Personal and Ubiquitous Computing
https://doi.org/10.1007/s00779-019-01287-0

ORIGINAL ARTICLE

Utility analysis on privacy-preservation algorithms for online social networks: an empirical study

Cheng Zhang[1] · Honglu Jiang[1] · Xiuzhen Cheng[1] · Feng Zhao[2] · Zhipeng Cai[3] · Zhi Tian[4]

Received: 5 February 2019 / Accepted: 20 April 2019

Abstract

Social networks have gained tremendous popularity recently. Millions of people use social network apps to share precious moments with friends and family. Users are often asked to provide personal information such as name, gender, and address when using social networks. However, as the social network data are collected, analyzed, and re-published at a large scale, personal information might be misused by unauthorized third parties and even attackers. Therefore, extensive research has been carried out to protect the data from privacy violations in social networks. The most popular technique is graph perturbation, which modifies the local topological structure of a social network user (a vertex) via various randomization techniques before the social graph data is published. Nevertheless, graph anonymization may affect the usability of the data as random noises are introduced, decreasing user experience. Therefore, a trade-off between privacy protection and data usability must be sought. In this paper, we employ various graph and application utility metrics to investigate this trade-off. More specifically, we conduct an empirical study by implementing five state-of-the-art anonymization algorithms to analyze the graph and application utilities on a Facebook and a Twitter dataset. Our results indicate that most anonymization algorithms can partially or conditionally preserve the graph and application utilities and any single anonymization algorithm may not always perform well on different datasets. Finally, drawing on the reviewed graph anonymization techniques, we provide a brief overview on future research directions and challenges involved therein.

Keywords Online social networks · Data utility · Data anonymization

1 Introduction

As one type of the technological products of Web 2.0, online social networks (OSNs) have become the main platforms for people to share information and communicate with each other on the Internet, which significantly revolutionizes the way people interact. When using OSNs, users are asked to provide personal information such as name, gender, date of birth, address, marital status, and e-mail. Meanwhile, geographical information, photos, videos, and interactions among friends are also stored by OSN service providers. The collection of the above information leads to the generation of massive nonlinear, large-capacity, scale-free, and high-dimensional data that can be modeled as social graphs and published to third parties for different purposes such as business analysis and academic research. Since a social graph often consists of private information of the users, the data owners typically employ graph anonymization techniques to break the linkage between an identity and its associated sensitive information via pseudonyms and various structure perturbation methods to disturb the original social graphs before publishing.

Typically, graph perturbation is performed by adding random noises to edges of a social graph, and the stronger the noise, the stronger the privacy protection strength. Nevertheless, the more noises added to a social graph, the less usable the graph data to the end users. Therefore, a trade-off between privacy preservation and data usability in anonymized social graphs must be sought. On the other hand, different anonymization algorithms employ different mechanisms to perturb social graphs; thus, it is necessary to analyze the impact of different anonymization algorithms on data usability, as this analysis can help understand how an anonymization algorithm affects social graph usability and make contributions to designing new anonymization algorithms that can better trade-off privacy protection and graph data usability. Moreover, most existing research focuses on a single anonymization technique; thus, a comparison

✉ Feng Zhao
zhaofeng@guet.edu.cn

Extended author information available on the last page of the article.

Published online: 07 August 2019

study on the trade-off level exploited by popular anonymization algorithms over the same social network datasets is desperately needed.

In this paper, we provide an empirical study to investigate the trade-off between privacy protection and data usability of a few popular anonymization algorithms. More specifically, we implement five structure-based anonymization algorithms, namely Random Add/Del [42], Random Switch [42], Clustering [36], K-Degree [3], and Random Walk [25], with different privacy parameters, to anonymize two real-world social graphs that have different scales and social structures and perform a comparison study. Note that pseudonyms are also employed in the anonymized graphs. To quantify the trade-off between privacy protection and data usability preservation, we employ *utility* as a performance metric and consider a few graph utility and application utility metrics. Our evaluation results indicate that most anonymization algorithms can partially preserve the utility metrics under our consideration with small noise ratios. Some anonymization algorithms have the ability to preserve a specific structural property but it may destroy other structural properties, and the topological structure of a social graph is able to affect the utility preservation of an anonymized graph. We also present a few open problems and outstanding challenges on graph anonymization.

The rest of the paper is organized as follows. In Section 3, we detail our social network model and present the popular utility measurement metrics. In Section 4, we implement different anonymization algorithms on two real-world social graphs and analyze our experimental results over different utility metrics. Open problems and future research challenge are reported in Section 5. Section 6 concludes this paper.

2 Graph anonymization techniques

In this section, we briefly summarize the major existing graph anonymization techniques, which are divided into six categories according to [17], namely, naive identity removal (pseudonyms), edge randomization [42], K-anonymity [3, 5, 14, 23, 33, 44, 45], clustering [12, 36], differential privacy [8, 18, 22, 39], and Random Walk [24].

Naive identity removal is the simplest method to anonymize network data, but it cannot provide sufficient effective protection over privacy. It has been proven to be vulnerable to structure-based de-anonymization attacks which can map a vertex from the anonymized social graph to a vertex in the original social graph (a.k.a., reference graph) based on the local structure similarity of these two vertices in their corresponding social graphs [20, 26, 27, 32, 37, 41].

Edge randomization can be realized via Random Add/Del or Random Switch [42]. Random Add/Del is an approach to remove a random subset of edges from the original graph and then add the same amount of random edges into the modified one. Random Switch first randomly chooses two distinct existing edges $e_{i,j}$ and $e_{x,y}$, then switch their endpoints, forming two new edges $e_{i,x}$ and $e_{j,y}$ that are not present in the original graph. One of the advantages of edge randomization is that it can resist de-anonymization attacks under probabilistic conditions [42]. However, both approaches could disconnect a graph and introduce a mass of noise that can significantly decrease the data usability.

Random Walk-based anonymization protects edge privacy by replacing the edge $e_{i,j}$ between vertices v_i and v_j with an edge between v_i and the endpoint of a random walk which starts from v_j.

K-Anonymity was first presented in [34] in 1998. It requires that the published data contains a certain number (at least K) of indistinguishable records so that an attacker cannot distinguish one record belonging to any individual from others, thus protecting individuals' privacy. Many variants of K-anonymity have been proposed to protect graph data in the past years, including K-degree Anonymization [3, 23], K-neighborhood Anonymization [44], K-isomorphism Anonymization [5], and K-automorphism Anonymization [45], just to name a few. K-Degree Anonymization was proposed to defend against degree attacks [23]. It first modifies the degree sequence of a graph to generate a K-anonymous degree sequence in which each degree appears at least K times, then constructs an anonymized graph based on the newly generated K-anonymous sequence. K-Neighborhood Anonymization was designed to protect privacy against neighborhood attacks [44] by first grouping vertices with similar neighborhoods together, then anonymizing the neighborhoods to be isomorphic in the same group. This process can be done in two steps. In the first step, the neighborhoods of all vertices in a graph are extracted and a coding technique is employed to label and index the neighborhoods for improving the performance of calculating the structure similarity between two neighborhoods and identifying isomorphic neighborhoods. In the second step, at least K vertices with high neighborhood similarity are greedily organized into a group and then each group is anonymized so that any neighborhood has at least K-1 isomorphic neighborhoods in the same group. In order to defend against structural attacks such as degree attacks, subgraph attacks, 1-neighbor-graph attacks, and hub-fingerprint attacks [13, 23, 44], K-isomorphism Anonymization was proposed in [5], in which a social graph is first partitioned into K disjoint subgraphs with the same number of vertices, then these K subgraphs are modified by adding or deleting edges to make all K subgraphs isomorphic. Similarly, K-automorphism Anonymization [45] can also protect privacy against various structural attacks. This anonymization method guarantees that, for any vertex

in a graph, there are always other K-1 symmetric vertices with respect to K-1 automorphic functions. From the comparison study carried out by [17], it can be seen that K-neighborhood, K-isomorphism, and K-automorphism are all with high time complexities which drastically reduce their anonymization efficiency. In this study, we implement K-degree [3] only for its simplicity for our utility analysis.

Clustering/class-based techniques [2, 36] are used to anonymize users into clusters (equivalently, groups or classes). They partition vertices into different classes according to various similarity criteria such as Euclidean distance. To achieve privacy protection, the commonality and similarity of the data within a cluster and the differences of characteristics among different clusters are exploited. Therefore, within each cluster, users' distinguishing features need to be hidden. The idea of reducing/eliminating individuals' characteristics difference is a commonly used method of data hiding nowadays. In [2], Bhagat et al. presented a label list anonymization, which groups the vertices into classes based on their "labels" (i.e., attributes). In [36], a bounded t-means clustering algorithm and a union-split clustering scheme were first presented to effectively cluster similar vertices into groups, then an inter-cluster matching method was employed to anonymize the social networks by strategically adding and removing edges based on vertices' inter-cluster connectivity. In this paper, we adopt the bounded t-means clustering algorithm in [36] to implement our clustering anonymization.

