Graph Data Anonymization, De-anonymization Attacks, and De-anonymizability Quantification: A Survey

Shouling Ji, *Member, IEEE,* Prateek Mittal, *Member, IEEE,* and Raheem Beyah, *Senior Member, IEEE*

***Abstract*—owadays, many computer and communication systems generate graph data. Graph data span many different domains, ranging from online social network data from networks like Facebook to epidemiological data used to study the spread of infectious diseases.owadays, many computer and communication systems generate graph data. Graph data span many different domains, ranging from online social network data from networks like Facebook to epidemiological data used to study the spread of infectious diseases.N Graph data are shared regularly for many purposes including academic research and for business collaborations. Since graph data may be sensitive, data owners often use various anonymization techniques that often compromise the resulting utility of the anonymized data. To make matters worse, there are several state-of-the-art graph data de-anonymization attacks that have proven successful in recent years.**

In this paper, we survey the graph data anonymization, de-anonymization, and de-anonymizability quantification techniques in the past decade. Specifically, we systematically classify, summarize, and analyze state-of-the-art graph data anonymization, de-anonymization, and de-anonymizability quantification techniques. For existing graph data anonymization techniques, we classify them into six categories and analyze their utility performance with respect to 15 fundamental graph utility metrics and 7 high-level application utility metrics. For existing de-anonymization attacks, we classify them into two categories and examine their performance with respect to scalability, practicability, robustness, etc. We also analyze the resistance of existing graph anonymization techniques against existing graph de-anonymizaiton attacks. For existing de-anonymizability quantifications, we classify them according to whether they consider seed information or not, and analyze them in terms of their soundness. Our analysis demonstrates that (i) most anonymization schemes can partially or conditionally preserve most graph utility while losing some application utility; and (ii) state-of-the-art anonymization schemes are vulnerable to several or all of the emerging structure-based de-anonymization attacks. The actual vulnerability of each anonymization algorithm depends on how much and which data utility it preserves. Based on our summarization and analysis, we discuss the research evolution, future directions, and challenges in the research area of graph data anonymization, de-anonymization, and de-anonymizability quantification.

***Index Terms*—Graph data, social networks, anonymization, de-anonymization, quantification.**

S. Ji is with the College of Computer Science and Technology, Zhejiang University, Hangzhou, Zhejiang 310027, China and is with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332, USA.
E-mail: sji@zju.edu.cn, sji@gatech.edu

R. Beyah is with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332, USA.
E-mail: rbeyah@ece.gatech.edu

P. Mittal is with the Department of Electrical Engineering, Princeton University, Princeton, NJ 08540, USA.
E-mail: pmittal@princeton.edu

I. Introduction

MANY computing and communication system generated data have a graph structure, e.g., social networks, collaboration networks, email networks, autonomous networks, and P2P networks [1]–[8]. Even mobility traces, e.g., WiFi traces, Bluetooth traces, instant message traces, and check-ins, can be modeled by graphs by applying some sophisticated techniques [3], [4], [6]. Generally, these data can be called *graph data*. For research purposes, data and network mining tasks, and commercial applications, these graph data are often transferred, shared, or provided to the public, research community, and/or commercial partners. Therefore, it is critical to protect users' privacy during the data transfer, sharing, and/or publishing.

To protect graph users' privacy, several anonymization techniques have been proposed to anonymize graph data, which can be classified into six categorizes: *Naive ID Removal*, *Edge Editing (EE) based techniques* [9], *k-anonymity based techniques* [10]–[14] [1], *Aggregation/Class/Cluster based techniques* [15]–[17], *Differential Privacy (DP) based techniques* [18]–[22], and *Random Walk (RW) based schemes* [23] [24]. Basically, these anonymization techniques try to perturb the original graph structure to protect users' privacy while preserving as much data utility as possible.

On the other hand, following Narayanan and Shmatikov's seminal work [1], many new structure-based de-anonymization attacks to graph data have been proposed, which can be categorized into two classes: *seed-based attacks*, e.g., Narayanan et al. attack [1], Srivatsa-Hicks attacks [3], Yartseva-Grossglauser attack [25], Ji et al. attacks [26] [27], and Korula-Lattanzi attack [28], and *seed-free attacks*, e.g., Pedrsani et al. attack [29] and Ji et al. attack [4] [5]. The main idea of structure-based de-anonymization attacks is to de-anonymize anonymized users in terms of their uniquely distinguishable structural characteristics.

Furthermore, recently, the issue of graph data de-anonymizability quantification has also garnered significant research interests [4], [6], [25], [28], [30], [31], where researchers study why graph data can be de-anonymized, what are the de-anonymization conditions, and how many users are de-anonymizable.

[1]Here, k is an anonymization parameter, which is usually an integer.

Contributions. To summarize the research advances in the graph data anonymization, de-anonymization, and de-anonymizability quantification area, enhance the understanding of the rapid research progress, provide a comprehensive research picture, and facilitate the future research, in this paper, we systematically summarize, classify, and analyze state-of-the-art graph data anonymization techniques, de-anonymization attacks, and de-anonymizability quantification solutions. Specifically, our main contributions are as follows.

- We systematically classify and analyze state-of-the-art graph data anonymization schemes and their performance with respect to 22 graph and application utilities (including joint degree, local and global clustering coefficient, network resilience, infectiousness, role extraction, reliable email, community structure, etc.). The evaluation results demonstrate that most existing anonymization algorithms can partially or conditionally preserve most graph utility. However, all the anonymization schemes lose one or more application utility.
- We summarize and analyze the fundamental properties of existing structure-based de-anonymization attacks, e.g., the algorithms' scalability, practicability, and robustness. Our results show that modern structure-based de-anonymization attacks are powerful and robust. Then, we analytically evaluate the performance of state-of-the-art graph data anonymization schemes on defending against modern structure-based de-anonymization attacks. Existing anonymization techniques, e.g., k-anonymity based schemes and DP based schemes, are vulnerable to modern structure-based de-anonymization attacks. Their actual vulnerability depends on how much data utility is preserved in the anonymized data. Therefore, it is still an open yet serious problem to develop effective graph data anonymization techniques.
- We summarize and classify existing de-anonymizability quantification techniques. We also comprehensively analyze existing quantification techniques with respect error tolerance, practicality, and generality.
- We analyze the technique evolution and provide future research directions for graph data anonymization, de-anonymization, and de-anonymizability quantification. We also discuss the potential challenges of each future research direction.

The rest of this paper is organized as follows. We give a brief overview of graph data and its sharing and availability in Section II. In Section III, we summarize, classify, and analyze state-of-the-art graph data anonymization schemes as well as their utility performance. In Section IV, we summarize, classify, and analyze structure-based de-anonymization attacks, and analyze the effectiveness of existing anonymization schemes against modern de-anonymization attacks. We survey and analyze existing de-anonymizability quantification techniques in Section V. The evolution and future research directions of graph data anonymization, de-anonymization, and de-anonymizability quantification are discussed in Section VI. We discuss the related work in Section VII and conclude this paper in Section VIII.

II. Graph Data: Source and Application

A. Overview of Graph Data

Nowadays, many computer and communication systems generate graph data [9]–[24], [32], [33]. Below, we summarize representative computer-generated graph data.

- *Social Networks* [34]–[45]. It is natural to represent social networks (e.g., Facebook [34], Google+ [35], Twitter [36], LinkedIn [37], YouTube [38], LiveJournal [39], Orkut [40], Slashdot [41], and Pokec [42]) as graphs, where nodes denote users and links/edges denote the social relationships (friendship, circle-relationship, follow relationship, etc.) among users;
- *Communication Data* [3], [46]–[55]. Another typical category of graph data are the communication data, including phone-call networks [46]–[49], email networks [50]–[52], wiki-Talk networks [53], [54], instance message network [3], etc. To represent communication networks, users are modeled by nodes and the communication relationship (phone calls, emails, talks, etc.) are modeled by links among nodes;
- *Mobility Traces* [3], [56]. Mobility traces, e.g., WiFi traces [3], Bluetooth traces [3], check-ins [56], usually consist of records of format (*user ID, latitude, longitude, timestamp, location ID*). They can be transferred to user-connect graphs by applying sophisticated data processing techniques (e.g., entropy-based techniques) [3], [56], where nodes represent users and links/edges indicate the co-appearance or connection relation;
- *Epidemiological and Health-care Data* [57]–[59]. Many health-care data are in the graph form, leveraging which health-care professionals can study disease propagation as well as other social health problems [57]–[59]. For instance, to study the sexual contact-based disease transmission, an adolescent romantic and sexual network is published in [59], which consists of a population of over 800 adolescents residing in a mid-sized town in the mid-western United States.
- *Collaboration Networks* [50], [60]–[62]. The collaboration networks, e.g., Arxiv [50], [63], the computer science collaboration network DBLP and ArnetMiner [60], [61], represent the collaboration relationships among researchers. Straightforwardly, collaboration networks can be modeled by graphs where nodes represent researchers and links represent collaborations.
- *Citation Networks* [64], [65]. The citation networks carry the citation information among research papers, which are naturally graph data.
- *Web Graphs* [51], [66], [67]. Web graphs indicated the link information among web pages, where nodes represent web pages and edges represent hyperlinks among them.
- *Internet Peer-to-Peer Networks and Other Network Topologies* [50], [68]. Peer-to-Peer networks and other network topologies can be modeled by graph data, where nodes represent network terminals in the network and edges represent the connections among them.

- *Autonomous System Graphs* [64]. The graph of routers comprising the Internet can be organized into subgraphs called Autonomous Systems (AS) [64]. Each AS exchanges traffic flows with some neighbors (peers). Therefore, graphs can be constructed to represent *who-talks-to-whom* relationships among AS.

B. Graph Data Sharing and Availability

Graph data publishing/sharing/transferring has important implications for research, government, commercial, and civil applications. Below, we discuss some typical graph data publishing/sharing/transferring scenarios.

1) Academic Research: As it has been well known, real-world data publishing/sharing/transferring is the most valuable resource for academic research, e.g., personalized advertising, sense-making, influence maximization, innovation/disease diffusion, similar users searching, user classification, reliable email, secure routing, and Sybil detection [1]–[4], [6], [9]–[30], [32], [33], [69]–[78].

During the annually KDD Cup events[2], several datasets (including graph datasets) are published for data mining and knowledge discovery tasks [69]. For instance, several network topological structure datasets, social network datasets, customer relationship graphs are published or shared with researchers. Similarly, many other academic events/instituions regularly provide graph data to the research community [1], [70]–[73], [75], [79]. Recently, Twitter introduced its data sharing project to the academia with the following statement [70]:

> *Today we're introducing a pilot project we're calling Twitter Data Grants, through which we'll give a handful of research institutions access to our public and historical data.*
> ...
> *Our Data Grants program aims to change that by connecting research institutions and academics with the data they need.*

In order to promote and prosper real-world data driven research, many other real-world graph data are shared with researchers, e.g., Facebook data [71], [73], QQ data [72], Microblog data [73], Citation network data [73].

2) Government Data Mining Tasks: In addition to serve academia research, graph data are frequently shared/transferred for government data mining tasks. For instance, customer understanding and international fraud detection can be achieved by leveraging the structure and pattern analysis of phone-call networks [80]. Furthermore, communication data (e.g., phone-call networks, email networks) can also be applied to serious national security data mining tasks, such as graph theory-based terrorist analysis [81]. Recently, it has been shown a lot that government agencies employ graph data for several kinds of data mining tasks [82]:

> *In the name of fighting terrorism, the US government has been mining data collected from phone companies such as Verizon for the past seven years and from Google, Facebook, and other social media firms for at least four years, according to government documents leaked this week to news organizations.*
> ...

In addition, some companies have been reported to sell graph data-based data mining solutions to governments.

3) Business Applications: Data sharing/transferring is one of the common business mode of nowadays companies. For instance, Google, Facebook, and QQ share their data with business partners for personalized precision advertising and targeted advising, under which cost savings and maximized advertising effectiveness can be achieved. In addition to advertising, graph data are also shared among companies to build enterprise applications to improve business decisions [83]:

> *Twitter and IBM announced a significant partnership today that will involve Twitter sharing its data with IBM for integration into IBM's enterprise solutions, including the Watson cloud platform.* ...

To prove that companies do share data, we examine the privacy policies of some companies as follows.

> *According to google, information used during registration, used while using the services are collected by google and might be given to trusted parties for processing* ...
>
> –Google Plus Privacy Policy [84].
>
> ... *With the information Facebook have, they can share the payment information to complete a purchase, send email to invite others on behalf on the user, share information with marketers, help others to find you, and give search engines access to your public information.*
>
> –Facebook Privacy Policy [85].
>
> ... *If making purchases on Twitter, Twitter can share information such as address and name. In addition user information might be sold in case that Twitter is in a part of a buy out or merger or if the information is need for legal reasons. Lastly public information can be shared to others for reasons such as advertisers whose link a user clicked on.*
>
> –Twitter Privacy Policy [86].

From the above policies, users data (graph data) are explicitly been claimed to be shared among partners.

4) Civil Applications: Graph data are also published/shared/transferred for many civil applications. A typical such scenario is to analyze the propagation of infectious diseases, e.g., the flu, HIV, Ebola [87]–[89]. Real-world graph data are valuable to accurate disease propagation analysis. As shown in [59], when analyzing sexual contact-based disease diffusion, real sexual networks-based analysis is very different from that leveraging simulated or randomly generated graph data. Recently, when studying the Ebola Outbreak 2014, the Ebola Hemoragic Fever propagation in a modern city is modeled and analyzed based on the social graphs and other data [89].

5) Other Scenarios: Graph data are widely available in many other scenarios or through multiple means.

[2]KDD is the abbreviation for "ACM SIGKDD Conference on Knowledge Discovery and Data Mining".

- For conducting research, developing web and mobile applications, designing data visualizations, and other applications, government agencies regularly release data by law [90].
- Many graph data can be crawled employing an API or screen-scraping, e.g., Google+ [91], Facebook [71] [62], Twitter [62], [92], YouTube [62], [92], [93], LinkedIn [94].
- Graph data are widely available on many data sharing websites [62], [92], [95]–[97]. For instance, many social network data, communication network data, mobility traces, collaboration data, review data, autonomous system graphs are available at Stanford SNAP [62], ASU Network Data Repository [92], Dartmouth CRAWDAD [95], UCI Network Data Repository [96], CMU Datasets [97], etc.
- Recently, with the emergence of *data brokers*, many graph data, especially the sensitive data such as medical records, financial information, credit reports, social relations, and other personal profiles, are easily obtained [98]–[100].

  . . . . . .

  Consumer data companies are scooping up huge amounts of consumer information about people around the world and selling it, providing marketers details about whether you're pregnant or divorced or trying to lose weight, about how rich you are and what kinds of cars you drive. But many people still don't know data brokers exist.

  . . . . . .

  As we highlighted last year, some data companies recorded then resell all kinds of information you post online, including your screen names, website addresses, interests, hometown and professional history, and how many friends or followers you have [99].

C. Summary

In this section, we discuss the sources, availability, and application of graph data. From our discussion, graph data are ubiquitous now and can be generated by many computing and communication systems. Since they are valuable and crucial for academia research, government data mining tasks, business and civil applications, and many other applications, graph data are widely published, shared, and transferred. On the other hand, a lot of user and/or system private information that is embedded in the graph data suffers from violation and being leaked during the data publishing, sharing, and transferring. Therefore, understanding the privacy leakage risk, potential de-anonymization attacks, and possible anonymization techniques of graph data is important for secure data publishing, sharing, and transferring.

