shen, X.S.: Privacy leakage via de-anonymization and aggregation in heterogeneous social networks. Trans. Dependable Secure Comput. **17**(2), 350–362 (2020)
7. Liu, K., Terzi, E.: Towards identity anonymization on graphs. In: Proceedings of the ACM SIGMOD International Conference on Management of Data, Vancouver, BC, Canada, pp. 93–106. ACM (2008)
8. Zhou, B., Pei, J.: Preserving privacy in social networks against neighborhood attacks. In: 24th International Conference on Data Engineering, Mexico, pp. 506–515. IEEE (2008)
9. Zou, L., Chen, L., Ozsu, M.: K-automorphism: a general framework for privacy preserving network publication. Proc. VLDB Endow. **2**(1), 946–957 (2009)
10. Liu, Q., Wang, G., Li, F., Yang, S., Wu, J.: Preserving privacy with probabilistic indistinguishability in weighted social networks. IEEE Trans. Parallel Distrib. Syst. **28**(5), 1417–1429 (2017)
11. Campan, A., Truta, T.: A clustering approach for data and structural anonymity in social networks. In: 2nd ACM SIGKDD International Workshop Privacy Security Trust in KDD, USA, pp. 33–54. ACM (2008)
12. Ding, X., Wang, C., Choo, K.K.R., Jin, H.: A novel privacy preserving framework for large scale graph data publishing. IEEE Trans. Knowl. Data Eng. **33**(2), 331–343 (2021)
13. Zheleva, E., Getoor, L.: Preserving the privacy of sensitive relationships in graph data. In: Bonchi, F., Ferrari, E., Malin, B., Saygin, Y. (eds.) PInKDD 2007. LNCS, vol. 4890, pp. 153–171. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-78478-4_9
14. Cormode, G., Srivastave, D., Yu, T., Zhang, Q.: Anonymizing bipartite graph data using safe grouping. VLDB Endow. 833–844 (2008)
15. Cheng, J., Fu, A., Liu, J.: K-isomorphism: privacy preserving network publication against structural attacks. In: ACM SIGMOD International Conference on Management of Data, Indianapolis, Indiana, USA, pp. 459–470. ACM (2010)
16. Yuan, M., Chen, L., Yu, P.S., Yu, T.: Protecting sensitive lables in social network data anonymization. IEEE Trans. Knowl. Data Eng. **25**(3), 633–647 (2013)
17. Li, X.Y., Zhang, C., Jung, T., Qian, J., Chen, L.: Graph-based privacy-preserving data publication. In: IEEE INFOCOM 2016-The 35th Annual IEEE International Conference on Computer Communications, San Francisco, CA, USA, pp. 1–9. IEEE (2016)
18. Huang, H., Zhang, D., Xiao, F., Wang, K., Gu, J., Wang, R.: Privacy-preserving approach PBCN in social network with differential privacy. IEEE Trans. Netw. Serv. Manag. **17**(2), 931–945 (2020)
19. Tsai, S., Tzeng, W., Wu, H.: On the Jensen-Shannon divergence and variational distance. IEEE Trans. Inf. Theory **51**(9), 3333–3336 (2005)
20. Riesen, K., Bunke, H.: Approximation graph edit distance computation by means of bipartite graph matching. Image Vis. Comput. **27**(7), 950–959 (2009)