Differential privacy was first introduced by Dwork and McSherry [9] in 2005. It was initially developed for traditional databases, and has been widely used in various traditional data analysis tasks before being applied to social network data. By adding carefully calculated random noises to query results, differential privacy ensures that changes to any individual record in the database do not statistically distinguish between query results. This model has two advantages. On one hand, it does not need to take into account the background knowledge possessed by any attacker; on the other hand, it relies on a sound mathematical foundation. The key parameter of differential privacy is the privacy budget, which determines the crucial trade-off between the privacy preservation level and the data utility of social network data. How to choose the budget value to maximize the preservation of private data while protecting data usability is a great challenge. For social networks, two main definitions of "neighbor" for differential privacy were introduced, namely edge differential privacy [11, 31] and node differential privacy [4, 7]. Differential privacy was applied to not only the simple statistical analysis of attribute information such as vertex degree distribution and attribute value distribution [19, 40], but also the more complicated analysis on social network structure information [29]. In this empirical study, we choose not to implement any differential privacy algorithm as it applies to query results while our focus is the anonymization of the social network data via perturbation that can change the data itself.

In practice, naive identity removal is typically combined with one or more of the other graph anonymization techniques to protect privacy in social networks. As a matter of fact, edge randomization, K-anonymity, clustering, and random walk all provide privacy protection via adding noises to perturb the social network graph topological structures. More importantly, the more noises added to the original graph, the stronger the protection of the social graph. Nevertheless, the perturbation may negatively affect the usability of the published social network data, providing bad user experience. Therefore, it is important to study the trade-off between privacy protection and anonymized data usability. In this paper, we provide an empirical study to investigate this trade-off by implementing and applying five graph anonymization techniques on two large-scale real-world social network datasets and analyze their graph and application utilities.

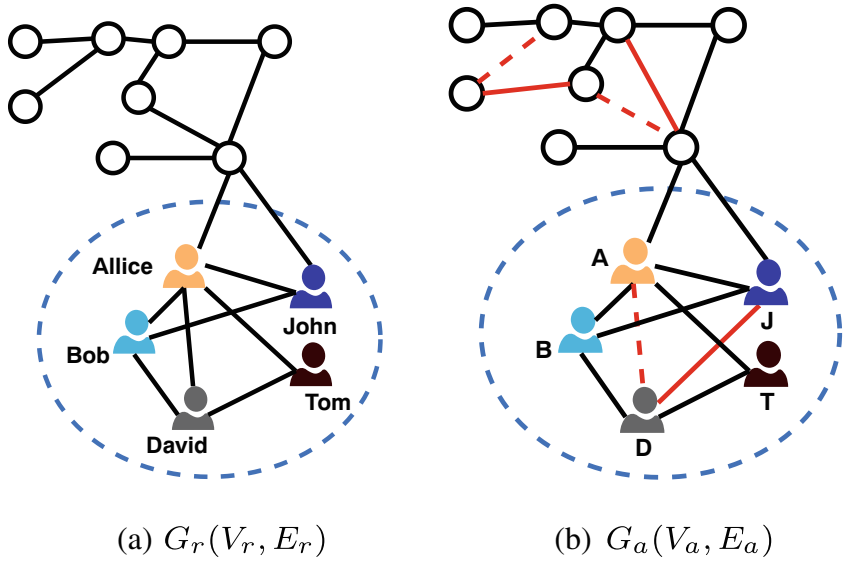
3 Network model and utility metrics

The main objective of social network anonymization is to maximize users' privacy protection level while making the network data usable as much as possible. A popular parameter to quantify the trade-off between privacy protection and data usability for an anonymized graph is utility; the higher the utility, the better the data usability; the lower the utility, the better the privacy protection. A number of utility metrics have been discussed by the existing research. In this section, we first present our social network model and then review a few common utility metrics that can be applied to anonymized social graphs.

3.1 Social network model

Generally, a social network can be modeled as a simple undirected graph $G = (V, E)$, with a set of vertices denoted as $V = \{v_1, v_2, \ldots, v_n\}$, and a set of unlabeled edges as $E = \{e_{i,j} = (v_i, v_j) | v_i, v_j \in V, i \neq j\}$. A vertex $v_i \in V$ represents a single unique user in the social network. An edge $(v_i, v_j) \in E$ signifies a social relationship between two social users v_i and v_j.

Figure 1a shows an example original social graph $G_r(V_r, E_r)$ with real identities, a.k.a. reference graph. Figure 1b represents the corresponding anonymized social graph $G_a(V_a, E_a)$ obtained by perturbing $G_r(V_r, E_r)$ via pseudonyms and Random Add/Del: the identities in V_a (like the name) that can be used to uniquely identify vertices are replaced by random characters (pseudonyms), and E_a is generated by removing some existing edges (red dashed



(a) $G_r(V_r, E_r)$ (b) $G_a(V_a, E_a)$

Fig. 1 An example original social network graph (**a**) and the corresponding anonymized one (**b**) via pseudonyms and the Random Add/Del anonymization technique

lines) from E_r and adding new edges (red solid lines) into it (Random Add/Del).

3.2 Utility metrics

Measurements of the topology of a complex and large-scale social network are essential for characterizing, analyzing, and modeling the network. An anonymization scheme can be evaluated from two aspects: anonymization level (privacy-preserving level) and data usability level, which are quantified with a single parameter "utility." The utilities of the social graph can be categorized as *graph utilities* and *application utilities*. Different graph utilities indicate distinct preservation levels of the fundamental structural characteristics of the anonymized graph relative to the original graph, and application utilities are used to measure the usability of the anonymized graph for real applications. In this section, we discuss a few graph utility and application utility metrics.

3.2.1 Graph utility metrics

The following graph utility metrics are employed in our empirical study.

- Degree (deg). The degree of a vertex v in a graph $G = (V, E)$ is the total number of vertices adjacent to v, which is the most fundamental attribute of a vertex.

$$deg(v) = |\{u|(u, v) \in E\}| \tag{1}$$

- Global transitivity (GT). The global transitivity (global clustering coefficient) measures the degree on which two neighbors of a vertex tend to be connected. The global clustering coefficient is the ratio of all the triangles and connected triplets in the graph. A triplet includes three vertices that are connected by two (or three) edges. Here, we present the following definition based on undirected unweighed graphs which is known as transitivity.

$$C = \frac{3N_{\triangle}}{N_3} \tag{2}$$

where $N_{\triangle}$ is the number of triangles and N_3 is the number of triplets in the graph.
- Shortest path (SP). The shortest path is a fundamental characteristic of a graph. It refers to the path between any two vertices that has the minimum sum of the edge weights. When the edges are unweighted, the shortest path refers to one with a minimum number of hops.
- Closeness centrality (CC). Closeness centrality of vertex $v \in V$ is calculated as the reciprocal of the sum of the shortest path length $p(v, u)$ from v to any other vertex $u \in V$ with $(u \neq v)$. It measures how long it takes to spread information from v to all the other reachable vertices in the graph.
- Betweenness centrality (BC). The betweenness centrality BC_v of a vertex $v \in V$ is defined as the number of times all the shortest paths going through v in the graph. If a vertex v has a high betweenness centrality, obviously v holds an important position and has a significant influence over the graph.
- Eigenvector centrality (EC) [28]. Eigenvector centrality calculates the centrality for a vertex based on the centrality of its neighbors. It measures the transitive influence or connectivity of the vertices. The EC of the ith node in a graph is the ith value of the eigenvector calculated from the largest eigenvalue of the adjacency matrix of the graph.

Table 1 Structural properties of the social graphs

Dataset	Vertices	Edges	Average degree
Facebook	4039	88,234	43.69
Twitter	899,403	1,786,583	3.97

3.2.2 Application utility metrics

There exist a number of application utility metrics such as Community Detection [10], Sybil Detection [38], Structural Role Extraction [15], and Epidemic Simulation (ES) [43]. Since the anonymization algorithms under our consideration introduce noises into the graph data and perturb the graph structures, most structure features could be changed, which can be clearly reflected by the epidemic simulation results. Therefore, in this empirical study, we implement only the epidemic simulation metric to measure the application utility of the anonymized social graphs.

A social network can be a snapshot of different social relationships between human beings in the real world, and a spreading of an epidemic has a close connection to human social relationships. The modeling of the epidemics on social networks has been studied in recent years [43]. Epidemic simulations are structure-sensitive, which means that the structural features of a social network, such as communities, triangles, high-degree vertices, and shortest paths, can greatly affect the simulation results.

The susceptible-infected-susceptible (SIS) model [1, 6] has been applied to an anonymized social graph to simulate the spreading of an infection (like the total number of infected or the duration of an epidemic). The anonymized social graph, which has similar epidemic simulation results as the original one, is deemed to preserve more application utility. The equations of the model with dynamics are shown as follows:

$$\frac{S^{(t+\Delta t)} - S^{(t)}}{\Delta t} = -\beta S^{(t)} I^{(t)} + \gamma I^{t} \tag{3}$$

$$\frac{I^{(t+\Delta t)} - I^{(t)}}{\Delta t} = \beta S^{(t)} I^{(t)} - \gamma I^{t} \tag{4}$$

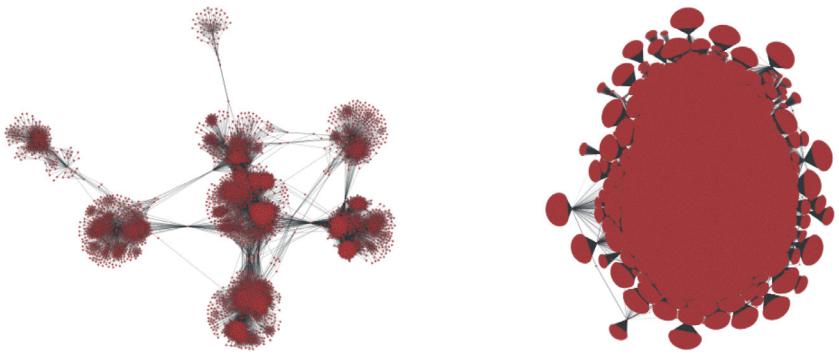
where S is the group of susceptible users who are not infected but may be infected after they meet with contagious individuals; I refers to the group of infected users; $S^{(t)}$ and $I^{(t)}$ are the numbers of users in S and I at time step t, respectively; β is the infection rate; and γ is the recovery rate. An individual in S can be infected and become a member of I; meanwhile, an individual in I can be recovered to become a member in group S.