In the following sections, we systematically survey, evaluate, and analyze the 15 years' advances of graph data anonymizaiton, de-anonymization, and de-anonymizability quantification research. To improve the paper readability, we summarize the abbreviations used in this paper in Table I.

TABLE I
ABBREVIATIONS.

Abbreviations	Full name
EE	Edge Editing
DP	Differential Privacy
RW	Random Walk
k-NA	k-Neighborhood Anonymity
k-DA	k-Degree Anonymity
k-auto	k-automorphism
k-iso	k-isomorphism
Deg.	Degree
JD	Joint Degree
ED	Effective Diameter
PL	Path Length
LCC	Local Clustering Coefficient
GCC	Global Clustering Coefficient
CC	Closeness Centrality
BC	Betweenness Centrality
EV	EigenVector
NC	Network Constraint
NR	Network Resilience
Infe.	Infectiousness
PR	PageRank
HS	Hub Score
AS	Authority Score
RX	Role eXtraction
RE	Reliable Email
IM	Influence Maximization
MINS	Minimum-sized Influential Node Set (MINS)
CD	Community Detection
SR	Secure Routing
SD	Sybil Detection
DV	Distance Vector
RST	Randomized Spanning Tree
RSM	Recursive Subgraph Matching
DA	De-Anonymization
ADA	Adaptive De-Anonymization
ER	Erdös-Rényi
PA	Preferential Attachment

III. Graph Data Anonymization and Utility Analysis

In this section, we summarize and classify existing anonymization techniques. Subsequently, we present the fundamental graph utility metrics as well as popular application utility metrics. Third, we analytically study the utility performance of existing graph data anonymization techniques. The performance of existing anonymization techniques against de-anonymization attacks is studied in Section IV-D. Note that, we mainly focus on analyzing anonymization-based privacy protection techniques in this paper. We do not consider the privacy-preserving techniques on encrypted data, e.g., privacy-preserving cloud computing [101]–[105] and privacy-preserving image data process [106], [107].

Although we study graph data anonymization and de-anonymization in this paper, for completeness and to demonstrate the evolution of anonymization techniques, we also briefly summarize the anonymization schemes for non-graph data, e.g., micro/tabular data, set-valued data.

A. Micro/Tabular Data Anonymization

1) k-anonymity and Variants: Security/privacy is an important concern when publishing, transferring, and/or sharing data. To protect data's security and privacy, dozens of

techniques have been proposed. Among them, k-*anonymity*, defined by Samarati and Sweeney [108], [109], opened a prosperous research area of data anonymization. Under k-anonymity, one user's data cannot be distinguished from at least $k-1$ other users' data in the publishing data. In general, to achieve k-anonymity is NP-hard. Therefore, many following works focus on designing efficient k-anonymization algorithms and/or extending k-anonymity to more effective *privacy models* (e.g., ℓ-*diversity* [110], t-*closeness* [111]) for specific data publishing applications.

Following [108], [109], LeFevre et al. provided a practical framework for efficient full-domain k-anonymity [112]. To improve the k-anonymity performance, Aggarwal et al. designed a $O(k)$-approximation algorithm [63] followed by Park and Shim who further improved the approximation ratio to $O(\log k)$ [113].

To better protect users' privacy, dozens of improved privacy models of k-anonymity have been proposed. Considering that the k-anonymity property protects against identity disclosure while failing to protect against attribute disclosure, Truta and Vinay proposed the p-sensitive k-anonymity property [114]. To defend against the *homogeneity attack* and *background knowledge attack* to k-anonymity, Machanavajjhala et al. proposed ℓ-*diversity* in [110], under which each equivalence class has at least ℓ well-represented values for each sensitive attribute. For protecting both identification information and sensitive relationship information in a dataset, Wong et al. extended k-anonymity to (α, k)-*anonymity* [115]. Since privacy disclosure may also happen under ℓ-diversity based on the attribute distribution, Li et al. proposed t-*closeness* in [111], which requires that the distribution of a sensitive attribute in any equivalence class should be close to the attribute distribution in the overall dataset. Similar to ℓ-diversity, to defend against the background knowledge attack on k-anonymity, Martin et al. proposed (c, k)-*safety*, where k characterizes the background knowledge and c indicates the desired privacy level [116]. To improve the accuracy of generalization based k-anonymity/ℓ-diversity, *permutation-based anonymization* was designed in [117] by Xiao and Tao and [118] by Zhang et al.

In [119], Wang and Fung proposed (X, Y)-*privacy* (including (X, Y)-*anonymity* and (X, Y)-*linkability*) to protect the privacy of sequential data releases, where X and Y are two attribute sets over the join of two sequential datasets. To address the inappropriateness of k-anonymity/ℓ-diversity in some situations, Nergiz et al. presented δ-*presence* under which an adversary cannot identify any individual as being in a dataset with certainty greater than δ [120]. To address the privacy leakage of dynamic datasets, Xiao and Tao proposed a new privacy model named m-*invariance*, where m measures the number of different users and sensitive values of each quasi-identification group [121], [122]. Considering the specific features of healthcare data, Mohammed proposed LKC-*privacy*, where L characterizes the adversary's power, and K and C measure the privacy thresholds of identity and attribute linkage, respectively [123].

Considering that many privacy models (e.g., t-closeness) require that groups of sensitive attributes follow specified distributions, Koudas et al. proposed P-*private generation*, under which a group of sensitive attribute values can be transformed to a certain target distribution P with minimal data distortion [124]. To defend the *structure-based attack* and *label-based attack* to recommendation data, Chang et al. extended k-anonymity to a *predictive anonymization* model, where privacy, utility, and performance are considered simultaneously [125]. In [126], [127], Aggarwal et al. and Mahmood et al. generalized k-anonymity to k-*Anonymous Cluster* (k-AC), which allows more information being published without compromising privacy. Considering different personal levels of desired privacy, Choromanski relaxed k-anonymity to b-*matching* from adaptive anonymity (b is short for *bipartite graph*) [128].

k-anonymity + Utility. To make the anonymized data useful, utility-based anonymization techniques are also extensively studied. In [129], LeFevre extended k-anonymity and ℓ-diversity to *workload-aware anonymization*. In [130], Xu et al. designed two heuristic local recordings for utility-based anonymization. Similarly, Kifer and Gehrke investigated utility preserved anonymization schemes which maintain the same privacy guarantees of k-anonymity and ℓ-diversity [131]. In [132], Brickell and Shmatikov evaluated the tradeoff between privacy and utility. Their results demonstrated that even modest privacy gains require almost complete destruction of the data mining utility.

2) Differential Privacy: Besides k-anonymity and its variants, *Differential Privacy* (DP), introduced by Dwork [133], [134], is another popular anonymization technique to provide a provable strong privacy guarantee. Initially, DP is designed for statistical databases aiming at maximizing the accuracy of queries while minimizing the chance of privacy leakage [133]. Following [133], many enhanced DP techniques have been proposed for different application scenarios.

In [135], Hay et al. proposed an approach to improve the accuracy of differentially private algorithms for both unattributed and universal histograms. In [136], Mohammed studied how to guarantee ϵ-DP under the *non-interactive* setting by probabilistically generalizing the raw data and then adding noise. To achieve ϵ-DP and meanwhile improve data's utility, Kellaris and Papadopoulos proposed a practical DP framework via *grouping* and *smoothing* [137]. To improve the accuracy of queries, Li et al. presented a two-stage , data and workload aware mechanism for answering sets of range queries under DP [138]. In [139], Qardaji et al. considered the scenario of differentially private releasing of *marginal contingency tables*. They introduced PriView, which computes marginal tables for a number of sets of attributes, and then reconstruct any designed k-way marginal based on these sets of attributes.

Similar to k-anonymity, many variants of ϵ-DP have been designed to better meet the privacy requirements of specific applications. In [134], Dwork et al. proposed a relaxed version of ϵ-DP, named (ϵ, δ)-*DP*, that permits both an additive term (quantified by δ) and a multiplicative term (indicated by ϵ). In [140], McSherry and Mironov applied (ϵ, δ)-DP to differentially private recommender systems. They designed and analyzed a recommender system built to provide modern privacy guarantees. In [141], Lee and Clifton et al. present-

ed an alternative of ϵ-DP called *ρ-Differential Identifiability* (ρ-DI), which provides the same guarantees as DP while bounds the probability of individual identification by ρ. Li et al. proposed a general privacy model $(\mathbb{D}, \gamma)$-*membership privacy*, where $\mathbb{D}$ captures all states of prior knowledge of an adversary and γ limits the increase in confidence of accurate membership assertion [142]. In [143], Li et al. studied the correlation between k-anonymity and DP. They demonstrated that k-anonymization, when done "safely" and preceded with a random sampling step, meets (ϵ, δ)-DP with reasonable parameters.

B. Set-valued Data Anonymization

Different from traditional micro/tabular data, *set-valued data*, e.g., transaction data, web search queries, click streams, and transit data, refer to the data in which each record owner is associated with a set of items [144]–[151]. In [144], He and Naughton extended k-anonymity to anonymize set-valued data through top-down and local generalization. Similarly, Xue et al. generalized k-anonymity and ℓ-diversity to set-valued data by nonreciprocal recording [145]. In [146], Terrovitis et al. proposed *k^m-anonymization*, which prevents an adversary from distinguishing a transaction from k transactions given him the knowledge of at most m items. In [147], Xu et al. proposed (h, k, p)-*coherence* for anonymizing transaction databases, which ensures that for an adversary of power p, the probability of identifying a transaction is limited to $1/k$ and the probability of linking an individual to a private item is limited to h. Another anonymization model is *ρ-uncertainty*, proposed by Cao et al. [148], which defends against sensitive associations without constraining the nature of an adversary's knowledge or falsifying data. Similar to for tabular data, DP is also extended to set-valued data anonymization. In [149]–[151], Chen et al. proposed several anonymization techniques with DP guarantee for set-valued data in different scenarios.

C. Graph Data Anonymization

Now, we discuss our main focus of this section: *anonymization techniques for graph data*. With the emergence of many graph data, e.g., social networks, Internet, WWW, collaboration networks, anonymous systems, mobility traces (which can modeled by graph data by applying sophisticated techniques [3], [4], [26], [27], [56]), and email networks, the security and privacy issues raised during the publishing of these data have attracted a lot of attention as of recent [9]–[23]. Compared to traditional relational data (e.g., micro/tabular/set-valued data), anonymizing graph data is more challenging. First and intuitively, the structure of graph data is much more complicated. Consequently, in addition to the semantic information carried by data, the correlation and structure information among users should also be protected. Second, it is more difficult to model the auxiliary information available to adversaries, e.g., the widely available and accessible social information make the secure publishing of social data extremely challengeable [1]. Last but not least, it is more challenging to quantitatively measure the information of anonymizing graph data than relational data [33]. Therefore, anonymization techniques for relational data (micro/tabular/set-valued data) cannot be applied to graph data, and thus researchers have spent a lot of efforts to design effective graph data anonymization techniques [33]. Below, we summarize and categorize existing graph data anonymization techniques.

1) Naive ID Removal: To publish graph data, a straightforward method is by *naive ID removal*. Although this method has been demonstrated to be extremely vulnerable to *structure based de-anonymization attacks* (see Section IV), it is still widely used because of its simplicity, easy applicability, and scalability (e.g., a recent privacy leakage incident of the data indicating the locations of New York City's taxi drivers due to the poor data anonymization [152]) [1], [3], [4], [74].

2) Edge Editing based Anonymization: To protect graph data's privacy, Ying and Wu proposed spectrum preserved Edge Editing (EE) based schemes *Add/Del* and *Switch* [9]. Let $G(V, E)$ be a graph dataset[3], where $V = \{i | i \text{ is a user}\}$ is the set of users and $E = \{e_{i,j} | i, j \in V$, there is a relationship between i and $j\}$ is the set of all the possible relationships (e.g., friendships, contacts, and collaboration relationships) among the users in V. Under Add/Del, k randomly chosen edges will be added to E followed by another k randomly chosen edges will be deleted from E. Under Switch, k *edge switches* are conducted, where for each edge switch, two existing edges $e_{i,j}$ and $e_{u,v}$, such that $e_{i,j}, e_{u,v} \in E$ and $e_{i,v}, e_{u,j} \notin E$, are randomly selected and switched to $e_{i,v}$ and $e_{u,j}$.

3) k-anonymity: As we discussed before, k-anonymity has been widely used to anonymize relational data. Similarly, many efforts have been spent to extend k-anonymity to graph data [10]–[14]. To defend against *neighborhood attacks*, Zhou and Pei proposed *k-Neighborhood Anonymity* (k-NA) for graph data [10]. k-NA is a two-step scheme. In the first step, the neighborhoods of all users (1-hop neighborhoods) are extracted and encoded in a concise way. In the second step, the users with similar neighborhoods are greedily grouped together until each group consists of at least k users, and then each group is anonymized such that any neighborhood has at least $k-1$ isomorphic neighborhoods in the same group. In another work, Liu and Terzi considered *degree attacks* and proposed *k-Degree Anonymity* (k-DA) for graph data, under which for each user, there exists at least $k-1$ other users with the degree [11]. k-DA also consists of two steps. First, based on the degree sequence of a graph, a new k-anonymous degree sequence (any degree appears at least k times in the sequence) is constructed. Second, an anonymized graph is constructed based on the k-anonymous degree sequence.

In [12], Zou et al. simultaneously considered four types of structural attacks to graph data: *neighborhood attacks* [10], *degree attacks* [11], *subgraph attacks* [15], [74], and *hub-fingerprint attacks* [15]. To defend against these attacks, they proposed *k-automorphism* (k-auto), under which for each user, there are always $k-1$ other symmetric users with respect to $k-1$ automorphic functions. To achieve k-auto, three techniques are developed, namely *graph partitioning*, *block*

[3]For simplicity and clarity, we use the same notation system as in existing work [4], [9]–[23], [25], [28], [33], the structure of a graph dataset is modeled as a graph $G(V, E)$.

alignment, and *edge copy*. Another similar work is [13], where Cheng et al. proposed k-*isomorphism* (k-iso) to defend against structural attacks. Under k-iso, a graph is partitioned and anonymized into k disjoint subgraphs such that all these subgraphs are isomorphic. To ensure k-iso, both baseline and refined algorithms are designed. Furthermore, the authors demonstrated that k-iso is equivalent to k-auto in defending against user-deanonymization attacks.

In [14], Yuan et al. considered personalized privacy protection for anonymizing graph data in terms of both semantic and structural information. Based on the adversary's semantic and structural background knowledge, they customized three levels of privacy protection. Subsequently, different techniques are designed based on label generation (semantically) and noising edge/user addition (structurally) to achieve k-anonymity.

4) Aggregation/Class/Cluster based Anonymization: Another popular idea to protect graph data is to anonymize users into *clusters* (equivalently, *groups, classes*) [15]–[17]. In [15], Hay et al. proposed an *aggregation based graph anonymization* algorithm, which first partitions users and then describes the graph at the level of partitions. The anonymized graph consists of *supernodes*, each corresponding to the users in a partition, and *superedges*, indicating the edge densities among supernodes. Another work in the semantics level is [16], where Bhagat et al. designed an *interactive query-oriented* anonymization algorithm to partition a graph into classes with respect to users' attributes (labels). In [17], Thompson and Yao first presented two clustering algorithms, named *bounded* t-*means* and *union-split* respectively, to classify users with similar rules into clusters. Subsequently, they proposed a *matching-based anonymization* scheme for graph data by strategically adding and removing edges according to users' inter-cluster connectivity.