4 Experiments and evaluation results

In this section, we provide a comprehensive empirical study to analyze the utility performance of five anonymization algorithms over two real-world social network datasets. We first present our experimental setup, then we report our simulation results.

4.1 Experimental setup

In our evaluations, five anonymization algorithms (Random Add/Delete, Random Switch, Clustering, K-degree and Random Walk) are implemented to analyze the preservation levels of the graph and application utilities on Twitter and Facebook datasets. Table 1 shows the structural properties of the networks derived from the datasets. The smaller Facebook dataset is downloaded from SNAP [21] while the larger dataset is collected from Twitter through the Twitter REST API. Figure 2 shows the visualizations of these two social graphs, which indicate that most vertices in the Facebook dataset have larger degrees but, in the Twitter dataset, 79.4% of the vertices (out of 713,805 in total) only have a single edge. This is because Breadth-first search (BFS) was used in the data crawling process, which starts from a single vertex (user) as the root and keeps on traversing all the connected "followers." After finishing crawling the



(a) Facebook (b) Twitter

Fig. 2 Visualizations of the two social graphs

root, these collected "followers" become the roots in the following crawling iterations which collect a large portion of vertices with a single edge. Since the graph structure of the Twitter dataset is too large for exhaustively computing the graph utilities of SP, CC, BC, and EC, we randomly sample 10,000 vertices from the Twitter dataset to form a smaller sized induced network for our utility evaluations.

All algorithm implementations and utility evaluations are run on an Intel i7-8750(H) at a 2.2-GHz machine with 32-G RAM running Ubuntu 18.04 LTS. All the reported results are the average of 20 runs to eliminate randomnesses.

The naive identity removal method only removes the identifiers and it does not change the graph structure; therefore, we implement it to obtain the baseline experimental results. The parameters for the other five anonymization algorithms are set as follows:

– The noise ratio of Random Add/Del and Random Switch is defined to be the ratio of the number of edges

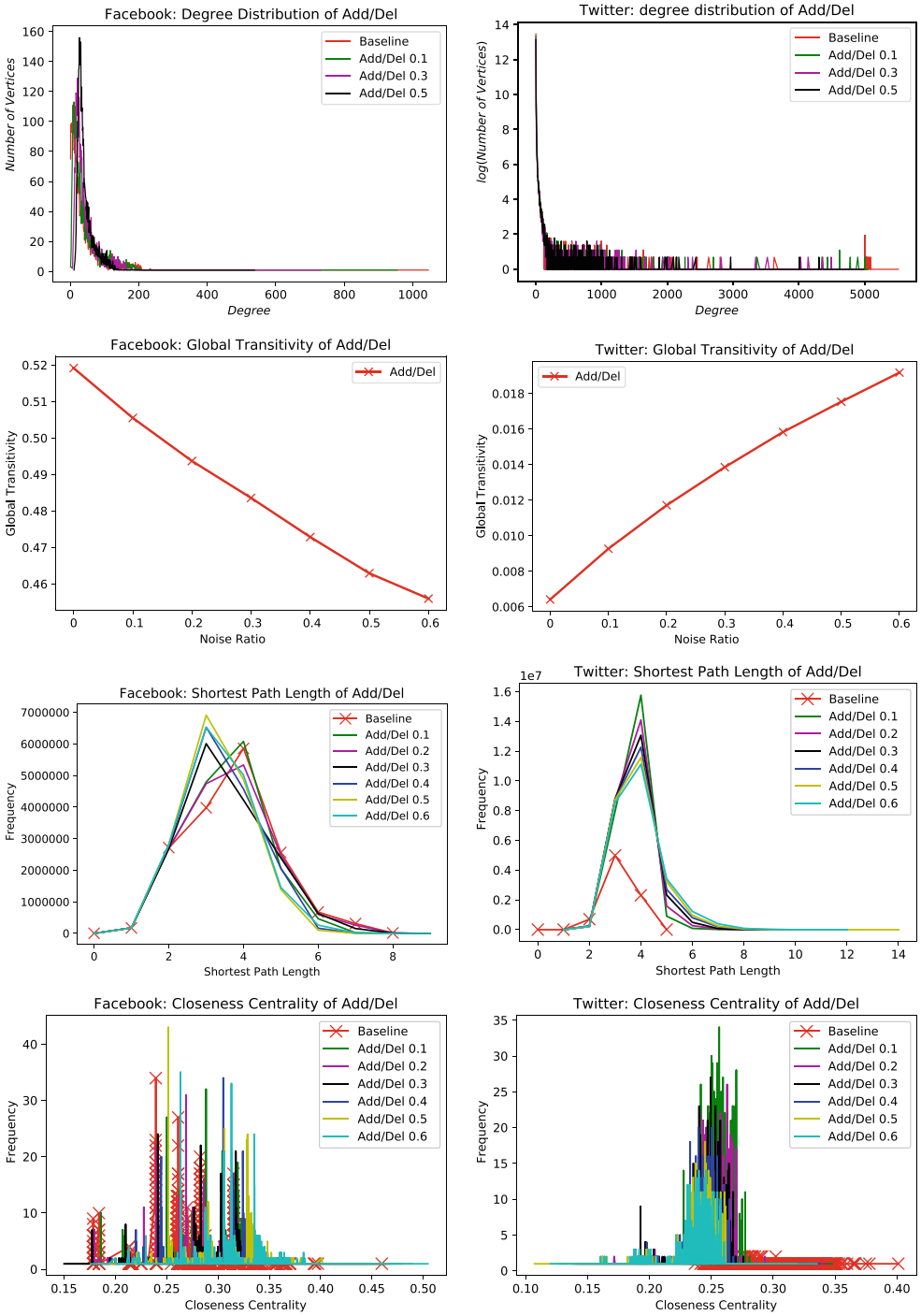
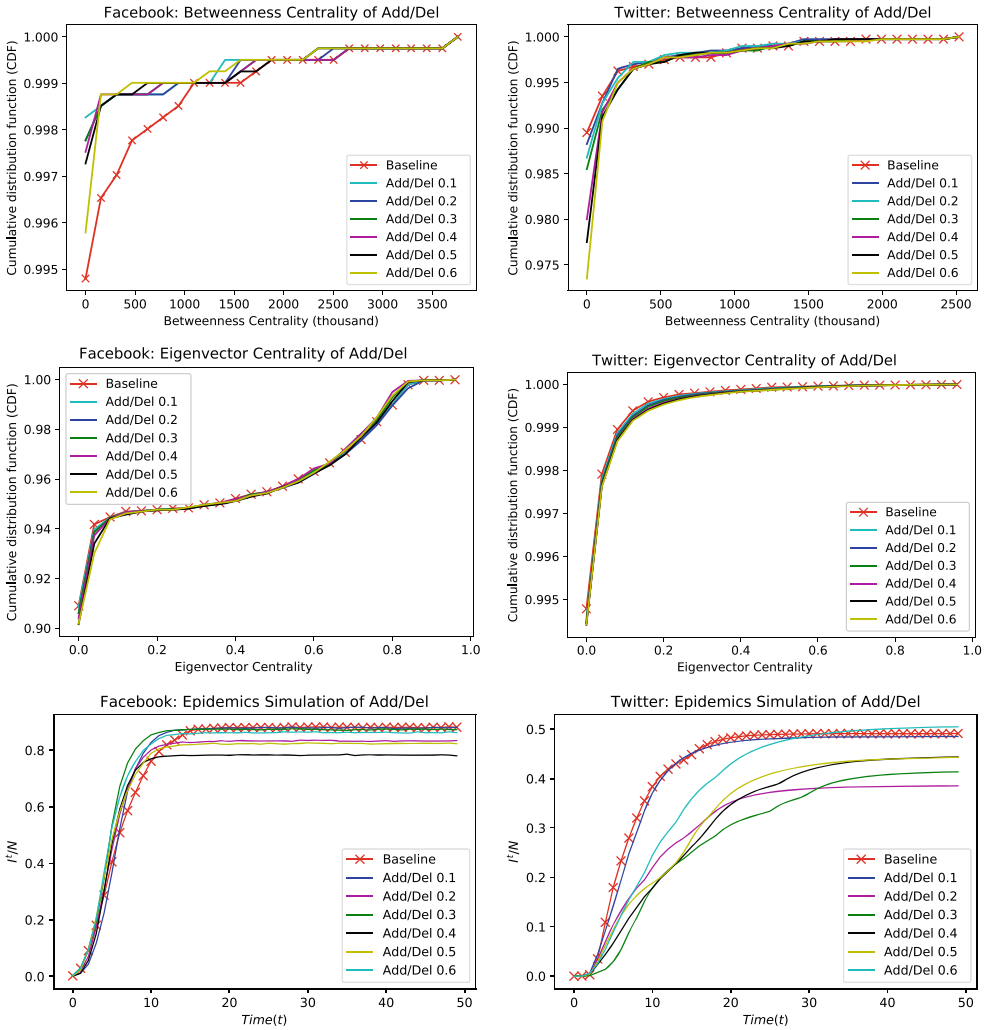


Fig. 3 Utility analysis of Random Add/Del: each row represents results of one utility on two datasets

Fig. 3 (continued)



added/removed over the total number of edges in a social graph. It ranges from 0.1 to 0.6 with an interval of 0.1 in our experimental studies.