5) Differential Privacy: Recently, there are some works that seek to enable differentially private graph data release. Aiming at protecting *edge/link privacy*, defined as *the privacy of users' relationship (e.g., friendship, contact, collaboration, email) in graph data*, in [18], Sala et al. introduced *Pygmalion*, a *differentially-private graph model*. In Pygmalioin, a graph is first modeled by dK-series, i.e., the degree distributions of connected components of some size K within a target graph. Subsequently, the dK-series is perturbed to meet ϵ-DP. Recently, to bypass many difficulties encountered when working with the worst-case sensitivity [18], Proserpio presented a general platform, named *wPING*, for differentially private data analysis and publishing [19], [20]. Compared to previous solutions which scale up the magnitude of noise for challenging queries, wPING achieves better accuracy by scaling down the contributions of challenging records. Similar to [18], Wang and Wu also employed the dK-graph generation model for enforcing *edge DP* in graph anonymization. Another recent work for edge DP is [22], where Xiao et al. observed that, by estimating the connection probabilities among users instead of considering the edges directly, the noise scale enforced by edge DP can be significantly reduced. Following this observation, they proposed a *Hierarchical Random Graph* (HRG) model based scheme to meet edge DP.

6) Random Walk based Anonymization: In [23], Mittal et al. proposed a *Random Walk (RW) based anonymization* technique for preserving *link (edge) privacy*. By this technique, an edge between two users i and j is replaced by another edge between i and u, where u is the destination of a random walk starting from j.

Note that, in addition to the above graph anonymization techniques, some other techniques also have been proposed for multiple scenarios. For instance, Liu et al. in [153] studied DP under dependent tuples and proposed dependent differential privacy, which could be future applied to graph data. In another work [154], Liu and Mittal designed LinkMirage which enables privacy preserving analytics on social relationships. There also some graph anonymization techniques that consider context/semantic information. For instance, Beato et al. focused on contextual privacy and proposed Friend in the Middle (FiM) to defend against the re-identification of users in social networks [155]. In another work [156], Beato studied a privacy issue for online social networks, whereby sensitive information can be inferred from the behavior and actions of users when browsing contents in an online social network. They proposed VirtualFriendship along with the concept of routing friends in terms of social trust to achieve communication privacy. In this paper, we mainly focus on the structure-based graph anonymization techniques that can be applied to protect general graph data (specifically, nodes/users) privacy.

D. Graph Anonymization and Utility Analysis

Basically, an anonymization scheme can be evaluated from two perspectives: *data utility preservation* and *resistance to de-anonymization attacks*. In this subsection, we focus on comprehensively analyzing the utility of existing graph data anonymization algorithms and defer the detailed resistance analysis to Section IV-D. Before performing the analysis, we first define the used utility metrics, which can be classified as *graph utility metrics* and *application utility metrics*.

1) Graph Utility Metrics: Graph utility captures how the anonymized data preserves fundamental structural properties of the original graph after applying an anonymization technique. Particularly, we examine 15 graph utilities of existing anonymization schemes as follows.

- *Degree (Deg.)*. Deg. refers to the degree distribution, which is one of the most fundamental characteristics of a graph.
- *Joint Degree (JD)*. JD refers to the degree distribution $\{p_{(x,y)}|p_{(x,y)}$ is the fraction of edges in a graph that connect users of degree x and degree $y\}$.
- *Effective Diameter (ED)*. ED is defined as the minimum number of hops in which 90% of all connected pairs of nodes can reach each other.
- *Path Length (PL)*. PL refers to the distribution of the shortest path lengths between all pairs of users.
- *Local Clustering Coefficient (LCC) and Global Clustering Coefficient (GCC)*. *Clustering coefficient* measures the degree to which users in graph data tend to cluster together. The LCC of a user quantifies how close its

neighbors are to being a *clique*. The GCC is based on triplets of users. Let n_t and n_c be the number of *triangles* and the number of *connected triples of users* in a graph, respectively. Then, GCC is defined as GCC $= \frac{3n_t}{n_c}$.

- *Closeness Centrality (CC).* CC is defined as the *inverse of the farness* of a user within a graph, which measures how long it takes to spread information from a user to all other users sequentially.
- *Betweenness Centrality (BC).* BC quantifies the number of times a user acts as a bridge along the shortest path between two other users.
- *EigenVector (EV).* The EV of the adjacency matrix A of a graph G is a non-zero vector $\mathbf{v}$ such that $A\mathbf{v} = \lambda\mathbf{v}$, where λ is some scalar multiplier.
- *Network Constraint (NC).* A user's NC measures the extent to which he links to others that are already linked to each other.
- *Network Resilience (NR)* [157]. NR measures how robust a graph is, which is defined as the number of users in the *largest connected component* when users are removed from the graph in the degree decreasing order.
- *Infectiousness (Infe.)* [158]. Infe. measures the number of users infected by a disease, given that a randomly chosen user is infected and each infected user transmits this disease to its neighbors with some *infection rate*.
- *PageRank (PR)* [159]. PR measures the importance of a user within a graph using the PR algorithm.
- *Hub Score (HS) and Authority Score (AS)* [159]. HS and AS are two metrics measuring the extent that a node points to others and is pointed to by others, respectively.

2) Application Utility Metrics: In reality, most data is published for data/network mining tasks, high-level applications, etc. Therefore, besides examining anonymized graph data's fundamental structural utility, it is also crucial to ensure the anonymized data is useful for actual applications. Toward this objective, we also evaluate 7 popular *application utilities* for anonymization schemes.

- *Role eXtraction (RX) [160].* Based on users' structural behavior, users in a graph can be labeled as different roles, e.g., *clique members*, *periphery-nodes*. Role extraction is an important operation for graph data that is useful for many network mining tasks such as sense-making, searching for similar users, etc. We measure the RX utility of an anonymization scheme using the method in [160].
- *Reliable Email (RE) [161].* RE is a whitelisting system leveraging users' neighborhoods to filter and block spam emails. To evaluate the structural utility of an anonymization scheme with respect to RE, we take a similar method as in [18] to compute the number of users in a network who can be spammed by a fixed number of compromised neighbors in a graph.
- *Influence Maximization (IM) [162].* The IM problem seeks to find a set of k users such that these k users have the maximum influence to the network under some influence propagation model, e.g., *linear threshold model*, *cascade model*. IM is important for many real world applications, e.g., advertisements, public relations campaigns. For our purpose, we evaluate the IM application utility of an anonymization scheme using the recently proposed method [162].
- *Minimum-sized Influential Node Set (MINS) [163].* MINS is another popular and important application utility metric that leverages a graph's structure, which is to identify the minimum-sized set of influential nodes, such that every other node in the network could be influenced no less than a threshold. MINS can be used in many meaningful applications, e.g., social problems (drinking, smoking, addicting to gaming) alleviation, new products promotion. We evaluate the MINS application utility of an anonymization scheme using the recent method [163].
- *Community Detection (CD) [164] [165].* CD is a popular application on graph data which enables comprehensive analysis of a network structure and supports other applications, e.g., classification, routing (information propagation). To measure the CD utility of an anonymization scheme, we employ the hierarchical agglomeration algorithm proposed in [164].
- *Secure Routing (SR) [166].* The structure of graph data can also be used to improve the performance and security of secure routing for systems such as P2P systems. For our purpose, we evaluate the SR application utility of an anonymization scheme using the method designed in [166].
- *Sybil Detection (SD) [167], [168].* In a Sybil attack, an adversary tries to subvert a system by forging multiple identities. Sybil attacks are a serious threat to both centralized and distributed systems, e.g., recommendation systems, anonymity systems. Recently, several effective schemes, e.g., SybilLimit [167], DSybil [168], have been proposed to defend against Sybil attacks. For our purpose, we evaluate the SD application utility of an anonymization scheme using the method in [167].

3) Anonymization vs Utility: Now, we are ready to comprehensively analyze existing graph data anonymization techniques. We summarize the *Objective* (Obj.), *Complexity* (Compl.), and *graph and application utility* performance, of existing graph data anonymization schemes in Table II, where *N/L = Node/Link (user/relationship) privacy*, ✓ = *preserving the utility*, ◐= *preserving the utility with partial loses*, ⧫ = *conditionally preserving the utility depending on parameters and considered data*, ✗ = *not preserving the utility*, ***n/a*** = *not available*, and n is number of users (nodes) in the anonymized graph. Here, we explicitly distinguish the difference between "partially preserve data utility" and "conditionally preserve data utility" by considering anonymization parameters. For instance, RW can partially preserve data utility given an arbitrary anonymization parameter (i.e., the random walk step), while k-anonymity schemes conditionally preserve data utility depending on the chosen anonymization parameter k. We analyze and discuss the results in Table II as follows.

- For the Naive ID removal scheme, it is straightforward that it preserves all the structural data utility. However,

TABLE II

ANALYSIS OF EXISTING GRAPH ANONYMIZATION TECHNIQUES. DEG. = DEGREE, JD = JOINT DEGREE, ED = EFFECTIVE DIAMETER, PL = PATH LENGTH, LCC = LOCAL CLUSTERING COEFFICIENT, GCC = GLOBAL CLUSTERING COEFFICIENT, CC = CLOSENESS CENTRALITY, BC = BETWEENNESS CENTRALITY, EV = EIGENVECTOR, NC = NETWORK CONSTRAINT, NR = NETWORK RESILIENCE, INFE. = INFECTIOUSNESS, PR = PAGERANK, HS = HUB SCORE, AS = AUTHORITY SCORE, RX = ROLE EXTRACTION, RE = RELIABLE EMAIL, IM = INFLUENCE MAXIMIZATION, MINS = MINIMUM-SIZED INFLUENTIAL NODE SET (MINS), CD = COMMUNITY DETECTION, SR = SECURE ROUTING, AND SD = SYBIL DETECTION. N/L = NODE/LINK (USER/RELATIONSHIP) PRIVACY, ✓ = PRESERVING THE UTILITY, ◐= PRESERVING THE UTILITY WITH PARTIAL LOSES, ♦ = CONDITIONALLY PRESERVING THE UTILITY DEPENDING ON PARAMETERS AND CONSIDERED DATA, ✗ = NOT PRESERVING THE UTILITY, N/A = NOT AVAILABLE, AND n IS NUMBER OF USERS (NODES) IN THE ANONYMIZED GRAPH.

	Obj.	Compl.	graph utility															application utility						
			Deg.	JD	ED	PL	LCC	GCC	CC	BC	EV	NC	NR	Infe.	PR	HS	AS	RX	RE	IM	MINS	CD	SR	SD
Naive	N	$O(n)$	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Add/Del [9]	N, L	$O(n^3)$	◐	♦	◐	◐	♦	♦	◐	◐	◐	♦	◐	◐	♦	◐	◐	✗	◐	◐	♦	✗	♦	♦
Switch [9]	N, L	$O(n^3)$	✓	♦	♦	◐	♦	♦	◐	◐	◐	◐	♦	◐	◐	◐	◐	♦	◐	◐	♦	♦	◐	◐
k-NA [10]	N	$O(n^4)$	♦	♦	♦	♦	♦	♦	◐	♦	♦	◐	◐	◐	♦	♦	♦	✗	◐	♦	♦	♦	♦	♦
k-DA [11]	N	$O(n^2)$	♦	♦	♦	♦	♦	♦	◐	♦	♦	◐	◐	◐	♦	♦	♦	✗	◐	♦	♦	♦	♦	♦
k-auto [12]	N	$\Omega(n^4)$	♦	♦	♦	♦	♦	♦	◐	♦	♦	◐	◐	◐	♦	♦	♦	✗	◐	♦	♦	♦	♦	♦
k-iso [13]	N, L	$\Omega(n^3)$	♦	♦	✗	✗	♦	✗	✗	✗	✗	♦	✗	✗	♦	♦	♦	✗	✗	✗	♦	♦	✗	♦
Aggregation [15]	N	$\Omega(n^2)$	♦	♦	♦	♦	♦	♦	◐	♦	♦	◐	◐	◐	♦	♦	♦	✗	◐	♦	♦	♦	♦	♦
Cluster [17]	N	$O(n^2)$	♦	♦	♦	♦	♦	♦	◐	♦	♦	◐	◐	◐	♦	♦	♦	✗	◐	♦	♦	♦	♦	♦
DP [18]	L	$O(n^2)$	♦	♦	♦	◐	♦	♦	◐	♦	♦	♦	♦	◐	♦	♦	♦	✗	◐	♦	♦	✗	♦	♦
DP [19], [20]	L	$O(n^2)$	♦	♦	♦	◐	♦	♦	◐	♦	♦	♦	♦	◐	♦	♦	♦	✗	◐	♦	♦	✗	♦	♦
DP [21]	L	$O(n^2)$	♦	♦	♦	◐	♦	♦	◐	♦	♦	♦	♦	◐	♦	♦	♦	✗	◐	♦	♦	✗	♦	♦
DP [22]	L	**n/a**	♦	♦	♦	◐	♦	♦	◐	♦	♦	♦	♦	◐	♦	♦	♦	✗	◐	♦	♦	✗	♦	♦
RW [23]	L	$O(n^2)$	✓	♦	♦	♦	♦	✗	◐	♦	♦	◐	♦	◐	◐	◐	◐	✗	◐	♦	✗	✗	◐	◐

it is also the most vulnerable scheme to structure-based de-anonymization attacks.

- Add/Del and Switch are both designed to protect the node and link privacy of graph data [9]. Since Add/Del randomly adds and deletes edges, which is an *global edge edition operation* and thus it may change many fundamental structural properties of a graph. It follows that it can conditionally or partially preserve both graph and application utilities. Some of these utilities, e.g., JD, LCC, GCC, NC, MINS, SR, SD, would be destroyed if too many existing edges are deleted while new edges are added. For Switch, it switches two randomly selected qualified edges, which can preserve the degree of each user and thus is a *local edge edition operation*. Consequently, Switch can preserve Deg. and other utilities with partial loss. Furthermore, compared to Add/Del, Switch can conditionally preserve the RX and CD utilities which are destroyed in Add/Del. This is because that Add/Del randomly changes users' degree in the global edge edition process and thus some global structure-sensitive high-level application utilities are lost or significantly affected. For the resistance of ADD/Del and Switch, they cannot defend against modern structure-based de-anonymization attacks as shown in [1], [3], [4], [26].
- For the k-anonymity based anonymization schemes (k-NA [10], k-DA [11], k-auto [12], and k-iso [13]), k-NA [10], k-DA [11], and k-auto [12] can partially/conditionally preserve the graph and most application utilities except for the RX utility. This is because the fundamental idea of k-anonymity based schemes is to make k users/subgraphs are structurally similar. Therefore, there is a tradeoff between anonymity and utility. If k is large, more users will be structurally similar while more utility will be lost. On the other hand, if k is chosen to be small, more utility will be preserved at the cost of low guaranteed anonymity. Furthermore, since any user is guaranteed to be structurally similar to at least $k-1$ other users and meanwhile, the RX utility tries to distinguish users based on their structural difference, it turns out k-anonymity based schemes cannot preserve the RX utility. As we discussed before, k-iso achieves structure anonymization by partitioning the original graph into k isomorphic subgraphs. Therefore, several fundamental properties of a graph will be destroyed, e.g., connectivity. It follows that several important graph and application utilities are lost in k-iso, e.g., PL, GCC, NR, Infe., RX, RE, IM, and SR. Finally, compared with other schemes, k-NA, k-auto, and k-iso have higher computational complexities.
- Similar to k-anonymity based schemes, the cluster based schemes [15], [17] can conditionally/partially preserve graph and application utilities except for RX. This is because the fundamental idea of cluster based schemes is to make the users within a cluster structurally indistinguishable. Therefore, to which extent these schemes can preserve data utility depends on the cluster size setting. Again, since RX is achieved based on users' structural difference, this utility is not preserved in cluster based schemes.
- For DP based schemes [18]–[22], their main objective is to protect link privacy by perturbing edges of a graph. The fundamental idea of these schemes is to make an anonymized graph structurally similar to its neighboring graphs and thus an adversary cannot infer the existence of an edge. Therefore, they can conditionally/partially preserve most graph and application utilities. On the other hand, if a high level of privacy is guaranteed, many edges in the graph are changed. Furthermore, similar to Add/Del, the edge perturbation in DP also belongs to *global edge edition*. Therefore, the global structure-

sensitive high-level application utilities, e.g., RX, MINS, and CD, are destroyed or significantly reduced in DP based schemes.

- In RW based scheme [23], link privacy is achieved by replacing a random walk path with an edge, and thus this scheme will not change the degree distribution of the original data. It follows several utilities, e.g., Deg., RX, SD, NR, Infe., can be preserved or partially preserved. However, some other global utilities, e.g. JD, GCC., are lost in the RW based scheme due to the lose of the basic PL utility.
- For k-auto [12], k-iso [13], and Aggregation [15], we provide the lower bounds of their complexities. This is because some of their operations induce exponential/quasi-exponential time complexity, e.g., subgraph matching. For DP based scheme in [22], its complexity depends on the convergence of a problem-dependent Markov Chain Monte Carlo (MCMC) procedure.

E. Summary

In this section, for completeness, we first summarize and discuss the anonymization techniques for traditional micro, tabular, and set-valued data. Then, we focus on discussing and analyzing graph data anonymization techniques in the literature. We classify them into six categories based on the employed techniques and discuss the primary idea of each category. Finally, we comprehensively evaluate and analyze both the graph utility and the application utility performance of existing graph anonymization techniques, where graph utility captures fundamental topological properties of the original data while the application utility measures the usability and value of the anonymized data for high-level applications, e.g., data mining and machine learning tasks. According to our analysis, one main conclusion is that most existing anonymization techniques can partially or conditionally preserve most graph utility. However, all the anonymization techniques lose one or more application utility.

IV. Graph Data De-anonymization and Analysis

In this section, we first summarize state-of-the-art data de-anonymization attacks. Subsequently, we systematically analyze existing graph data de-anonymization techniques. Third, we analytically study the performance of graph data anonymization techniques against the power of graph data de-anonymization attacks. Again, we start from discussing representative de-anonymization attacks on relational data.

A. Relational Data De-anonymization

In [169], Lakshmanan et al. studied how safe anonymized data is with respect to protecting users' identities. They proposed various classes of belief functions to capture various degrees of partial information possessed by an adversary, and derived formulas for computing the expected number of cracks. In [79], Narayanan and Shmatikov presented a class of statistical de-anonymization attacks against high-dimensional micro-data. They further demonstrated the effectiveness of these attacks by successfully de-anonymizing the Netflix Prize dataset. In [170], Cormode studied the effectiveness of the *minimality attack*, which is an information inferring attack raised due to over-eager attempts to minimize the information lost by anonymization. Through careful analysis and experiments, they concluded that the impact of such attacks can be minimized.

In [171], Nanavati et al. presented an attack against reviewer anonymity. They showed that with access to a relatively small corpus of reviews, simple classification techniques from existing toolkits can successfully de-anonymize reviewers with reasonably high accuracy. In [172], Cormode studied the ability of an adversary to use data meeting privacy definitions to build an accurate classifier. They showed that private data can be accurately inferred even under DP. Furthermore, they observed that DP and ℓ-diversity are similar against classifier-based inference attack. In [173], Merener improved Narayanan and Shmatikov's work [79] on the de-anonymization of micro-data. They provided new results by considering cases where the auxiliary information has error and the dataset contains null values. Given auxiliary information of user's behavior, Unnikrishnan and Naini studied strategies for de-anonymizing user statistics [174]. Particularly, they obtained an asymptotically optimal strategy when users' data following an independently and identically distribution model.

B. Graph Data De-anonymization

1) Seed-based De-anonymization: To de-anonymize graph data, it is intuitive to identify some users first as seeds. Then, the large scale de-anonymization is bootstrapped from these seeds. In [74], Backstrom et al. presented both active attacks and passive attacks to graph data. The active attacks are carried out in three steps. First, an adversary chooses a set of victims. Subsequently, the adversary creates some sybil accounts with edges linked to the victims, as well as a pattern of links among the sybil accounts before the data release. Finally, after data release, the adversary identifies the sybil accounts according to their structural pattern and then de-anonymizes the victims. In the passive attacks, an adversary is an internal user of the system and tries to de-anonymize the users around him after data release. The attacks in [74] have several limitations, e.g., they are not scalable, and they leverage sybil users that can be detected by modern sybil defense techniques [167], [168]. To improve the attacks in [74], Narayanan and Shmatikov presented a *scalable two-phase de-anonymization attack* to social networks [1]. In the first phase, some seed users are identified between the anonymized graph and the auxiliary graph. In the second phase, starting from the identified seeds, a self-reinforcing de-anonymization propagation process is iteratively conducted based on both graphs' structural characteristics, e.g., node degrees, nodes' eccentricity, edge directionality. Later, Narayanan employed a simplified version of the attack in [1] (using less de-anonymization heuristics) for link prediction [75]. Besides that, they also proposed a new simulated annealing-based weighted graph matching algorithm for the seed identifying phase (the first phase). In [2], Nilizadeh et al. further improved Narayanan and Shmatikov's

attack by proposing a *community-enhanced* de-anonymization scheme of social networks. Specifically, the scheme first de-anonymizes a social network at the community-level. Then, users within de-anonymized communities are further de-anonymized according to similar heuristics as in [1]. Actually, the community-level de-anonymization in [2] can also be applied to enhance other de-anonymization attacks [3], [4], [25], [26], [28], [29].

In [3], Srivatsa and Hicks presented three attacks to de-anonymize mobility traces, which can be modeled as contact graphs applying multiple preprocessing techniques (e.g., [3], [56]). Similar to Narayanan-Shmatikov attacks [1], [75], Srivatsa-Hicks attacks also consist of two phases, where the first phase is for seed identification and the second phase is for mapping (de-anonymization) propagation. To achieve mapping propagation, Srivatsa and Hicks proposed three heuristics based on *Distance Vector* (DV), *Randomized Spanning Trees* (RST), and *Recursive Subgraph Matching* (RSM). In [26] [27], Ji et al. defined three similarity metrics, namely *structural similarity*, *relative distance similarity*, and *inheritance similarity*, and proposed two two-phase de-anonymization attack frameworks, named De-Anonymization (DA) and Adaptive De-Anonymization (ADA), which are workable when the auxiliary data only has partial overlap with the anonymized data.

In [25], [28], besides quantifying the de-anonymizability of graph data, the authors also proposed de-anonymization attacks. In [25], Yartseva and Grossglauser proposed a very simple *percolation-based de-anonymization algorithm* to graph data. Given a seed mapping set, the algorithm incrementally maps every pair of users (from the anonymized and auxiliary graphs respectively) with at least r neighboring mapped pairs, where r is a predefined mapping threshold. Another similar attack was presented by Korula and Lattanzi [28], which is also starting from a seed set and iteratively maps a pair of users with the most number of neighboring mapped pairs.

2) Seed-free De-anonymization: Recently, following another track, some powerful seed-free de-anonymization attacks on graph data are proposed. Using degrees and distances to other nodes as a nodes' fingerprints, Pedarsani et al. proposed a *Bayesian model based seed-free algorithm* for graph data de-anonymization [29]. Starting from nodes with the highest degree, the algorithm iteratively updates the fingerprints of all the nodes and performs a *maximum weighted bipartite graph matching* for de-anonymization. Another seed-free de-anonymization attack to graph data was presented by Ji et al. [4] [5]. Unlike previous attacks, Ji et al.'s attack is an *optimization based single-phase cold start algorithm*. Following their theoretical analysis, their attack is iteratively conducted and self-reinforced with the objective of *minimizing the edge difference* between the anonymized graph and auxiliary graph.

3) Other Techniques: There are some other techniques that de-anonymize graph data, e.g., semantics based de-anonymization attacks [76] [175], attacks to *ego graphs* [77], attacks against the link privacy of graph data [78]. By leveraging *web browser history stealing attack*, Wondracek et al. presented a de-anonymization attack to social networks based on users' *group membership information* [76]. Another semantics information based de-anonymization attack is [175], where Qian et al. studied to de-anonymize social networks and infer private attributes using knowledge graphs. Since we focus on structure-based de-anonymization attacks in this paper, we do not consider this kind of semantics based attacks. In [77], Sharad and Danezis studied the de-anonymization attacks to ego graphs with graph radius of one or two, which is a very special case of the general graph de-anonymization attacks studied in this paper. In [78], Korolova studied the link privacy leakage of anonymized social networks. In this paper, we focus on the de-anonymization attacks to the nodes (i.e., users) of graph data.

C. Graph De-anonymization Analysis

In this subsection, we analyze the performance of existing graph data de-anonymization algorithms. We show the results in Table III, where *Compl. = algorithm complexity*, *SF = seed-free*, *AGF = auxiliary graph-free*, *SemF = semantics-free*, *A/P = active/passive attack*, *Scal. = scalable*, *Prac. = practical*, *Rob. = robust to noise*, ✓ = *true*, ◐= *partially true*, ♦ = *conditionally true*, ✗ = *false*, n is the number of users (nodes) in the anonymized graph, b and γ are some constant values, Λ is the number of seed mappings, and $O(\hbar)$ is the complexity of the enhanced de-anonymization scheme in the Nilizadeh et al. attack. We analyze and discuss the results in Table III as follows.

- Except for Backstrom et al. attacks, all the existing structure-based de-anonymization attacks are passive attacks and require auxiliary graphs to perform the attack, i.e., they employ the structural similarity between the anonymized graphs and auxiliary graphs to break the privacy of anonymized data. *Unfortunately, when we examine the anonymization schemes in Table II, none involve such auxiliary information in their threat models.* On the other hand, based on existing quantification results (e.g., [4], [6], [28], [30]), the similarity between anonymized graphs and auxiliary graphs is sufficient to make anonymized users perfectly or partially de-anonymizable.
- The Nilizadeh et al. attack has a complexity of $O(n^2+\hbar)$, where $O(n^2)$ is the complexity of the *community-level de-anonymization* and $O(\hbar)$ is the complexity of the enhanced de-anonymization attack, i.e., the user-level de-anonymization. For instance, if the Narayanan-Shmatikov attack is employed for user-level de-anonymization, $O(\hbar) = O(n^\Lambda + n^4)$. Similarly, the Nilizadeh et al. attack is conditionally seed-free depending on the enhanced algorithm.
- To perform Backstrom et al. attacks [74], an adversary either has to launch some Sybil users before the actual anonymized data release, or has to be an internal user that knows its neighborhoods. In either case, such attacks can only de-anonymize some users instead of in large scale. Furthermore, the attacks cannot tolerate a topological change of the original data. Therefore, Backstrom et al. attacks are not scalable or robust. For their practicability,

TABLE III
ANALYSIS OF EXISTING GRAPH DE-ANONYMIZATION TECHNIQUES. COMPL. = ALGORITHM COMPLEXITY, SF = SEED-FREE, AGF = AUXILIARY GRAPH-FREE, SEMF = SEMANTICS-FREE, A/P = ACTIVE/PASSIVE ATTACK, SCAL. = SCALABLE, PRAC. = PRACTICAL, ROB. = ROBUST TO NOISE, ✓ = TRUE, ◐= PARTIALLY TRUE, ♦ = CONDITIONALLY TRUE, ✗ = FALSE, n IS THE NUMBER OF USERS (NODES) IN THE ANONYMIZED GRAPH, b AND γ ARE SOME CONSTANT VALUES, Λ IS THE NUMBER OF SEED MAPPINGS, AND $O(\hbar)$ IS THE COMPLEXITY OF THE ENHANCED DE-ANONYMIZATION SCHEME IN THE NILIZADEH ET AL. ATTACK.

	Compl.	SF	AGF	SemF	A/P	Scal.	Prac.	Rob.
Backstrom et al. [74]	$O(n^2)$	✓	✓	✓	A, P	✗	◐	✗
Narayanan-Shmatikov [1]	$O(n^\Lambda + n^4)$	✗	✗	✓	P	✓	✓	✓
Narayanan et al. [75]	$\sim O(n^4)$	✗	✗	✓	P	✓	✓	✓
Nilizadeh et al. [2]	$O(n^2 + \hbar)$	♦	✗	✓	P	♦	♦	♦
Srivatsa-Hicks-DV [3]	$\Lambda! O(n^3)$	✗	✗	✓	P	♦	♦	✓
Srivatsa-Hicks-RST [3]	$> \Lambda! O(n^3)$	✗	✗	✓	P	♦	♦	✓
Srivatsa-Hicks-RSM [3]	$\Lambda! O(n^{b+1})$	✗	✗	✓	P	♦	♦	✓
Pedarsani et al. [29]	$O(n^3)$	✓	✗	✓	P	✓	♦	♦
Yartseva-Grossglauser [25]	$O(n^3)$	✗	✗	✓	P	✓	♦	✓
Ji et al.-DA [26]	$O(n^3)$	✗	✗	✓	P	✓	✓	✓
Ji et al.-ADA [26]	$O(n^3)$	✗	✗	✓	P	✓	✓	✓
Korula-Lattanzi [28]	$O(n^3 \log n)$	✗	✗	✓	P	✓	♦	✓
Ji et al. [4]	$O(n^{\Theta(1)\log\gamma+1})$	✓	✗	✓	P	✓	✓	✓

it depends on whether an adversary can successfully launch Sybil users or be an internal user and obtain his neighborhoods.