- The parameter of K denotes the cluster size of the Clustering and the group size of the K-degree. It is set to 10, 20, 40, 60, 80, 100, 200, 300, and 400 in our studies.
- In Random Walk, w represents the step length which is set to 2, 5, and 10 in our studies.
- In Epidemic Simulation, the rate of infection $\beta = 0.2$; the rate of recovery $\gamma = 0.1$; and N is the total number of individuals with $S^{(t)} + I^{(t)} = N$.

4.2 Evaluation results

In this section, we report the utility evaluation results of each social graph by applying different anonymization algorithms to the Facebook and Twitter datasets.

For each anonymization algorithm, we plot seven pairs of figures to illustrate the utility analysis results, and each pair reports the results of one utility metric over two datasets. The first pair of figures depicts the degree distributions of the social graphs, with the x-axis being the vertex degrees in an ascending order and the y-axis being the number of vertices (note that the y-axis for the Twitter dataset is in logscale for better illustration); the second pair presents the global transitivity of the social graphs, with the x-axis being the noise ratio, cluster (group) size, or random walk length, and the y-axis being the value of global transitivity; the third pair illustrates the frequency distribution of the shortest path length, with the x-axis being the shortest path lengths in an ascending order and the y-axis showing the frequency of each value of the shortest path length; the fourth pair reports the frequency distribution of closeness centrality, with the x-axis being closeness centrality and the y-axis being frequency; the fifth and sixth pairs depict the cumulative distributions of the betweenness centrality and eigenvector centrality, respectively, with x-axis being the corresponding centrality value and the y-axis being the frequency; while the last pair reports the epidemics simulation results, with the x-axis being the time step and the y-axis being the ratio of the number of infected users

over the total number of users at each time step. In the following subsection, we detail the empirical results of each anonymization algorithm.

4.2.1 Random Add/Del

Random Add/Del randomly removes some edges and then randomly add the same amount of new edges to the graph. However, the network could get disconnected from the rest of the whole graph if some critical edges are removed. In order to prevent this, whenever the degree of a vertex drops to zero, we randomly add a new edge to that vertex; we also check the connectivity of the modified graph to ensure that the graph is still connected after Random Add/Del. Figure 3 presents the utility analysis results. One can see that the utilities such as deg, CC, and ES can be partially preserved by Random Add/Del when few edges are modified. Note that we plot the logarithm of deg for the Twitter dataset for a

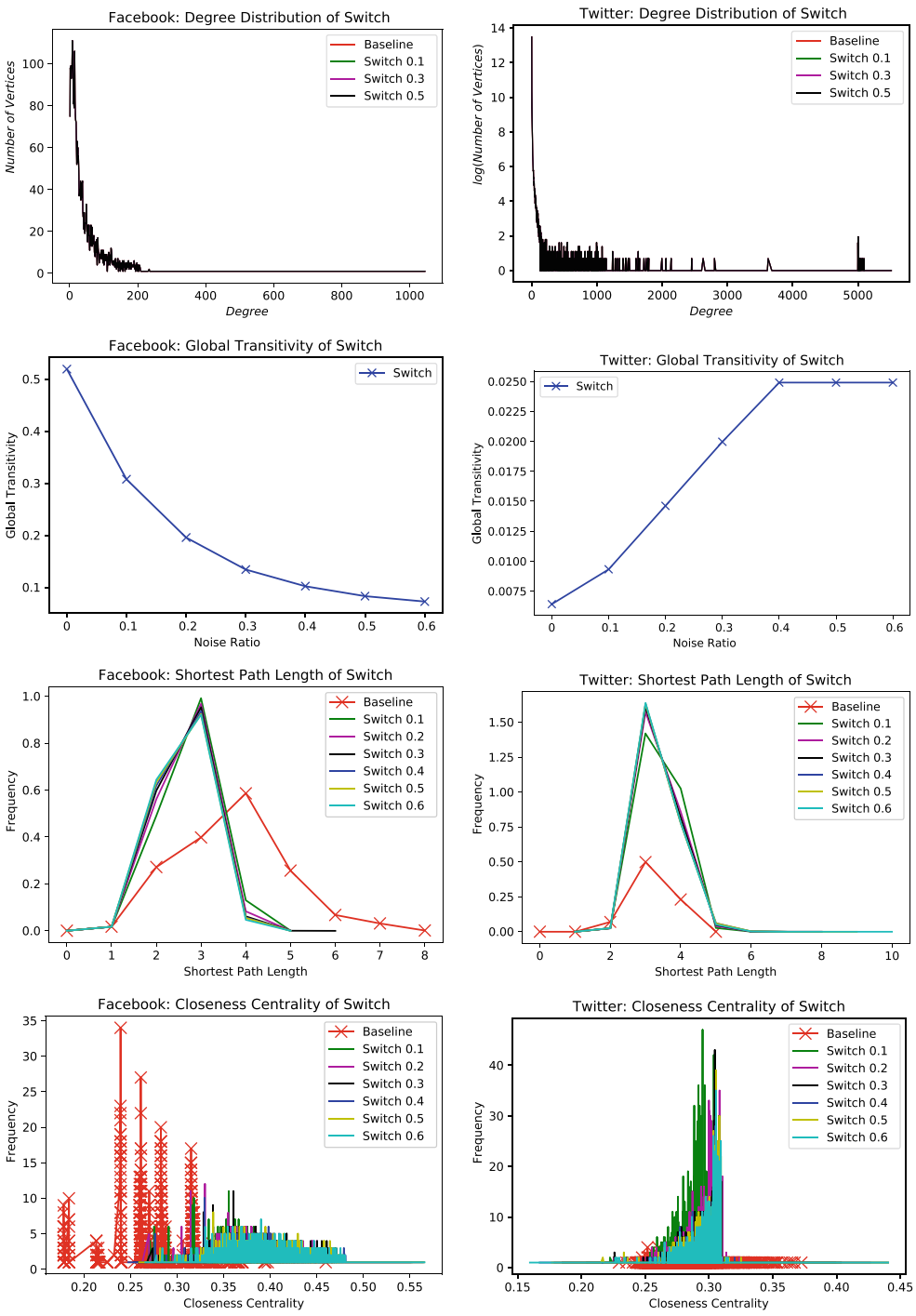
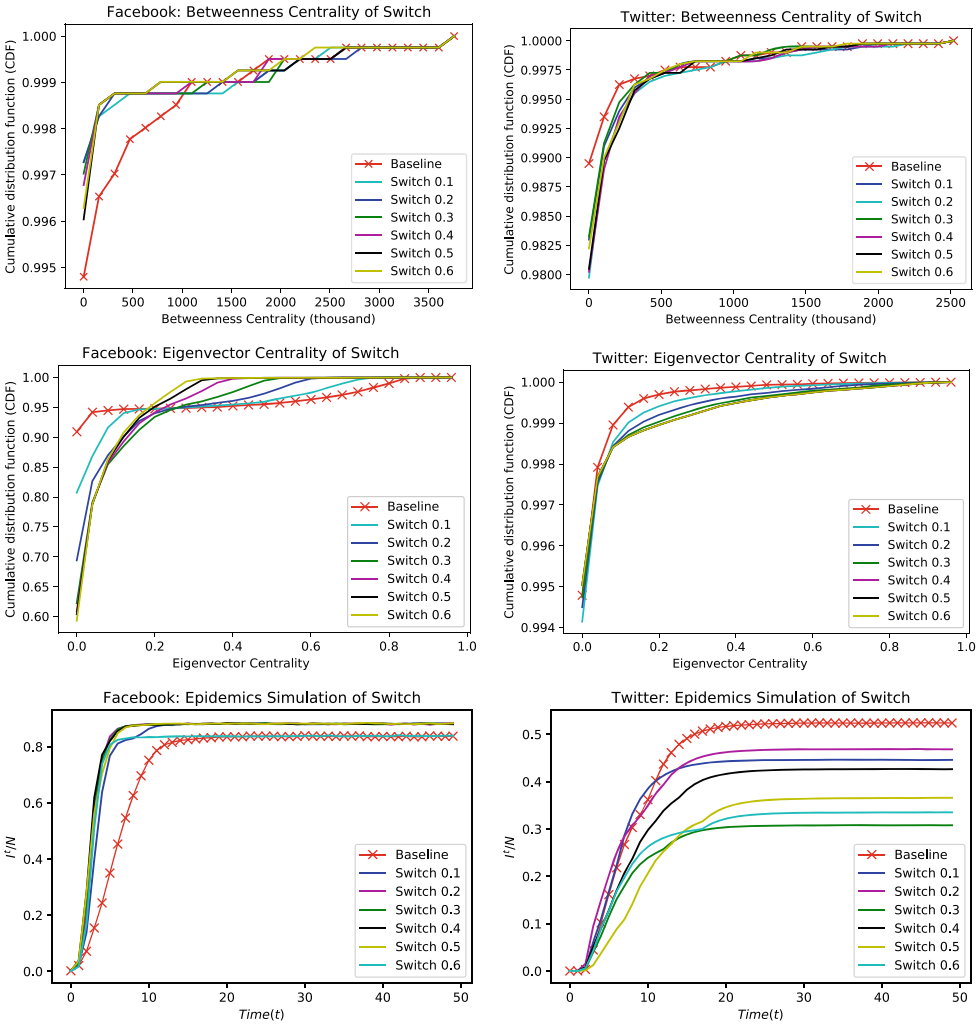


Fig. 4 Utility analysis of Random Switch: each row represents results of one utility on two datasets

Fig. 4 (continued)



better presentation. We also observe that Random Add/Del can also well-preserved EC, but has a negative impact on BC. The GT of the Facebook dataset decreases with an increasing noise ratio since Random Add/Del can destroy the triangular structures that exist in the social graph. However, the GT in the Twitter dataset increases with an increasing noise ratio. This is because in the Twitter dataset, a large number of vertices with a single degree may generate more new triangles after adding new edges to them. The SP of the Facebook dataset can be partially preserved when a small number of edges are changed, but that of the Twitter dataset is affected dramatically by these edge changes.