- All the examined de-anonymization attacks are semantics-free. This is because the structural information itself is sufficient to perfectly or partially de-anonymize graph users, which can be seen from existing quantification results. Furthermore, compared to semantics information, structural information is widely available in large scale, resilient to noise, and easily computable [1], [3], [4]. Following this fact, all the attacks except for Backstrom et al. attacks are (conditionally) scalable, practical, and robust.
- Specifically, Srivatsa-Hicks attacks [3] are conditionally scalable and practical. This is because their attacks should try all the possible $\Lambda!$ seed mappings, which is very time consuming. For instance, for a large Λ, e.g., $\Lambda > 20$, the attacks are not practically feasible. Pedarsani et al. attack [29] is conditionally practical and robust. This is because it is very sensitive to the graph density of the anonymized data. Generally, this attack is suitable for sparse graphs however has a significant performance degradation when the graph density increases. Yartseva-Grossglauser attack [25] is conditionally practical because it is designed to de-anonymize users of degree no less than 4 in the anonymized data. In many real world graph datasets, the users with degree less than 4 could dominate or take a large part of graph data based on the statistics in [4]. The conditional practicability of Korula-Lattanzi attack [28] comes from its improper assumption that $\Theta(\iota \cdot n)$ ($\iota \in (0, 1]$ is a constant) users are available, which is too strong to hold for real world de-anonymization attacks. Note that, the *community-level de-anonymization* proposed in [2] is scalable (with complexity of $O(n^2)$). However, the Nilizadeh et al. attack [2] is conditionally scalable, practical, and robust. This is because, if the community-level de-anonymization of [2] is employed to enhance Srivatsa-Hicks attacks, Pedarsani et al. attack, Yartseva-Grossglauser attack, or/and Korula-Lattanzi attack, it is conditionally scalable, practical, and/or robust. The Narayanan-Shmatikov attack [1], Narayanan et al. attack [75], and Ji et al. attacks [4], [26] adaptively perform de-anonymization employing several heuristics based on graph's local and global structural characteristics. It follows they are scalable, practical, and robust as long as similarity exists between anonymized graphs and auxiliary graphs.
- For seed-based attacks (e.g., Narayanan-Shmatikov attack, Srivatsa-Hicks attacks) and seed-free attacks (e.g., Pedarsani et al. attack, Ji et al. attack), they all have advantages depending on the application scenarios. On one hand, seed-based attacks are more stable with respect to de-anonymizing arbitrary anonymized graphs. The reason is straightforward since seed knowledge provides more auxiliary information to an adversary. On the other hand, it is possible that in some scenarios that seeds are not available, and thus seed-free attacks are more general. Furthermore, if there is some error in the seed seeking phase (which is possible in real world attacks), seed-based attacks will suffer performance de-gradation or will possibly fail.
- Backstrom et al. attacks can be defended against by state-of-the-art anonymization algorithms. This is because an implicit assumption in Backstrom et al. attacks is that data publishers only anonymize the data by *naive ID removal*, i.e., no edge change (e.g., addition, deletion, switching) happened during the anonymization. Evidently, this assumption does not hold in any state-of-the-art anonymization schemes, and thus Backstrom et al. attacks can be defended. However, for other modern structure-based de-anonymization attacks ([1]–[4], [25], [26], [28], [29], [75]), we analyze their effectiveness in the following subsection.

D. Anonymization vs De-anonymization Analysis

After carefully analyzing existing anonymization and de-anonymization techniques, we summarize the ***vulnerability*** of

TABLE IV
DE-ANONYMIZATION ATTACKS VS ANONYMIZATION TECHNIQUES. NAIVE = NAIVE ID REMOVAL, EE = EE BASED SCHEMES [9], k-ANONY. = k-ANONYMITY BASED SCHEMES [10]–[14], CLUSTER = CLUSTER BASED SCHEMES [15]–[17], DP = DP BASED SCHEMES [18]–[22], RW = RANDOM WALK BASED SCHEME [23], AND ✓, ⧫, AND ✗ = THE ANONYMIZATION SCHEME IS VULNERABLE, CONDITIONALLY VULNERABLE, AND INVULNERABLE (I.E., RESISTANT) TO THE DE-ANONYMIZATION ATTACK, RESPECTIVELY.

	Naive	EE [9]	k-anony. [10]–[13]	Cluster [15], [17]	DP [18]–[22]	RW [23]
Backstrom et al. [74]	✓	✗	✗	✗	✗	✗
Narayanan-Shmatikov [1]	✓	✓	⧫	⧫	✓	✓
Narayanan et al. [75]	✓	✓	⧫	⧫	✓	✓
Nilizadeh et al. [2]	✓	⧫	⧫	⧫	✗	✗
Srivatsa-Hicks [3]	✓	✓	⧫	⧫	✓	✓
Pedarsani et al. [29]	✓	✓	⧫	⧫	✓	✓
Yartseva and Grossglauser [25]	✓	✓	⧫	⧫	✓	✓
Ji et al. [26]	✓	✓	⧫	⧫	✓	✓
Korula-Lattanzi [28]	✓	✓	⧫	⧫	✓	✓
Ji et al. [4]	✓	✓	⧫	⧫	✓	✓

state-of-the-art anonymization schemes in Table IV, where *Naive = naive ID removal*, *EE = EE based schemes [9]*, *k-anony. = k-anonymity based schemes [10]–[14]*, *Cluster = cluster based schemes [15]–[17]*, *DP = DP based schemes [18]–[22]*, *RW = random walk based scheme [23]*, and ✓, ⧫, *and ✗ = the anonymization scheme is vulnerable, conditionally vulnerable, and invulnerable (i.e., resistant) to the de-anonymization attack, respectively*. Here, we analyze and discuss the results in Table IV as follows.

- It has been shown in both academia and real world that naive ID removal anonymization cannot protect graph data's privacy. Therefore, naive anonymization is vulnerable to all the existing structure-based de-anonymization attacks.
- As we analyzed before, all other state-of-the-art anonymization schemes (e.g., EE, k-anony., Cluster, D-P, and RW) are resistant to Backstrom et al. attacks. Again, this is because an assumption of Backstrom et al. attacks is that data is anonymized by naive ID removal techniques, which is not true under state-of-the-art anonymization schemes.
- For EE based anonymization schemes ([9]), they are conditionally vulnerable to Nilizadeh et al.'s attack [2] and vulnerable to all the other modern structure-based de-anonymization attacks [1]–[4], [25], [26], [28], [29], [75]. This is because although EE can partially modify the structure of a graph, to preserve data utility, many structural properties, e.g., neighborhood, degree distribution, closeness/betweenness centrality distribution, and path length distribution, are generally preserved. Therefore, given an auxiliary graph consisting of the same or overlapping group of users with the anonymized graph, powerful de-anonymization heuristics can be designed based on these structural properties to break the privacy of EE based anonymization schemes. Furthermore, the availability of seed users make such heuristics more robust to the noise introduced by EE. For instance, Narayanan-Shmatikov's attack breaks EE by employing degree and neighborhood similarity [1], Srivatsa-Hicks's attacks break EE by employing path length and neighborhood similarity [3], Ji et al.'s attacks break EE by employing centrality similarity [26], etc. As we analyzed in Table II, EE based anonymization schemes (e.g., Add/Del) may destroy graphs' community utility, and thus they are conditionally vulnerable to Nilizadeh et al. attack [2].
- k-anonymity based anonymization schemes ([10]–[13]) are conditionally vulnerable to modern structure-based de-anonymization attacks [1]–[4], [25], [26], [28], [29], [75]. The reasons are as follows: k-anonimity is initially designed for traditional relational data, which makes any user semantically indistinguishable with $k-1$ other users. Unlike relational data which are structurally independent with each other, users in graph data have strong structure correlation in addition to semantic similarity. When researchers extended k-anonymity to graph data, they extended the concept of traditional semantics to graph data as different structural properties (e.g., degree, neighborhood, and subgraph), and designed schemes to make k users structurally indistinguishable with respect to some structural semantics, i.e., degree, neighborhood, subgraph, etc. *However, even if users in graph data cannot be distinguished with respect to some structural semantics, e.g., degree, neighborhood, subgraph, they can be de-anonymized by some other structural semantics, e.g., path length distribution, closeness centrality, betweenness centrality, or the combinations of several structural semantics.* Theoretically, the only way to make users indistinguishable with respect to all structural semantics is that this graph should be a *complete graph* or a *completely disconnected graph*, which also implies that all the data utility is destroyed. Therefore, as long as some data utility is preserved in the anonymized data, k-anonymity based schemes are vulnerable to modern structure-based de-anonymization attacks. The degree of vulnerability depends on how much data utility is preserved.
- Cluster based schemes ([15], [17]) are also conditionally vulnerable to modern structure-based de-anonymization attacks [1]–[4], [25], [26], [28], [29], [75]. The analysis is similar to that of k-anonymity. This is because the fundamental idea of cluster based schemes is to cluster users first and then to make the users within a cluster indistinguishable with respect to neighborhoods. Again, even if users are indistinguishable by neighborhoods, they

can be de-anonymized by other structural semantics or the combinations of other semantics, e.g., centralities scores, path distribution. Consequently, cluster based schemes are vulnerable as long as some data utilities are preserved in the anonymized data, and the vulnerability depends on the degree of data utility.

- DP and RW based schemes ([18]–[23]) are vulnerable to modern structure-based de-anonymization attacks except Nilizadeh et al.'s attack [2]. The reasons are as follows: First, they are designed with the objective of protecting the *link privacy* of graph data. It follows that no dedicated node privacy protection techniques are considered. Second, to protect link privacy, the edges are perturbed in DP based schemes and random walk paths are replaced by edges in the RW based scheme, both with a nice theoretical privacy guarantee. However, after the edge anonymization process, many data utilities, e.g., degree, path length distribution, are still preserved. This implies that, given an auxiliary graph, users are still de-anonymizable based on several structural semantics under DP and RW based schemes. Furthermore, as shown by Narayanan et al. in [75], links among users can be easily identified in the auxiliary graph after de-anonymizaing users. Again, as we analyzed in Table II, since DP and RW based schemes cannot preserve data's community utility, they are resistant to Nilizadeh et al.'s attack.

In summary, based on our analysis, state-of-the-art anonymization schemes are still vulnerable to modern de-anonymization attacks. The fundamental reason is that: first, existing anonymization schemes only ensure that graph data users indistinguishable with respect to some structural semantics (properties). However, other structural semantics, especially global ones, and the combinations of multiple structural semantics can still enable effective de-anonymization of users; and second, as one of the main objectives, all the anonymization schemes try to preserve as much data utility as possible. However, these data utility from the adversary's perspective is equivalent to structural information, which can be used along with an auxiliary graph for conducting powerful de-anonymization attacks.

E. Summary

In this section, we first discuss traditional representative relational data de-anonymization attacks. Subsequently, we systematically summarize and analyze existing graph data de-anonymization attacks, which are classified into seed-based de-anonymization attacks, seed-free de-anonymization attacks, and other de-anonymization attacks. Third, we analyze existing graph de-anonymization attacks in detail with respect to different performance metrics, e.g., scalability and robustness. The primary conclusion is that most structure-based de-anonymization attacks, especially seed-free de-anonymization attacks, are powerful and robust for large-scale graph data. Finally, we formally analyzed the vulnerability of state-of-the-art anonymization techniques against modern de-anonymization attacks. The most important conclusion is that existing anonymization techniques are still vulnerable when defending against de-anonymization attacks. The actual vulnerability depends on multiple factors. Therefore, it is still a serious yet open problem to develop effective graph data anonymization techniques.

V. Graph Data De-anonymizability Quantification

In this section, we summarize and analyze existing de-anonymizability quantification results.

A. Seed-based Quantification

In [25], Yartseva and Grossglauser quantified the de-anonymizability of graph data by analyzing a percolation-based graph matching algorithm under the *Erdös-Rényi (ER) random graph model* $G(n,p)$ (a random graph consists of n nodes/users, and an edge exists between any pair of nodes with probability p). Under the ER model, the degree distribution of the considered graph data should follow the *Poisson distribution* [4], [159]. However, real world graph data may follow any distribution (e.g., many social networks follow the *power-law distribution*), and more importantly, seldom do we see any graph data following the Poisson distribution [4], [159]. Therefore, the quantification under the ER model is only mathematically meaningful but not practical. Nevertheless, it can shed light on more practical quantification. Another limitation of [25] is that it leverages seed-associated structural information for de-anonymizability quantification. In fact, as shown in [4], [30], graph data is de-anonymizable based solely on data's structural information, i.e., without seed.

Following the same direction, Korula and Lattanzi conducted another seed-based de-anonymizablity quantification of graph data under both the ER model and the *Preferential Attachment (PA) model* [28]. Again, several limitations make the quantification in [28] unpractical. First, as we mentioned before, the ER model is a theoretical model (i.e., it is not practical). Accordingly, the PA model is more practical compared to the ER. However, it still has some limitations, e.g., the existence of *self-loops*. Second, as in [25], the quantification in [28] only considers the structural information associated with seeds. Finally and more importantly, the quantification in [28] is valid under a strong assumption of existing dense seeds ($\Theta(\iota \cdot n)$ available seeds, $\iota \in (0,1]$ is a constant), which is not true for real world de-anonymization attacks. Recently, Ji et al. quantified the seed-based de-anonymizability of social networks [6] [7] under both the ER model and a general *statistical graph model*. Compared to previous seed-based works, the quantification in [6] [7] considers the structural information among anonymized users in addition to the structural information between anonymized users and seeds.

B. Seed-free Quantification

In [30], Pedarsani and Grossglauser quantified the de-anonymizability of graph data under the ER model. They showed that an anonymized graph is de-anonymizable when certain conditions on the structures of anonymized and auxiliary graphs are satisfied. Again, the quantification is under the mathematical ER model, which cannot be applied to

real world graph data [4], [159]. Furthermore, for a de-anonymization attack, although it is improper to assume the availability of dense seeds, it is still reasonable to have some seed mappings as pre-knowledge [1], [3], [26], [74]. However, the quantification in [30] does not rely on seeds. Recently, Ji et al. improved the quantification in [30]. They quantified the *perfect* and *error-tolerated* de-anonymizability of graph data under a general *configuration model* [159], where the considered graph data can have an *arbitrary degree sequence*. Similar to [30], the quantification in [4] does not relay on seeds.

Recently, Ji et al. proposed a new and more sound concept of *relative de-anonymizability*, which quantifies the de-anonymizablity of graph data adaptively and more accurately by taking into account users' structural differences [31]. Compared to existing de-anonymizability quantification, the relative de-anonymizability quantification has two main advantages: (i) unlike existing works where all users are treated as equivalent, users' de-anonymizability is adaptively quantified according to their relative structural importance; and (ii) the relative de-anonymizability quantification extends existing results, and thus is much general than existing results.

C. De-anonymizability Quantification Analysis

We summarize the existing de-anonymizability quantifications in Table V, where *SBDA = Seed-Based De-Anonymization quantification*, *SFDA = Seed-Free De-Anonymization quantification*, *ET = Error Tolerance*, *Prac. = practical*, ✓ = *true*, ◐= *partially true*, and ✗ = *false*.

From Table V, we can see that most existing quantifications are either seed-based or seed-free. For the quantifications based on the ER model [25], [28], [30], they are not practical as we discussed in Section V-A. For the quantification based on the PA model [28], it is conditionally practical since it is based on an improper assumption: *the existence of dense seed users*, which usually does not hold in real world de-anonymization attacks.

Furthermore, the quantifications in [25], [28], [30] cannot tolerate any de-anonymization error, which induces a de-anonymizability condition that is too strict. In contrast, the quantifications in [4], [6], [31] consider possible de-anonymization error, which are more general and practical.

As discussed before, all the existing quantifications except for [31] overlook the structural differences among users by treating them as structurally equivalent. It follows that their obtained quantification bounds are loose. In contrast, the relative quantification adaptively quantifies the de-anonymizability of users in terms of their structural importance. Congruently, it is more sound and the obtained quantification bounds are more accurate.

D. Summary

In this section, we systematically discuss and analyze existing graph data de-anonymizability quantification techniques, which are generally classified into seed-based de-anonymizability quantification and seed-free de-anonymizability quantification, respectively. The main implication of existing de-anonymizability quantification is that structural information is important and theoretically, it is sufficient for conducting large-scale graph data de-anonymization even without of any seed knowledge. This is also consistent with many existing empirical results in the graph de-anonymization literature.

VI. Research Evolution, Future Research, and Challenges

A. Research Evolution Discussion

As a fundamental and challenging problem space, data anonymization/de-anonymization has attracted a significant amount of attention from researchers. With the emergence of *big data*, this research becomes even more important and more challenging. Particularly, we summarize the evolution of data anonymization/de-anonymization research in Tables VI and VII, from which we have the following observations.