4.2.2 Random switch

Random switch randomly chooses two distinct existing edges and by switching their endpoints to create new edges that are not present in the original graph. As shown in Fig. 4, this algorithm can totally preserve the degree distribution of the vertices in a social graph, and conditionally preserve GT, SP, CC, and BC. Although Random Switch creates new edges between distinct vertices without changing their degrees, the internal structure is disturbed and the high-degree vertices are more likely to connect to the vertices with low degrees; this kind of changes has a great impact on EC and ES. From the perspective of privacy protection, Random Add/Del and Random Switch are vulnerable to several existing structure-based de-anonymization attacks, as reported in [16, 30, 35].

4.2.3 Clustering

The goal of Clustering anonymization is to divide vertices into clusters, and make the vertices in a cluster have similar structures to prevent a single vertex from being uniquely identified. In our simulations, the bounded t-means clustering algorithm [36] is first applied to partition vertices into groups, then within each group we add or remove edges for each vertex to match the degree of the cluster center. We also remove edges between high-degree vertices and add new ones between low-degree vertices to perturb the

inter cluster edges. From Fig. 5, one can see that with the increase of the group size, GT decreases on both datasets. Moreover, the algorithm can preserve EC well since the structures among the vertices with medium degrees are not severely disturbed. On the other hand, SP, CC, BC, and ES on both datasets are not preserved well as the structures within a cluster and among high-degree vertices are seriously distorted, even when a small cluster size is applied.

4.2.4 *K*-degree

The K-degree algorithm [3, 23] requires a vertex to have the same degree with at least $K-1$ other vertices in the social

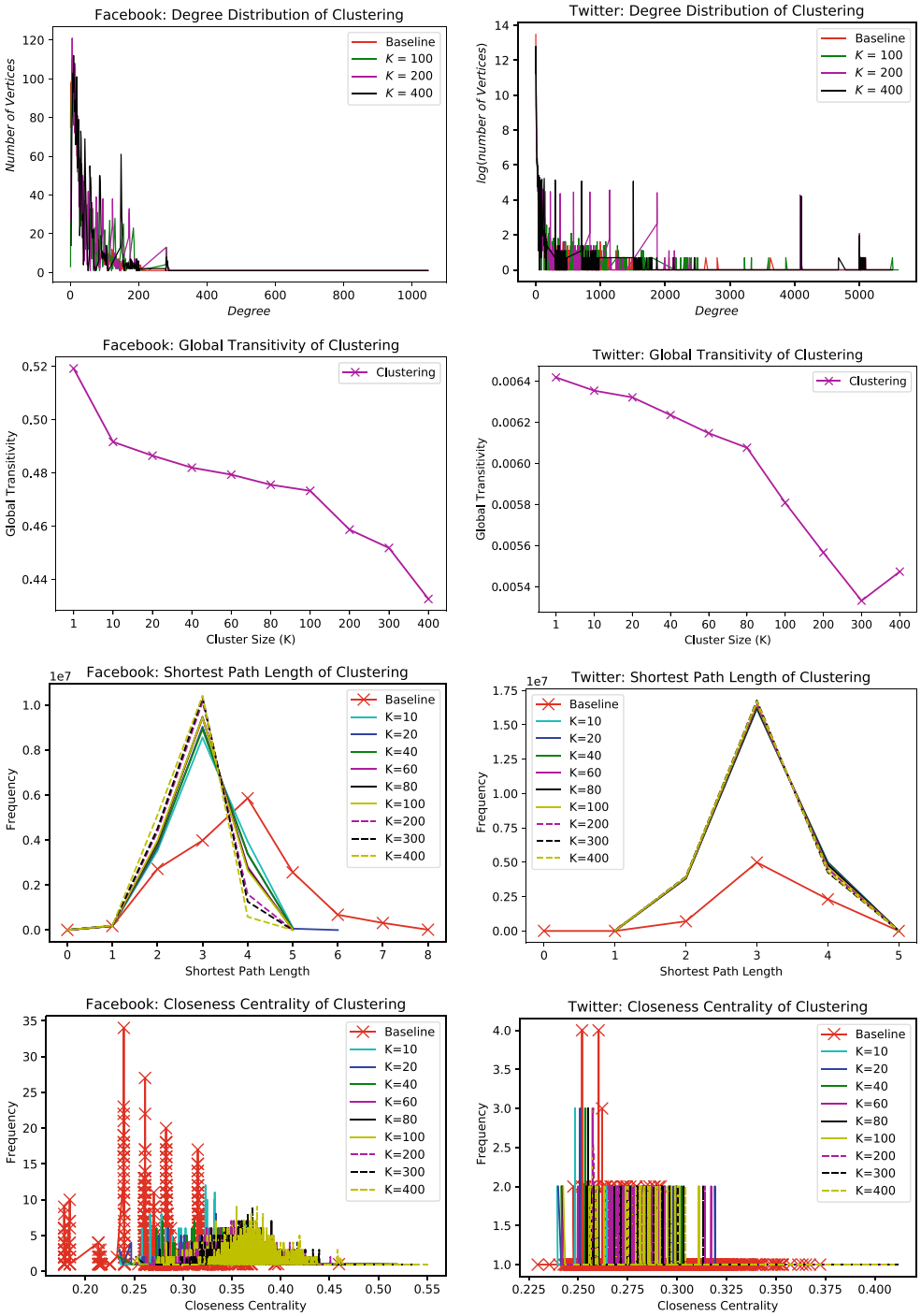
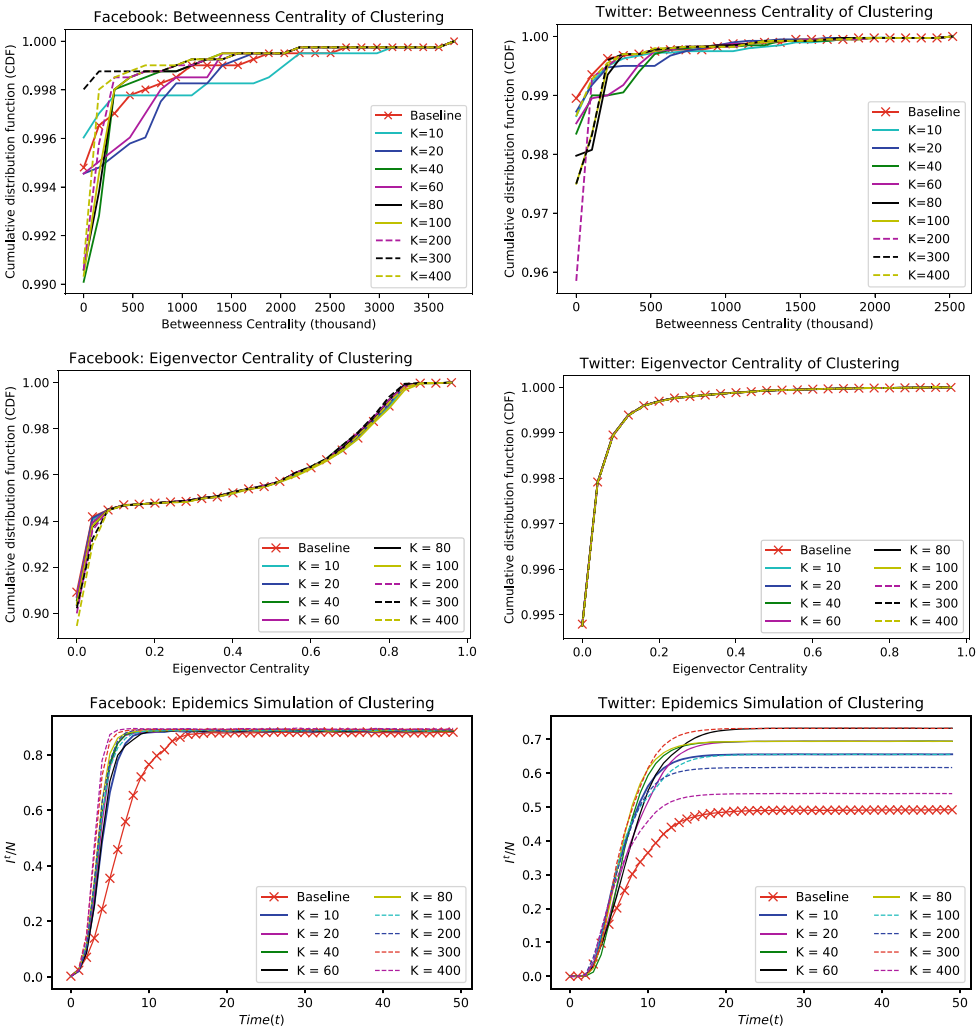


Fig. 5 Utility analysis of Clustering: each row represents results of one utility on two datasets

Fig. 5 (continued)



graph. When K is large, the K-degree algorithm needs to add or delete more edges in order to satisfy this requirement. Figure 6 presents the utility results. One can see that K-degree has the greatest impact on degree distribution with an increasing K. The GT of the Facebook dataset rises at first and then starts to decline after $K = 60$. We also observe that GT is inversely proportional to the number of triplets in the Facebook dataset. Since the K-degree anonymization algorithm reduces the number of vertices with small degrees, the number of triplets goes down, resulting in a rising global transitivity trend. However, when K becomes larger than 100, the number of vertices with larger degrees becomes substantially larger, i.e., many non-existing edges are being added into these vertices; thus, the number of triplets goes up, resulting in a declining global transitivity trend. Other graph utilities such as SP, CC, BC, and EC can be partially preserved if K is chosen to be small. Furthermore, the application utility ES is not well preserved on both datasets since the K-degree algorithm introduces a lot of noises into the anonymized graph, severely disturbing the graph structure of the anonymized graph.