- For anonymization techniques, following the seminal works of k-anonymity and DP, many schemes have been proposed to address the security and privacy concerns of both relational data and graph data in different scenarios, e.g., ℓ-diversity, (α, k)-anonymity, t-closeness, δ-presence, m-invariance, k^m-anonymity, (ϵ, δ)-DP, $(\mathbb{D}, \gamma)$-membership. This is mainly because these two privacy models provide formal methodologies for implementation, theoretical privacy guarantee, and moderate data utility preservation. Specifically, it seems that DP has attracted more research attention than k-anonymity recently. We conjecture that this is because DP is a relatively new technique and it provides an even stronger privacy guarantee than k-anonymity. However, proper application of DP for graph data anonymization is still in its infancy. Furthermore, the research of graph data anonymization started later than that of relational data, which is generally consistent with the evolution of computer data.
- With the popularity of graph data, more de-anonymization attacks on them have been presented as of recent. Similar to understanding the fundamental reasons that are responsible for the success of modern heuristic graph data de-anonymization attacks, researchers also began to conduct the research on quantifying the de-anonymizability of graph data.
- Most state-of-the-art graph data anonymization and de-anonymization schemes are based only on data's structural information. This is because (i) similar to the semantic information, the structure itself is also important information carried by graph data, which can be used for many data mining tasks and high level applications; (ii) many users in graph data have unique/quasi-unique topological structures, which can be used for identifying/quasi-identifying users; and (iii) compared to semantic information, structure information is easier to

TABLE V
SUMMARIZATION OF DE-ANONYMIZABILITY QUANTIFICATION RESULTS. SBDA = SEED-BASED DE-ANONYMIZATION QUANTIFICATION, SFDA = SEED-FREE DE-ANONYMIZATION QUANTIFICATION, ET = ERROR TOLERANCE, PRAC. = PRACTICAL, ✓ = TRUE, ◐= PARTIALLY TRUE, AND ✗ = FALSE.

Quantification	Model	SBDA	SFDA	Error Tolerance	Practicality	Generality
Pedarsani-Grossglauser [30]	ER	✗	✓	✗	✗	✗
Yartseva-Grossglauser [25]	ER	✓	✗	✗	✗	✗
Korula-Lattanzi [28]	ER	✓	✗	✗	✗	✗
	PA	✓	✗	✗	◐	◐
Ji et al. [4]	configuration	✗	✓	✓	✓	◐
Ji et al. [6]	ER	✓	✗	✓	✗	✗
	statistical	✓	✗	✓	✓	◐
Ji et al. [31]	configuration	✓	✓	✓	✓	✓

TABLE VI
ANONYMIZATION TECHNIQUES AND DE-ANONYMIZATION ATTACKS ON RELATIONAL (MICRO/TABULAR/SET-VALUED) DATA. THE *anonymization techniques that are italicized* ARE FOR SET-VALUED DATA WHILE THE OTHERS ARE FOR MICRO/TABULAR DATA.

year	anonymization	de-anonymization
2001/2	k-anonymity [108], [109]	
2005	k-anonymity [63], [112]	Lakshmanan et al. [169]
2006	ℓ-diversity [110], (α, k)-anonymity [115], permutation [117], (X, Y)-privacy [119], k-AC [126], utility-aware [129]–[131], ϵ-DP [133], (ϵ, δ)-DP [134]	
2007	k-anonymity [113], t-closeness [111], (c, k)-safety [116], permutation [118], δ-presence [120], m-invariance [121]	
2008	m-invariance [122], utility-aware [132], *k^m-anonymity [146], (h, k, p)-coherence [147]*	Narayanan-Shmatikov [79]
2009	LKC-privacy [123], P-private [124] (ϵ, δ)-DP [140], *k-anonymity [144]*	
2010	predictive [125], ϵ-DP [135], *ρ-uncertainty [148]*	Cormode et al. [170]
2011	ϵ-DP [136] *[149]*	Nanavati et al. [171], Cormode [172]
2012	k-AC [127], ρ-DI [141], (ϵ, δ)-DP [143], *k-anonymity/ℓ-diversity [145], ϵ-DP [150], [151]*	Merener [173]
2013	b-matching [128], ϵ-DP [137], $(\mathbb{D}, \gamma)$-membership [142]	Unnikrishnan-Naini [174]
2014	ϵ-DP [138], [139]	

TABLE VII
ANONYMIZATION, DE-ANONYMIZATION, AND QUANTIFICATION OF GRAPH DATA. **BOLD TECHNIQUES ARE ANONYMIZATION ALGORITHMS, DE-ANONYMIZATION ATTACKS, OR DE-ANONYMIZABILITY QUANTIFICATION TECHNIQUES BASED ONLY ON DATA'S STRUCTURAL INFORMATION.**

year	anonymization	de-anonymization	quantification
2007		**Backstrom et al. [74]**	
2008	**Add/Del [9], Switch [9], k-NA [10], k-DA [11], aggregation [15]**	**Korolova et al. [78]**	
2009	**k-auto [12]**, class [16], **cluster [17]**	**Narayanan-Shmatikov [1]**	
2010	**k-iso [13],** k-anonymity [14]	Wondracek et al. [76]	
2011	**ϵ-DP [18]**	**Narayanan et al. [75]**	**Pedarsani-Grossglauser [30]**
2012	**ϵ-DP [19]**	**Srivatsa-Hicks [3]**	
2013	**ϵ-DP [21], randomization [23]**	**Pedarsani et al. [29], Sharad-Danezis [77] Yartseva-Grossglauser [25]**	**Yartseva-Grossglauser [25]**
2014	**ϵ-DP [20], [22]**	**Ji et al. [26], Korula-Lattanzi [28] Ji et al. [4], Nilizadeh et al. [2]**	**Korula-Lattanzi [28] Ji et al. [4]**
2015			**Ji et al. [6]**
2016			**Ji et al. [31]**

obtain and analyze, which can be exploited for fast and effective de-anonymization attacks. Therefore, to protect graph data, researchers seek to anonymize the structural information, while to break the privacy of graph data, researchers try to exploit such information.

B. Future Research and Challenges

In this subsection, we discuss the future research directions and challenges of graph data anonymization, de-anonymization, and de-anonymizability quantification.

1) Graph Data Anonymization: According to our analytical results in Table IV, state-of-the-art anonymization techniques, e.g. k-anonymity based schemes, DP based schemes, are all still vulnerable to modern structure-based de-anonymization attacks. Their actual vulnerability depends on how much data utility is preserved in the anonymized data. Therefore, it is still an open yet very serious problem to *develop effective graph data anonymization techniques to defend against modern structure-based de-anonymization attacks*. The main challenges are two-fold.

- First, guaranteeing data utility is one of the primary objectives when publishing graph data. However, as we explained before, the preserved graph and application utilities enable adversaries to conduct large-scale de-

anonymization attacks. Therefore, it is a big challenge to effectively anonymize graph data with objective data utility preservation and without enabling adversaries to utilize these data utilities.

- Second, many local and global structural characteristics (or, structural scemantics), e.g., deg., LCC, CC, BC, are enabled in graph data's structure. State-of-the-art anonymization techniques can only make graph users structurally indistinguishable with respect to one or several semantics, e.g., degree and neighborhood. However, as we explained before, in many scenarios, several other structural semantics and their combinations are sufficient to enable a structure-based de-anonymization attack to uniquely distinguishable graph users. Therefore, it is also a key challenge to make graph users structurally indistinguishable with respect to the complete structural semantics.

Based on the above challenge analysis, it is difficult, if not impossible, to develop some anonymization techniques that can preserve all the data utility. Therefore, the potential promising direction is to develop *application-oriented graph data anonymization techniques* with the target of preserving desired data utility. Furthermore, if we take account the time dimension, graph data may be dynamically changed over time. For instance, social networks are dynamically changed instead of being static [176]. Hence, developing anonymization techniques for dynamic graph data is also a future research direction.

2) Graph Data De-anonymization: Future de-anonymization research may follow two directions.

- First, it is interesting to study how to combine the advantages of different algorithms and develop new stable and improved de-anonymization schemes. To achieve this, the challenge is to decide which structural characteristics should be employed and how to use these characteristics during the de-anonymization process. This is because some structural characteristics are local (e.g., Deg.) while others are global (e.g., CC and BC). Therefore, it is better to seek a balance between the employed local and global structural semantics. Meanwhile, some structural characteristics may carry similar structural semantics, and thus simultaneously employing such characteristics will not lead to too much improvement. Furthermore, according to our evaluation experience, the sequence and weights of applying different structural characteristics may induce very different de-anonymization performance.
- Second, instead of trying to design a uniformly optimal de-anonymization algorithm, it is better to develop some *anonymization technique-oriented and application-aware de-anonymization schemes*. This is because, for some anonymization algorithms, e.g., most k-anonymity based schemes and DP based schemes, they mainly achieve anonymity by local graph perturbation. In this scenario, the global graph characteristics based de-anonymization algorithms will be more effective. On the other hand, for some anonymization algorithms, e.g., *Add/Del* and RW, they mainly achieve anonymity through global graph perturbation. Therefore, to de-anonymize the data anonymized by these techniques, the local graph characteristics based de-anonymization schemes will be better. Furthermore, according to our de-anonymization evaluation experience, some de-anonymization attacks are more effective to de-anonymize dense graphs, e.g., Narayanan-Shmatikov attack and Ji et al. attack, while some other attacks are more effective to de-anonymize sparse graphs, e.g., Srivatsa-Hicks-DV, Pedarsani et al. attack. Therefore, when developing new de-anonymization algorithms, it is helpful to take account both the attacked anonymization technique and the attacked application.

3) De-anonymizability Quantification: There are still some open problems in the graph de-anonymizability quantification area.

- First, in all the existing quantifications, only local structural characteristics (degree and neighborhood) are considered. As shown by the empirical results in [8], in some scenarios, global structural characteristics are more powerful and stable in conducting de-anonymization attacks. Furthermore, as shown in the recent work [2], community information is also helpful in improving the performance of a de-anonymization attack. Therefore, it is expected to extend existing quantification by incorporating global structural characteristics and other network attributes into consideration, e.g., CC, BC, and community information.
- Second, in all the existing de-anonymizability quantification, the impacts of applying anonymization techniques are not considered. Therefore, it is also interesting to take account different anonymization techniques and conduct *anonymization technique-oriented de-anonymizability quantification*. The main challenge is how to develop a reasonable model to characterize the anonymized data.
- Third, although we have some progress in de-anonymizability quantification, it is still an open problem on *utility-de-anonymizability tradeoff quantification*, i.e., how to quantify the correlation between graph data utility and de-anonymizability. Such quantification will be helpful in guiding future anonymization and de-anonymization techniques development. To achieve such quantification, the main challenge is also how to develop a reasonable utility model and then quantify the data's de-anonymizability under the utility model.

C. Summary

In this section, we discuss the research evolution of graph data anonymization, de-anonymization, and de-anonymizability quantification, followed by the discussion of future research directions. For graph data anonymization, it is difficult, if not impossible, to develop some technique that can preserve all the data utility based on our analytical results. Thus, one promising direction is to develop application-oriented graph data anonymization techniques with the target of preserving desired data utility. Furthermore, for graph de-anonymizability quantification research, it is meaningful to develop anonymization technique-oriented quantification techniques. It is also a promising direction to systematically study

the tradeoff among graph anonymity, utility, and de-anonymity, which can also guide and facilitate graph anonymization research.

VII. Related Work

In this section, we discuss the related work with an emphasis of remarking on the differences between this work and surveys in existing literature.

In [177], Fung et al. made a comprehensive survey on privacy preserving data publishing techniques for relational data, which are inherently different graph data as we analyzed in Section III-C. In this paper, we mainly focus on systematically survey, analyze and evaluate graph data anonymization, de-anonymization, and de-anonymizability quantification techniques.

In [33], Zhou et al. conducted a brief survey on the anonymization techniques (before 2008) for privacy preserving publishing of social network data. In the survey, they mainly discussed two kinds of graph anonymization approaches: clustering-based approaches and graph modification approaches. In [178], Sharma also conducted a brief survey on graph data anonymization techniques. They mainly summarized two kinds of approaches: k-anonymity based techniques and randomization based techniques (e.g., Add/Del). In [179], Xu et al. studied the privacy preserving data mining techniques. They identified four types of users involved in data mining applications and for each type of users, they discussed the privacy concerns and the potential privacy preserving methods. Different from [33], [178], and [179], we give a much more comprehensive survey on graph anonymization techniques including six categories of approaches as well as a much more comprehensive utility analysis. In addition to anonymization techniques, we also systematically survey and analyze existing graph de-anonymization attacks, graph de-anonymizability quantification techniques, and the vulnerability of anonymization techniques against de-anonymization attacks. Furthermore, many new advances of graph anonymization techniques discussed in this paper are not included in [33], [178], and [179]. In summary, to the best of our knowledge, this is the first and most comprehensive work that systematically surveys, evaluates, and analyzes the 15 years' advances of graph data anonymization, de-anonymization, and de-anonymizability quantification research.

VIII. Conclusion

In this paper, we systematically summarize, classify, and analyze state-of-the-art graph data anonymization algorithms, structure-based de-anonymization attacks, and de-anonymizability quantification techniques. For existing graph data anonymization techniques, we classify them into six categories and analyze their performance with respect to 22 utility metrics. For existing de-anonymization attacks, we classify them into two categories and examine their performance with respect to scalability, practicability, robustness, etc. We also analyze the resistance of existing graph anonymization techniques against existing graph de-anonymizaiton attacks. For existing de-anonymizability quantifications, we classify them according to whether they consider seed information or not, and analyze them in terms of their soundness. Our analysis demonstrates that (i) most anonymization schemes can partially or conditionally preserve most graph utility while losing some application utility; and (ii) state-of-the-art anonymization schemes are vulnerable to several or all of the emerging structure-based de-anonymization attacks. The actual vulnerability of each anonymization algorithm depends on how much and which data utility it preserves. Based on our summarization and analysis, we discuss the research evolution, future directions, and challenges in the research area of graph data anonymization, de-anonymization, and de-anonymizability quantification.

Acknowledgment

The authors are very grateful to Weiqing Li, Xin Hu, Ting Wang, and Mudhakar Srivatsa for insightful discussions on graph anonymization and de-anonymization techniques.

This work was partly supported by the Provincial Key Research and Development Program of Zhejiang under No. 2016C01G2010916 and by the CCF-Tencent Open Research Fund under No. CCF-Tecent AGR20160109.

References

[1] A. Narayanan and V. Shmatikov. De-anonymizing social networks. *Proceedings of the 30th IEEE Symposium on Security and Privacy*, pages 173–187, 2009.

[2] S. Nilizadeh, A. Kapadia, and Y.-Y. Ahn. Community-enhanced de-anonymization of online social networks. *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security (CCS)*, pages 537–547, 2014.

[3] M. Srivatsa and M. Hicks. Deanonymizing mobility traces: Using social networks as a side-channel. *Proceedings of the 2012 ACM conference on Computer and communications security (CCS)*, pages 628–637, 2012.

[4] S. Ji, W. Li, M. Srivatsa, and R. Beyah. Structural data de-anonymization: Quantification, practice, and implications. *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security (CCS)*, pages 1040–1053, 2014.

[5] S. Ji, W. Li, M. Srivatsa, and R. Beyah. Structural data de-anonymization: Theory and practice. *IEEE/ACM Transactions on Networking (ToN)*, pages 1–14, 2016 (DOI: 10.1109/TNET.2016.2536479).

[6] S. Ji, W. Li, N. Gong, P. Mittal, and R. Beyah. On your social network de-anonymizablity: Quantification and large scale evaluation with seed knowledge. *Proceedings of the 2015 Network and Distributed System Security (NDSS) Symposium*, (99):1–15, 2015.

[7] S. Ji, W. Li, M. Srivatsa, and R. Beyah. Seed based de-anonymizability quantification of social networks. *IEEE Transactions on Information Forensics & Security (TIFS)*, 11(7):1398–1411, 2016.

[8] S. Ji, W. Li, P. Mittal, X. Hu, and R. Beyah. Secgraph: A uniform and open-source evaluation system for graph data anonymization and de-anonymization. *Proceedings of the 24th USENIX Security Symposium*, pages 303–318, 2015.

[9] X. Ying and X. Wu. Randomizing social networks: a spectrum preserving approach. *Proceedings of the SIAM International Conference on Data Mining (SDM)*, 8:739–750, 2008.

[10] B. Zhou and J. Pei. Preserving privacy in social networks against neighborhood attacks. *Proceedings of the IEEE 24th International Conference on Data Engineering (ICDE)*, pages 506–515, 2008.

[11] K. Liu and E. Terzi. Towards identity anonymization on graphs. *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 93–106, 2008.

[12] L. Zou, L. Chen, and M. T. Özsu. K-automorphism: A general framework for privacy preserving network publication. *Proceedings of the VLDB Endowment*, 2(1):946–957, 2009.

[13] J. Cheng, A. Fu, and J. Liu. K-isomorphism: Privacy preserving network publication against structural attacks. *Proceedings of the 2010 ACM SIGMOD International Conference on Management of data*, pages 459–470, 2010.

[14] M. Yuan, L. Chen, and P. Yu. Personalized privacy protection in social networks. *Proceedings of the VLDB Endowment*, 4(2):141–150, 2010.

[15] M. Hay, G. Miklau, D. Jensen, D. Towsley, and P. Weis. Resisting structural re-identification in anonymized social networks. *Proceedings of the VLDB Endowment*, 1(1):102–114, 2008.

[16] S. Bhagat, G. Cormode, B. Krishnamurthy, and D. Srivastava. Class-based graph anonymization for social network data. *Proceedings of the VLDB Endowment*, 2(1):766–777, 2009.

[17] B. Thompson and D. Yao. The union-split algorithm and cluster-based anonymization of social networks. *Proceedings of the 4th International Symposium on Information, Computer, and Communications Security (ASIACCS)*, pages 218–227, 2009.

[18] A. Sala, X. Zhao, C. Wilson, H. Zheng, and B. Zhao. Sharing graphs using differentially private graph models. *Proceedings of the 2011 ACM SIGCOMM conference on Internet measurement conference (IMC)*, pages 81–98, 2011.

[19] D. Proserpio, S. Goldberg, and F. McSherry. A workflow for differentially-private graph synthesis. *Proceedings of the 2012 ACM workshop on Workshop on online social networks*, pages 13–18, 2012.

[20] D. Proserpio, S. Goldberg, and F. McSherry. Calibrating data to sensitivity in private data analysis. *Proceedings of the 40th International Conference on Very Large Data Base (VLDB)*, 7(8):637–648, 2014.

[21] Y. Wang and X. Wu. Preserving differential privacy in degree-correlation based graph generation. *Transactions on data privacy (TDP)*, 6(2):127–145, 2013.

[22] Q. Xiao, R. Chen, and K. Tan. Differentially private network data release via structural inference. *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD)*, pages 911–920, 2014.

[23] P. Mittal, C. Papamanthou, and D. Song. Preserving link privacy in social network based systems. *Proceedings of the 20th Annual Network and Distributed System Security Symposium (NDSS)*, pages 1–15, 2013.

[24] Y. Liu, S. Ji, and P. Mittal. Smartwalk: Enhancing social network security via adaptive random walks. *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security (CCS)*, pages 492–503, 2016.

[25] L. Yartseva and M. Grossglauser. On the performance of percolation graph matching. *Proceedings of the first ACM conference on Online social networks*, pages 119–130, 2013.

[26] S. Ji, W. Li, M. Srivatsa, J. He, and R. Beyah. Structure based data de-anonymization of social networks and mobility traces. *Proceedings of the Information Security (ISC)*, pages 237–254, 2014.

[27] S. Ji, W. Li, M. Srivatsa, J. He, and R. Beyah. General graph data de-anonymization: From mobility traces to social networks. *ACM Transactions on Information and System Security (TISSEC)*, 18(4):1–29, 2016.

[28] N. Korula and S. Lattanzi. An efficient reconciliation algorithm for social networks. *Proceedings of the VLDB Endowment*, 7(5):377–388, 2014.

[29] P. Pedarsani, D. R. Figueiredo, and M. Grossglauser. A bayesian method for matching two similar graphs without seeds. *Proceedings of the Allerton Conference on Communication, Control, and Computing*, pages 1598–1607, 2013.

[30] P. Pedarsani and M. Grossglauser. On the privacy of anonymized networks. *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD)*, pages 1235–1243, 2011.

[31] S. Ji, W. Li, S. Yang, P. Mittal, and R. Beyah. On the relative de-anonymizability of graph data: Quantification and evaluation. *Proceedings of The 35th Annual IEEE International Conference on Computer Communications (INFOCOM)*, pages 1–9, 2016.

[32] C. Task and C. Clifton. A guide to differential privacy theory in social network analysis. *Proceedings of the 2012 International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pages 411–417, 2012.

[33] B. Zhou, J. Pei, and W.-S. Luk. A brief survey on anonymization techniques for privacy preserving publishing of social network data. *ACM SIGKDD Explorations Newsletter*, 10(2):12–22, 2008.

[34] Facebook. *https://www.facebook.com/*.

[35] Google+. *https://plus.google.com/*.

[36] Twitter. *https://twitter.com/twitter*.

[37] LinkedIn. *https://www.linkedin.com/*.

[38] Youtube. *https://www.youtube.com/*.

[39] LiveJournal. *http://www.livejournal.com/*.

[40] Orkut. *https://orkut.google.com/*.

[41] SlashDot. *http://slashdot.org/*.

[42] Pokec. *http://pokec.azet.sk/*.

[43] L. Tabourier, A. S. Libert, and R. Lambiotte. Predicting links in ego-networks using temporal information. *EPJ Data Science*, 5(1):1–26, 2016.

[44] T. Ma, J. Zhou, M. Tang, Y. Tian, A. Al-Dhelaan, M. Al-Rodhaan, and S. Lee. Social network and tag sources based augmenting collaborative recommender system. *IEICE transactions on Information and Systems*, (4):902–910, 2015.

[45] Q. Wang, Y. Zhang, X. Lu, Z. Wang, Z. Qin, and K. Ren. Real-time and spatio-temporal crowd-sourced social network data publishing with differential privacy. *IEEE Transactions on Dependable and Secure Computing (TDSC)*, (99):1–1, 2016 (DOI: 10.1109/TDSC.2016.2599873).

[46] J.-P. Onnela, J. Saramaki, J. Hyvonen, G. Szabo, D. Lazer, K. Kaski, J. Kertesz, and A.-L. Barabasi. Structure and tie strengths in mobile communication networks. *Proceedings of the National Academy of Sciences*, 104(18):7332–7336, 2007.

[47] A. A. Nanavati, S. Gurumurthy, G. Das, D. Chakraborty, K. Dasgupta, S. Mukherjea, and A. Joshi. On the structural properties of massive telecom call graphs: Findings and implications. *Proceedings of the 15th ACM international conference on Information and knowledge management (CIKM)*, pages 435–444, 2006.

[48] R. Lambiotte, V. D. Blondel, C. Kerchove, E. Huens, C. Prieur, Z. Smoreda, and P. V. Dooren. Geographical dispersal of mobile communication networks. *Physica A: Statistical Mechanics and its Applications*, 387(21):5317–5325, 2008.

[49] M. Kurucz, A. Benczúr, K Csalogány, and L. Lukács. Spectral clustering in telephone call graphs. *Proceedings of the 9th WebKDD and 1st SNA-KDD 2007 workshop on Web mining and social network analysis*, pages 82–91, 2007.

[50] J. Leskovec, J. Kleinberg, and C. Faloutsos. Graph evolution: Densification and shrinking diameters. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 1(1):2–2, 2007.

[51] J. Leskovec, K. Lang, A. Dasgupta, and M. Mahoney. Community structure in large networks: Natural cluster sizes and the absence of large well-defined clusters. *Internet Mathematics*, 6(1):29–123, 2009.

[52] B. Klimmt and Y. Yang. Introducing the enron corpus. *CEAS*, 2004.

[53] J. Leskovec, D. Huttenlocher, and J. Kleinberg. Signed networks in social media. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI)*, pages 1361–1370, 2010.

[54] J. Leskovec, D. Huttenlocher, and J. Kleinberg. Predicting positive and negative links in online social networks. *Proceedings of the 19th international conference on World wide web (WWW)*, pages 641–650, 2010.

[55] P. Guo, J. Wang, B. Li, and S. Lee. A variable threshold-value authentication architecture for wireless mesh networks. *Journal of Internet Technology*, 15(6):929–936, 2014.

[56] H. Pham, C. Shahabi, and Y. Liu. Ebm- an entropy-based model to infer social strength from spatiotemporal data. *Proceedings of the 2013 international conference on Management of data*, pages 265–276, 2013.

[57] Add Health. Deductive disclosure. *http://www.cpc.unc.edu/projects/addhealth/ data/dedisclosure*, 2008.

[58] *http://www.cpc.unc.edu/projects/addhealth*, 2008.

[59] P. Bearman, J. Moody, and K. Stovel. Chains of affection: The structure of adolescent romantic and sexual networks. *American Journal of Sociology*, 110(1):44–91, 2004.

[60] *http://www.informatik.uni-trier.de/∼ley/db/*.

[61] *http://arnetminer.org*.

[62] *http://snap.stanford.edu/data/index.html*.

[63] G. Aggarwal, T. Feder, K. Kenthapadi, R. Motwani, R. Panigrahy, D. Thomas, and A. Zhu. Approximation algorithms for k-anonymity. *Journal of Privacy Technology (JOPT)*, pages 1–18, 2005.

[64] J. Kleinberg J. Leskovec and C. Faloutsos. Graphs over time: Densification laws, shrinking diameters and possible explanations. *Proceedings of the 11th ACM SIGKDD international conference on Knowledge discovery in data mining (KDD)*, pages 177–187, 2005.

[65] J. Gehrke, P. Ginsparg, and J. M. Kleinberg. Overview of the 2003 kdd cup. *ACM SIGKDD Explorations Newsletter*, 5(2):149–151, 2003.

[66] *Google programming contest*, 2002.

[67] R. Albert, H. Jeong, and A.-L. Barabasi. Diameter of the world-wide web. *Nature*, 401(6749):130–131, 1999.

[68] M. Ripeanu, I. Foster, and A. Iamnitchi. Mapping the gnutella network: Properties of large-scale peer-to-peer systems and implications for system design. *IEEE Internet Computing Journal*, 6:50–57, 2002.

[69] *http://www.sigkdd.org/kddcup/index.php*.

[70] *https://blog.twitter.com/2014/introducing-twitter-data-grants*.

[71] *http://danzarrella.com/about-my-facebook-sharing-dataset-and-methodology#*.

[72] *https://www.kddcup2012.org/c/kddcup2012-track1*.

[73] *http://icwsm.org/2013/datasets/datasets/*.

[74] L. Backstrom, C. Dwork, and J. Kleinberg. Wherefore art thou r3579x? anonymized social networks, hidden patterns, and structural steganography. *Proceedings of the 16th international conference on World Wide Web*, pages 181–190, 2007.

[75] A. Narayanan, E. Shi, and B. Rubinstein. Link prediction by de-anonymization: How we won the kaggle social network challenge. *Proceedings of the The 2011 International Joint Conference on Neural Networks (IJCNN)*, pages 1825–1834, 2011.

[76] G. Wondracek, T. Holz, E. Kirda, and C. Kruegel. A practical attack to de-anonymize social network users. *Proceedings of the 2010 IEEE Symposium on Security and Privacy*, pages 223–238, 2010.

[77] K. Sharad and G. Danezis. De-anonymizing d4d datasets. *Proceedings of the Workshop on Hot Topics in Privacy Enhancing Technologies (PETS)*, pages 1–17, 2013.

[78] A. Korolova, R. Motwani, S. Nabar, and Y. Xu. Link privacy in social networks. *Proceedings of the 17th ACM conference on Information and knowledge management (CIKM)*, pages 289–298, 2008.

[79] A. Narayanan and V. Shmatikov. Robust de-anonymization of large sparse datasets. *Proceedings of the IEEE Symposium on Security and Privacy*, pages 111–125, 2008.

[80] G. J. Wills. Nicheworks - interactive visualization of very large graphs. *Journal of Computational and Graphical Statistics*, 8(2):190–212, 1999.

[81] B. Hayes. Connecting the dots: Can the tools of graph theory and social-network studies unravel the next big plot? *American Scientist*, 94(5):400–404, 2006.

[82] M. Clayton. Nsa data-mining 101: two 'top secret' programs and what they do. *http://www.csmonitor.com/USA/2013/0607/NSA-data-mining-101-two-top-secret-programs-and-what-they-do*.

[83] S. Perez. Twitter partners with ibm to bring social data to the enterprise. *http://techcrunch.com/2014/10/29/twitter-partners-with-ibm-to-bring-social-data-to-the-enterprise/, 2014*.

[84] *http://www.google.com/policies/privacy/*.

[85] *https://www.facebook.com/note.php?note_id=%20322194465300*.

[86] *https://twitter.com/privacy*.

[87] E. K. Lee, C. H. Chen, F. Pietz, and B. Benecke. Disease propagation analysis and mitigation strategies for effective mass dispensing. *AMIA annual symposium proceedings*, pages 427–427, 2010.

[88] *http://www.andrew.cmu.edu/user/rkoganti/realistic.html*.

[89] *http://www.slideshare.net/jlcaut/ebola-hemoragic-fever-propagation-in-a-modern-city-using-sir-model*.

[90] *https://www.data.gov/*.

[91] *http://www.cs.berkeley.edu/ stevgong/dataset.html*.

[92] *http://socialcomputing.asu.edu/pages/datasets*.

[93] A. Mislove, M. Marcon, K. Gummadi, P. Druschel, and B. Bhattacharjee. Measurement and analysis of online social networks. *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement (IMC)*, pages 29–42, 2007.

[94] R. Li and K. C.-C. Chang. Egonet-uiuc: A dataset for ego network research. *http://arxiv.org/abs/1309.4157*, 2013.

[95] *http://crawdad.cs.dartmouth.edu/*.

[96] *http://networkdata.ics.uci.edu/resources.php*.

[97] *http://www.casos.cs.cmu.edu/computational_tools/data2.php*.

[98] K. Tummarello. How 'data brokers' are striking gold. *http://thehill.com/policy/technology/207809-how-data-brokers-are-striking-gold*, 2015.

[99] L. Beckett. Everything we know about what data brokers know about you. *http://www.propublica.org/article/everything-we-know-about-what-data-brokers-know-about-you*, 2015.