4.2.5 Random Walk

Random Walk–based perturbation algorithm can protect link privacy by replacing the edge between a vertex and its neighbor with an edge between the destination of a random walk starting from the neighbor node and the vertex. As shown in Fig. 7, Random Walk can totally preserve the degree utility, but has negative effects on GT since the internal structure is affected during anonymization. Other utilities like PL, CC, EC, and ES depend on the random walk length w and the social graph structure; a small w may lose more utilities than a large w.

4.2.6 Summary

In the previous subsections, we evaluate and analyze the graph utility and application utility performance of five graph anonymization algorithms on a Facebook dataset and a Twitter dataset. On the basis of the analyses above, we notice that most anonymization algorithms can partially preserve graph utility with a small noise ratio. Some anonymization algorithms such as Random Switch and Random Walk have the ability to preserve local graph utility (e.g., vertex degree), but they may also have negative impacts on global utility (e.g., global transitivity, shortest

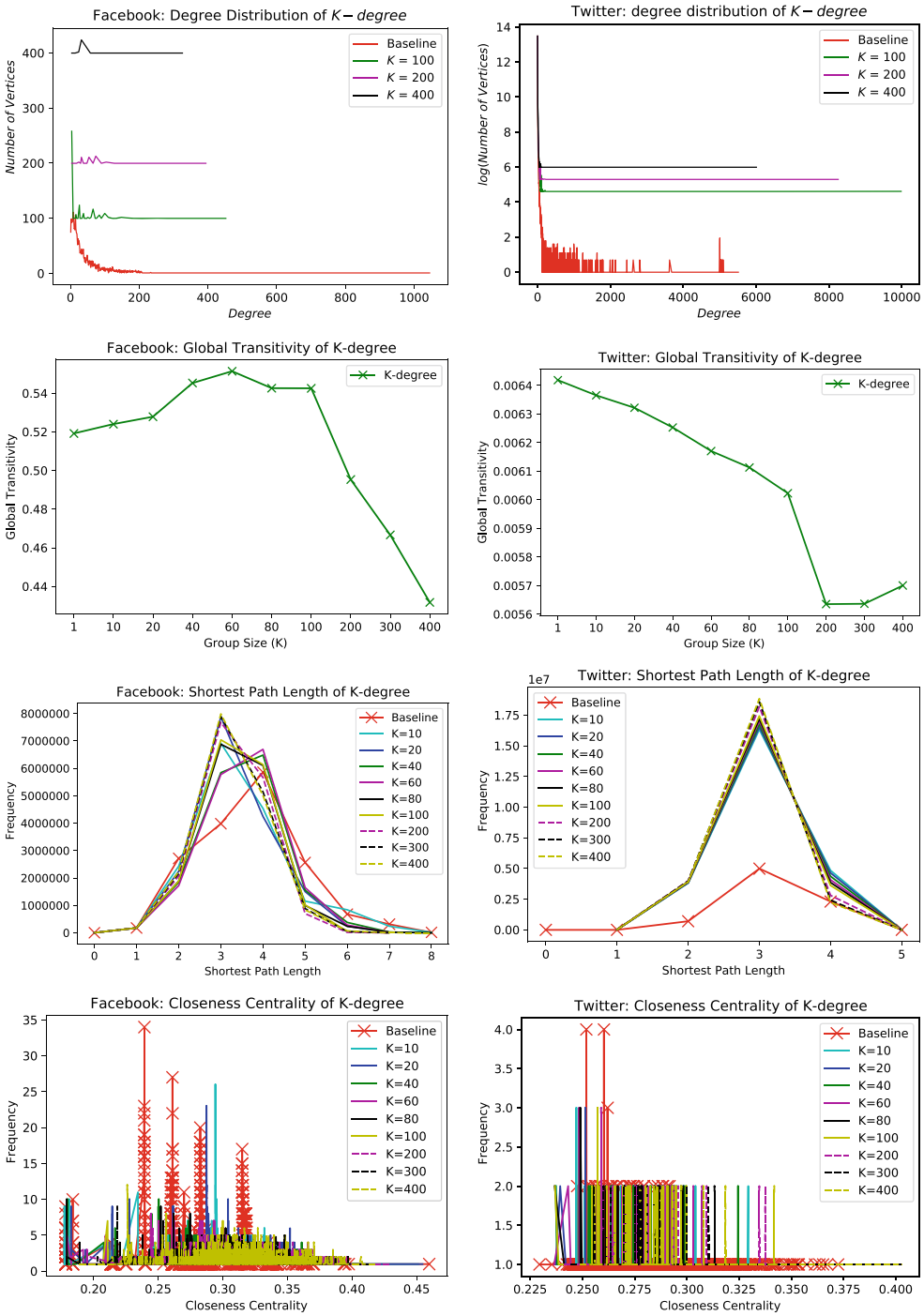
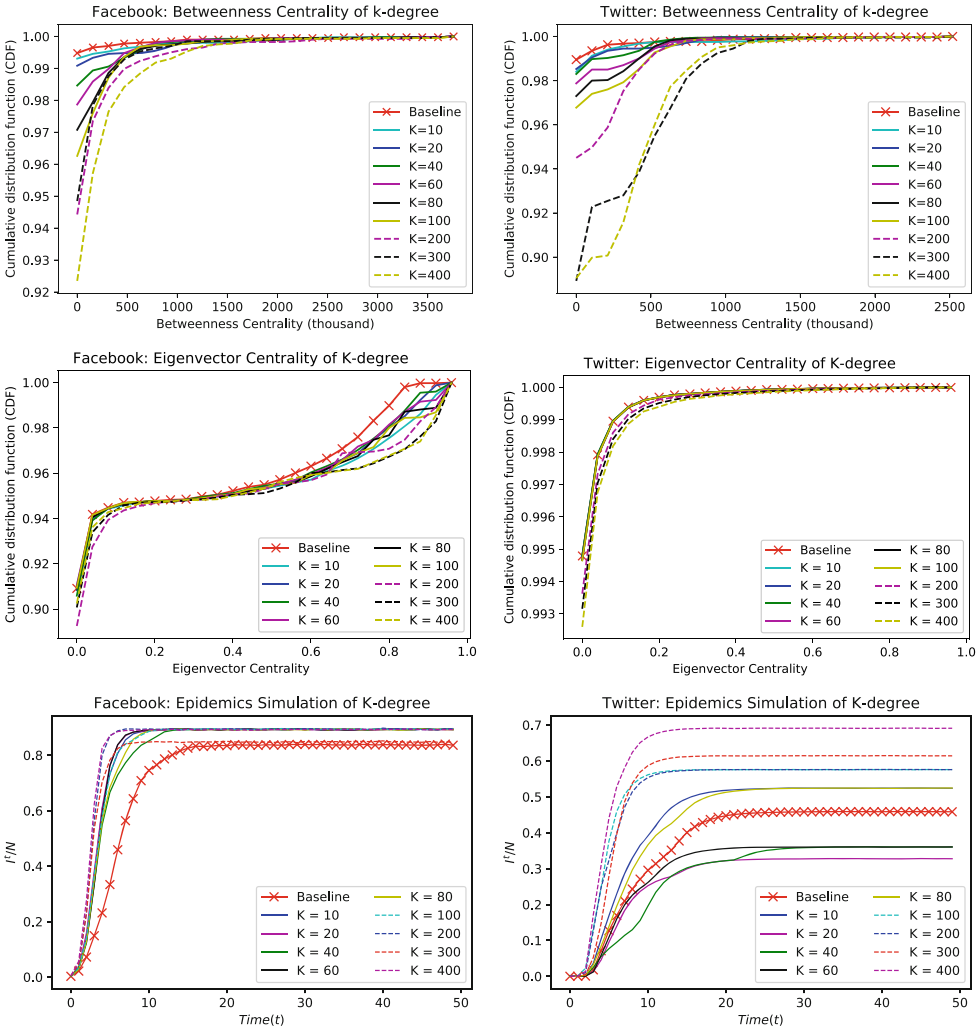


Fig. 6 Utility analysis of K-degree: each row represents results of one utility on two datasets

Fig. 6 (continued)



path length). On the other hand, one can see that the structure of a social graph is able to affect the utility preservation of an anonymized graph. From the perspective of de-anonymization, an attacker can measure the similarity of vertices in the anonymized graph and the reference graph which includes users' true identities. It can then design a multi-dimensional measurement of vertex similarity considering both local structures and global structures to improve the de-anonymization accuracy.

5 Open problems and future work

In the previous section, we present an empirical study to comprehensively analyze the utilities of the state-of-the-art structure-based anonymization techniques in online social networks and report a few interesting facts. In this section, we outline the potential challenges regarding future research directions.

5.1 The trade-off between privacy preservation and data utility

Preserving the usability of social network data while guaranteeing privacy preservation is the main objective of many social network service providers. Nevertheless, these two aspects are in fact contradictory to some extent. In order to achieve these two paradoxical objectives, we need to find an accurate trade-off, i.e., to effectively anonymize the graph data with data usability preservation to ensure that adversaries cannot manipulate the data for privacy breach and network users can still utilize the data without sacrificing quality.

On the other hand, based on the above analysis, one can see that current anonymization techniques can only make the social network vertices indistinguishable with respect to one or a few structural properties. It is difficult to make the vertices structurally indistinguishable with respect to all the structural properties of a graph. In fact, when such an ideal objective is achieved, data usability would be completely

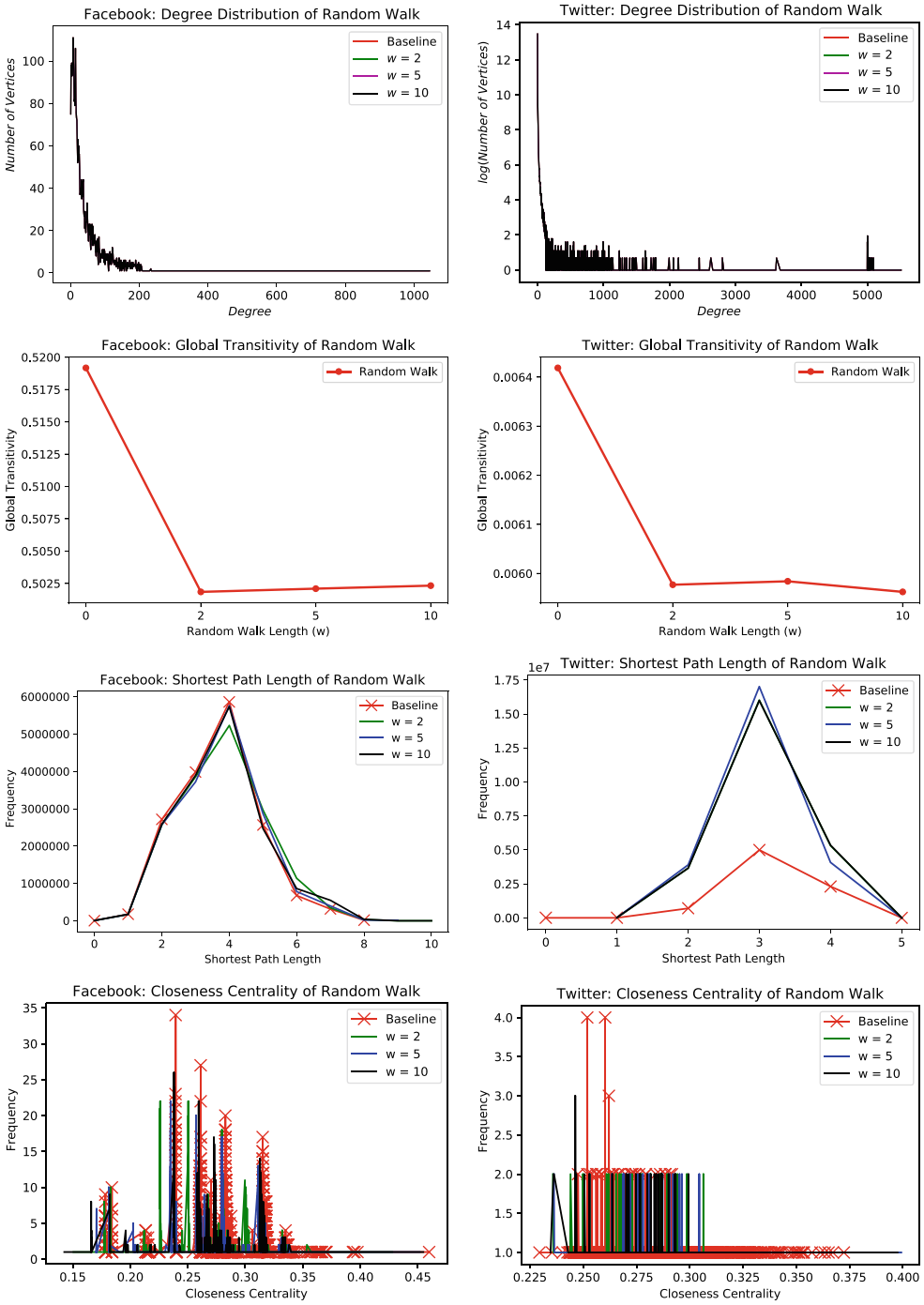


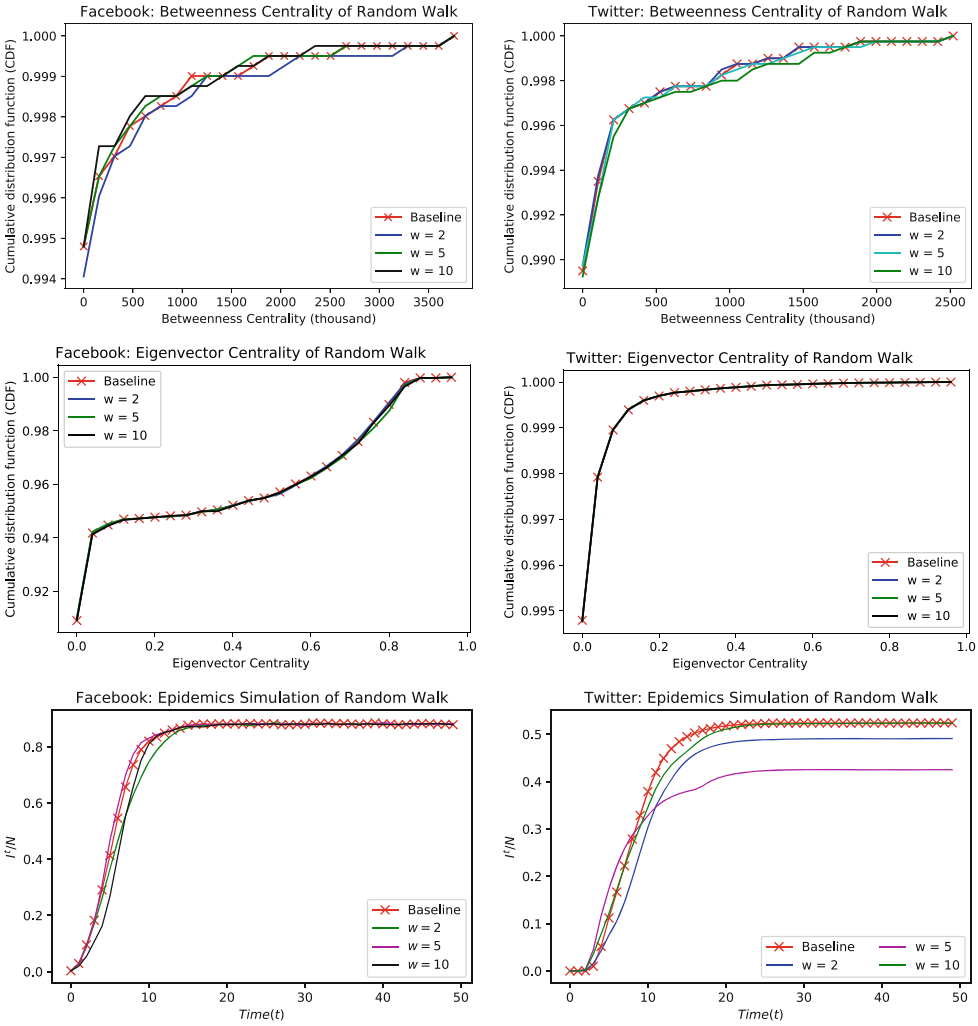
Fig. 7 Utility analysis of Random Walk: each row represents results of one utility on two datasets

destroyed so that the graph data becomes meaningless to the end users.

5.2 The efficiency and complexity of anonymization techniques

Since a social network is typically large scale with a big volume of data, some graph anonymization algorithms which are appropriate for a simple network may not be suitable. The big complexity and humongous size make the problem more challenging. Moreover, to guarantee the privacy preservation level, noises need to be added to the graph for perturbation operations. Therefore, it is possible for the network data to be useless due to the large scale of the network and the large magnitude of the added noise even for a low level of privacy preservation.

Fig. 7 (continued)



6 Conclusion

In this paper, we implement five state-of-the-art structure-based anonymization algorithms and analyze their performance in preserving the popular graph and application utilities on a Twitter and a Facebook dataset. We conclude that the structure of the datasets can significantly affect the performances of anonymization algorithms. More specifically, in our study, the Facebook dataset has a high average degree while the Twitter dataset with a large amount of single-degree vertices has a low average degree. As a result, Random Add/Del increases the shortest path length in the Twitter dataset but decreases the shortest path length in the Facebook dataset. Clustering and K-degree also perform better in the Twitter dataset than in the Facebook dataset. Random Walk performs better than the other algorithms on graph utilities. Meanwhile, the change in the number of high-degree vertices has a great impact on graph utilities and the application utility. Finally, we provide a brief overview on future research directions and summarize the challenges involved therein.

Funding information This work was partially supported by the US National Science Foundation under grants CNS-1704397, CNS-1704287, and CNS-1704274; the National Science Foundation of China under grants 61832012, 61871466, and 61771289; and the Guangxi Natural Science Foundation Innovation Research Team Project under Grant 2016GXNSFGA380002.

References

1. Bailey NT et al (1975) The mathematical theory of infectious diseases and its applications. Charles Griffin & Company Ltd, 5a Crendon Street, High Wycombe Bucks HP13 6LE
2. Bhagat S, Cormode G, Krishnamurthy B, Srivastava D (2009) Class-based graph anonymization for social network data. Proceedings of the VLDB Endowment 2(1):766–777
3. Casas-Roma J, Herrera-Joancomartí J, Torra V (2013) An algorithm for k-degree anonymity on large networks. In: Proceedings of the 2013 IEEE/ACM international conference on advances in social networks analysis and mining, ACM, pp 671–675
4. Chen S, Zhou S (2013) Recursive mechanism: towards node differential privacy and unrestricted joins. In: Proceedings of the 2013 ACM SIGMOD international conference on management of data, ACM, pp 653–664

5. Cheng J, Fu AWc, Liu J (2010) k-isomorphism: privacy preserving network publication against structural attacks. In: Proceedings of the 2010 ACM SIGMOD international conference on management of data, ACM, pp 459–470
6. Csárdi G., Nepusz T, Airoldi EM (2016) Statistical network analysis with igraph. Springer
7. Day WY, Li N, Lyu M (2016) Publishing graph degree distribution with node differential privacy. In: Proceedings of the 2016 international conference on management of data, ACM, pp 123–138
8. Dwork C (2008) Differential privacy: a survey of results. In: International conference on theory and applications of models of computation, Springer, pp 1–19
9. Dwork C, McSherry FD (2010) Differential data privacy. US Patent 7,698,250
10. Fortunato S (2010) Community detection in graphs. Phys Rep 486(3-5):75–174
11. Hay M, Li C, Miklau G, Jensen D (2009) Accurate estimation of the degree distribution of private networks. In: 2009. ICDM'09. ninth IEEE international conference on data mining, IEEE, pp 169–178
12. Hay M, Miklau G, Jensen D, Towsley D, Weis P (2008) Resisting structural re-identification in anonymized social networks. Proceedings of the VLDB Endowment 1(1):102–114
13. Hay M, Miklau G, Jensen D, Towsley D, Weis P (2008) Resisting structural re-identification in anonymized social networks. Proceedings of the VLDB Endowment 1(1):102–114
14. Heitmann B, Hermsen F, Decker S (2017) k-rdf-neighbourhood anonymity: combining structural and attribute-based anonymisation for linked data. In: Privon@ ISWC
15. Henderson K, Gallagher B, Eliassi-Rad T, Tong H, Basu S, Akoglu L, Koutra D, Faloutsos C, Li L (2012) Rolx: structural role extraction & mining in large graphs. In: Proceedings of the 18th ACM SIGKDD international conference on knowledge discovery and data mining, ACM, pp 1231–1239
16. Ji S, Li W, Srivatsa M, Beyah R (2014) Structural data de-anonymization: Quantification, practice, and implications. In: Proceedings of the 2014 ACM SIGSAC conference on computer and communications security, ACM, pp 1040–1053
17. Ji S, Mittal P, Beyah R (2016) Graph data anonymization, de-anonymization attacks, and de-anonymizability quantification: a survey. IEEE Commun Surv Tutorials 19(2):1305–1326
18. Kellaris G, Papadopoulos S (2013) Practical differential privacy via grouping and smoothing. Proceedings of the VLDB endowment 6(5):301–312
19. Kifer D, Machanavajjhala A (2011) No free lunch in data privacy. In: Proceedings of the 2011 ACM SIGMOD international conference on management of data, ACM, pp 193–204
20. Korayem M, Crandall DJ (2013) De-anonymizing users across heterogeneous social computing platforms. In: ICWSM
21. Leskovec J, Krevl A (2014) SNAP datasets: Stanford large network dataset collection http://snap.stanford.edu/data
22. Li C, Hay M, Miklau G, Wang Y (2014) A data-and workload-aware algorithm for range queries under differential privacy. Proceedings of the VLDB endowment 7(5):341–352
23. Liu K, Terzi E (2008) Towards identity anonymization on graphs. In: Proceedings of the 2008 ACM SIGMOD international conference on management of data, ACM, pp 93–106
24. Liu Y, Ji S, Mittal P (2016) Smartwalk: Enhancing social network security via adaptive random walks. In: Proceedings of the 2016 ACM SIGSAC conference on computer and communications security, ACM, pp 492–503
25. Mittal P, Papamanthou C, Song D (2012) Preserving link privacy in social network based systems. arXiv:1208.6189
26. Narayanan A, Shmatikov V (2008) Robust de-anonymization of large sparse datasets. In: 2008. SP 2008. IEEE symposium on security and privacy, IEEE, pp 111–125
27. Narayanan A, Shmatikov V (2009) De-anonymizing social networks. In: 2009 30th IEEE symposium on security and privacy, IEEE, pp 173–187
28. Newman ME (2016) Mathematics of networks. The new Palgrave dictionary of economics, pp 1–8
29. Nguyen BP, Ngo H, Kim J, Kim J (2015) Publishing graph data with subgraph differential privacy. In: International workshop on information security applications, Springer, pp 134–145
30. Nilizadeh S, Kapadia A, Ahn YY (2014) Community-enhanced de-anonymization of online social networks. In: Proceedings of the 2014 ACM SIGSAC conference on computer and communications security, ACM, pp 537–548
31. Nissim K, Raskhodnikova S, Smith A (2007) Smooth sensitivity and sampling in private data analysis. In: Proceedings of the thirty-ninth annual ACM symposium on theory of computing, ACM, pp 75–84
32. Qian J, Li XY, Zhang C, Chen L, Jung T, Han J. (2017) Social network de-anonymization and privacy inference with knowledge graph model. IEEE Transactions on Dependable and Secure Computing
33. Rong H, Ma T, Tang M, Cao J (2018) A novel subgraph k^+-isomorphism method in social network based on graph similarity detection. Soft Comput 22(8):2583–2601
34. Samarati P, Sweeney L (1998) Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression, Tech. rep., technical report, SRI International
35. Srivatsa M, Hicks M (2012) Deanonymizing mobility traces: using social network as a side-channel. In: Proceedings of the 2012 ACM conference on computer and communications security, ACM, pp 628–637
36. Thompson B, Yao D (2009) The union-split algorithm and cluster-based anonymization of social networks. In: Proceedings of the 4th international symposium on information, computer, and communications security, ACM, pp 218–227
37. Tian W, Mao J, Jiang J, He Z, Zhou Z, Liu J (2018) Deeply understanding structure-based social network de-anonymization. Prog Comput Sci 129:52–58
38. Wang B, Jia J, Zhang L, Gong NZ (2018) Structure-based sybil detection in social networks via local rule-based propagation. IEEE Transactions on Network Science and Engineering
39. Wang Q, Zhang Y, Lu X, Wang Z, Qin Z, Ren K (2018) Real-time and spatio-temporal crowd-sourced social network data publishing with differential privacy. IEEE Trans Dependable Secure Comput 15(4):591–606
40. Wang Y, Wu X (2013) Preserving differential privacy in degree-correlation based graph generation. Transactions on Data Privacy 6(2):127
41. Wu X, Hu Z, Fu X, Fu L, Wang X, Lu S (2018) Social network de-anonymization with overlapping communities: Analysis, algorithm and experiments. In: Proc IEEE INFOCOM
42. Ying X, Wu X (2008) Randomizing social networks: a spectrum preserving approach. In: Proceedings of the 2008 SIAM international conference on data mining, SIAM, pp 739–750
43. Zhang Z, Wang H, Wang C, Fang H (2015) Modeling epidemics spreading on social contact networks. IEEE Trans Emerg Top Comput 3(3):410–419
44. Zhou B, Pei J (2008) Preserving privacy in social networks against neighborhood attacks. In: 2008. ICDE 2008. IEEE 24th international conference on data engineering, IEEE, pp 506–515
45. Zou L, Chen L, Özsu MT (2009) K-automorphism: a general framework for privacy preserving network publication. Proceedings of the VLDB Endowment 2(1):946–957

Affiliations

Cheng Zhang[1] · Honglu Jiang[1] · Xiuzhen Cheng[1] · Feng Zhao[2] · Zhipeng Cai[3] · Zhi Tian[4]

Cheng Zhang
zhangchengcarl@gwu.edu

Honglu Jiang
hljiang0720@gwu.edu

Xiuzhen Cheng
cheng@gwu.edu

Zhipeng Cai
zcai@gsu.edu

Zhi Tian
ztian1@gmu.edu

1 Department of Computer Science, The George Washington University, Washington, DC, USA

2 Guangxi Colleges and Universities Key Laboratory of Complex System Optimization and Big Data Processing, Yulin Normal University, Yuzhou Qu, China

3 Department of Computer Science, Georgia State University, Atlanta, GA, USA

4 Department of Electrical and Computer Engineering, George Mason University, Fairfax, VA, USA