[100] Data-Broker. The data brokers: Selling your personal information. *http://www.cbsnews.com/news/the-data-brokers-selling-your-personal-information/*, 2015.

[101] Z. Fu, X. Wu, C. Guan, X. Sun, and K. Ren. Towards efficient multi-keyword fuzzy search over encrypted outsourced data with accuracy improvement. *IEEE Transactions on Information Forensics & Security (TIFS)*, 11(12):2706–2716, 2016.

[102] Z. Xia, X. Wang, X. Sun, and Q. Wang. A secure and dynamic multi-keyword ranked search scheme over encrypted cloud data. *IEEE Transactions on Parallel and Distributed Systems (TPDS)*, 27(2):340–352, 2015.

[103] Z. Fu, K. Ren, J. Shu, X. Sun, and F. Huang. Enabling personalized search over encrypted outsourced data with efficiency improvement. *IEEE Transactions on Parallel and Distributed Systems (TPDS)*, 27(9):2546–2559, 2016.

[104] Z. Fu, X. Sun, Q. Liu, L. Zhou, and J. Shu. Achieving efficient cloud search services: Multi-keyword ranked search over encrypted cloud data supporting parallel computing. *IEICE Transactions on Communications*, (1):190–200, 2015.

[105] Y. Ren, J. Shen, J. Wang, J. Han, and S. Lee. Mutual verifiable provable data auditing in public cloud storage. *Journal of Internet Technology*, 16(2):317–323, 2015.

[106] Z. Xia, X. Wang, L. Zhang, Z. Qin, X. Sun, , and K. Ren. A privacy-preserving and copy-deterrence content-based image retrieval scheme in cloud computing. *IEEE Transactions on Information Forensics & Security (TIFS)*, 11(11):2594–2608, 2016.

[107] S. Hu, Q. Wang, J. Wang, Z. Qin, and K. Ren. Securing sift: Privacy-preserving outsourcing computation of feature extractions over encrypted image data. *IEEE Transactions on Image Processing*, 25:3411–3425, 2016.

[108] P. Samarati. Protecting respondents' identities in microdata release. *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, 13(6):1010–1027, 2001.

[109] L. Sweeney. k-anonymity: A model for protecting privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems (UFKS)*, 10(5):557–570, 2002.

[110] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam. ℓ-diversity: Privacy beyond k-anonymity. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 1(1):3–3, 2007.

[111] N. Li, T. Li, and S. Venkatasubramanian. t-closeness: Privacy beyond k-anonymity and ℓ-diversity. *Proceedings of the IEEE 23rd International Conference on Data Engineering (ICDE)*, pages 106–115, 2007.

[112] K. LeFevre, D. J. DeWitt, and R. Ramakrishnan. Incognito: Efficient full-domain k-anonymity. *Proceedings of the 2005 ACM SIGMOD international conference on Management of data*, pages 49–60, 2005.

[113] H. Park and K. Shim. Approximate algorithms for k-anonymity. *Proceedings of the 2007 ACM SIGMOD international conference on Management of data*, pages 67–78, 2007.

[114] T. M. Truta and B. Vinay. Privacy protection: p-sensitive k-anonymity property. *The 22nd International Conference on Data Engineering Workshops*, pages 1–10, 2006.

[115] R. Wong, J. Li, A. Fu, and K. Wang. (α, k)-anonymity: An enhanced k-anonymity model for privacy-preserving data publishing. *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD)*, pages 754–759, 2006.

[116] D. J. Martin, D. Kifer, A. Machanavajjhala, and J. Gehrke. Worst-case background knowledge for privacy-preserving data publishing. *Proceedings of the IEEE 23rd International Conference on Data Engineering (ICDE)*, pages 126–135, 2007.

[117] X. Xiao and Y. Tao. Anatomy: Simple and effective privacy preservation. *Proceedings of the 32nd international conference on Very large data bases (VLDB)*, pages 139–150, 2006.

[118] Q. Zhang, N. Koudas, D. Srivastava, and T. Yu. Aggregate query answering on anonymized tables. *Proceedings of the IEEE 23rd International Conference on Data Engineering (ICDE)*, pages 116–125, 2007.

[119] K. Wang and B. C. M. Fung. Anonymizing sequential releases. *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD)*, pages 414–423, 2006.

[120] M. E. Nergiz, M. Atzori, and C. Clifton. Hiding the presence of individuals from shared databases. *Proceedings of the 2007 ACM SIGMOD international conference on Management of data*, pages 665–676, 2007.

[121] X. Xiao and Y. Tao. m-invariance: Towards privacy preserving re-publication of dynamic datasets. *Proceedings of the 2007 ACM SIGMOD international conference on Management of data*, pages 689–700, 2007.

[122] X. Xiao and Y. Tao. Dynamic anonymization: Accurate statistical analysis with privacy preservation. *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 107–120, 2008.

[123] N. Mohammed, B. C. M. Fung, P. C. K. Hung, and C. Lee. Anonymizing healthcare data: A case study on the blood transfusion service. *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD)*, pages 1285–1294, 2009.

[124] N. Koudas, D. Srivastava, T. Yu, and Q. Zhang. Distribution-based microdata anonymization. *Proceedings of the VLDB Endowment*, 2(1):958–969, 2009.

[125] C. Chang, B. Thompson, H. Wang, and D. Yao. Towards publishing recommendation data with predictive anonymization. *Proceedings of the 5th ACM Symposium on Information, Computer and Communications Security (ASIACCS)*, pages 24–35, 2010.

[126] G. Aggarwal, T. Feder, K. Kenthapadi, S. Khuller, R. Panigrahy, D. Thomas, and A. Zhu. Achieving anonymity via clustering. *Proceedings of the twenty-fifth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems (PODS)*, pages 153–162, 2006.

[127] A. N. Mahmood, M. E. Kabir, and A. K. Mustafa. New multi-dimensional sorting based k-anonymity microaggregation for statistical disclosure control. *Proceedings of the Security and Privacy in Communication Networks (SecureComm)*, pages 256–272, 2012.

[128] K. Choromanski, T. Jebara, and K. Tang. Adaptive anonymity via b-matching. *Proceedings of the Advances in Neural Information Processing Systems (NIPS)*, pages 3192–3200, 2013.

[129] K. LeFevre, D. DeWitt, and R. Ramakrishnan. Workload-aware anonymization. *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD)*, pages 277–286, 2006.

[130] J. Xu, W. Wang, J. Pei, X. Wang, B. Shi, and A. Fu. Utility-based anonymization using local recording. *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 785–790, 2006.

[131] D. Kifer and J. Gehrke. Injecting utility into anonymized datasets. *Proceedings of the 2006 ACM SIGMOD international conference on Management of data*, pages 217–228, 2006.

[132] J. Brickell and V. Shmatikov. The cost of privacy: Destruction of data-mining utility in anonymized data publishing. *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD)*, pages 70–78, 2008.

[133] C. Dwork. Differential privacy. *Automata, languages and programming (ICALP)*, pages 1–12, 2006.

[134] C. Dwork, K. Kenthapadi, F. McSherry, I. Mironov, and M. Naor. Our data, ourselves: Privacy via distributed noise generation. *Advances in Cryptology-EUROCRYPT 2006*, pages 486–503, 2006.

[135] M. Hay, V. Rastogi, G. Miklau, and D. Suciu. Boosting the accuracy of differentially private histograms through consistency. *Proceedings of the VLDB Endowment (VLDB)*, 2006.

[136] N. Mohammed, R. Chen, B. Fung, and P. Yu. Differentially private data release for data mining. *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD)*, pages 493–501, 2011.

[137] G. Kellaris and S. Papadopoulos. Practical differential privacy via grouping and smoothing. *Proceedings of the VLDB Endowment (VLDB)*, 6(5):301–312, 2013.

[138] C. Li, M. Hay, G. Miklau, and Y. Wang. A data- and workload-aware algorithm for range queries under differential privacy. *Proceedings of the VLDB Endowment (VLDB)*, 7(5):341–352, 2014.

[139] W. Qardaji, W. Yang, and N. Li. Priview: Practical differentially private release of marginal contingency tables. *Proceedings of the 2014 ACM SIGMOD international conference on Management of data*, pages 1435–1446, 2014.

[140] F. McSherry and I. Mironov. Differentially private recommender systems: Building privacy into the netflix prize contenders. *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD)*, pages 627–636, 2009.

[141] J. Lee and C. Clifton. Differential identifiability. *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD)*, pages 1041–1049, 2012.

[142] N. Li, W. Qardaji, D. Su, Y. Wu, and W. Yang. Membership privacy: A unifying framework for privacy definitions. *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security (CCS)*, pages 889–900, 2013.

[143] N. Li, W. Qardaji, and D. Su. On sampling, anonymization, and differential privacy or, k-anonymization meets differential privacy. *Proceedings of the 7th ACM Symposium on Information, Computer and Communications Security (ASIACCS)*, pages 32–33, 2012.

[144] Y. He and J. F. Naughton. Anonymization of set-valued data via top-down, local generation. *Proceedings of the VLDB Endowment*, 2(1):934–945, 2009.

[145] M. Xue, P. Karras, C. Raïssi, J. Vaidya, and K. Tan. Anonymizing set-valued data by nonreciprocal recording. *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 1050–1058, 2012.

[146] M. Terrovitis, N. Mamoulis, and P. Kalnis. Privacy-preserving anonymization of set-valued data. *Proceedings of the VLDB Endowment*, 1(1):115–125, 2008.

[147] Y. Xu, K. Wang, A. Fu, and P. Yu. Anonymizing transaction databases for publication. *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD)*, pages 767–775, 2008.

[148] J. Cao, P. Karras, C. Raïssi, and K. Tan. ρ-uncertainty: Inference-proof transaction anonymization. *Proceedings of the VLDB Endowment*, 3(1):1033–1044, 2010.

[149] R. Chen, N. Mohammed, B. Fung, B. Desai, and L. Xiong. Publishing set-valued data via differential privacy. *Proceedings of the VLDB Endowment*, 4(11):1087–1098, 2011.

[150] R. Chen, B. Fung, B. Desai, and N. Sossou. Differentially private transit data publication: A case study on the montreal transportation system. *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD)*, pages 213–221, 2012.

[151] R. Chen, G. Acs, and C. Castelluccia. Differentially private sequential data publication via variable-length n-grams. *Proceedings of the 2012 ACM conference on Computer and communications security (CCS)*, pages 638–649, 2012.

[152] D. Goodin. *http://arstechnica.com/tech-policy/2014/06/poorly-anonymized-logs-reveal-nyc-cab-drivers-detailed-whereabouts/*.

[153] C. Liu, S. Chakraborty, and P. Mittal. Dependence makes you vulnerable: Differential privacy under dependent tuples. *NDSS*, pages 1–15, 2016.

[154] C. Liu and P. Mittal. Linkmirage: Enabling privacy-preserving analytics on social relationships. *NDSS*, pages 1–15, 2016.

[155] F. Beato, M. Conti, and B. Preneel. Friend in the middle (fim): Tackling de-anonymization in social networks. *Fifth International Workshop on SECurity and SOCial Networking*, pages 279–284, 2013.

[156] F. Beato, M. Conti, B. Preneel, and D. Vettore. Virtualfriendship: Hiding interactions on online social networks. *IEEE Conference on Communications and Network Security (CNS)*, pages 328–336, 2014.

[157] R. Albert, H. Jeong, and A.-L. Barabasi. Error and attack tolerance of complex networks. *Nature*, 406(6794):378–382, 2000.

[158] D. J. Watts and S. H. Strogatz. Collective dynamics of 'small-world' networks. *Nature*, 393(6684):440–442, 1998.

[159] M. E. J. Newman. Networks: An introduction. *Oxford University Press*, 2010.

[160] K. Henderson, B. Gallagher, T. Eliassi-Rad, H. Tong, S. Basu, L. Akoglu, D. Koutra, L. Li, and C. Faloutsos. Rolx: Structural role extraction & mining in large graphs. *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD)*, pages 1231–1239, 2012.

[161] S. Garriss, M. Kaminsky, M. J. Freedman, B. Karp, D. Mazières, and H. Yu. Re: Reliable email. *Proceedings of the 3rd conference on Networked Systems Design & Implementation*, 3:22–22, 2006.

[162] A. Goyal, F. Bonchi, and L. V. S. Lakshmanan. A data-based approach to social influence maximization. *Proceedings of the VLDB Endowment*, 5(1):73–84, 2011.

[163] J. He, S. Ji, R. Beyah, and Z. Cai. Minimum-sized influential node set selection for social networks under the independent cascade model. *Proceedings of the 15th ACM international symposium on Mobile ad hoc networking and computing (Mobihoc)*, pages 93–102, 2014.

[164] J Yang and J. Leskovec. Overlapping community detection at scale: A nonnegative matrix factorization. *Proceedings of the sixth ACM international conference on Web search and data mining (WSDM)*, pages 587–596, 2013.

[165] B. Feil, R. Thiraporn, and P. Stamp. Revealing intricate properties of communities in the bipartite structure of online social networks. *IEEE Ninth International Conference on Research Challenges in Information Science*, pages 227–231, 2015.

[166] S. Marti, P. Ganesan, and H. Garcia-Molina. Sprout: P2p routing with social networks. *Proceedings of the Current Trends in Database Technology-EDBT 2004 Workshops*, pages 425–435, 2005.

[167] H. Yu, P. B. Gibbons, M. Kaminsky, and F. Xiao. Sybillimit: A near-optimal social network defense against sybil attacks. *Proceedings of the IEEE Symposium on Security and Privacy (S&P)*, pages 3–17, 2008.

[168] H. Yu, C. Shi, M. Kaminsky, P. B. Gibbons, and F. Xiao. Dsybil: Optimal sybil-resistance for recommendation systems. *Proceedings of the 30th IEEE Symposium on Security and Privacy (S&P)*, pages 283–298, 2009.

[169] L. Lakshmanan, R. Ng, and G. Ramesh. To do or not to do: The dilemma of disclosing anonymized data. *Proceedings of the 2005 ACM SIGMOD international conference on Management of data*, pages 61–72, 2005.

[170] G. Cormode, D. Srivastava, N. Li, and T. Li. Minimizing minimality and maximizing utility: Analyzing method-based attacks on

anonymized data. *Proceedings of the VLDB Endowment*, 3(1):1045–1056, 2010.

[171] M. Nanavati, N. Taylor, W. Aiello, and A. Warfield. Herbert west - deanonymizer. *Proceedings of the 6th USENIX conference on Hot topics in security*, pages 6–6, 2011.

[172] G. Cormode. Personal privacy vs population privacy: Learning to attack anonymization. *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD)*, pages 1253–1261, 2011.

[173] M. Merener. Theoretical results on de-anonymization via linkage attacks. *Transactions on Data Privacy (TDP)*, 5(2):377–402, 2012.

[174] J. Unnikrishnan and F. M. Naini. De-anonymizing private data by matching statistics. *Proceedings of the Allerton Conference on Communication, Control, and Computing*, pages 1616–1623, 2013.

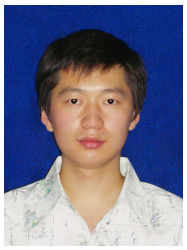
[175] J. Qian, X.-Y. Li, C. Zhang, and L. Chen. De-anonymizing social networks and inferring private attributes using knowledge graphs. *IEEE INFOCOM*, pages 1–9, 2016.

[176] A. Mohaisen and Y. Kim. Dynamix: Anonymity on dynamic social structures. *ASIA CCS*, pages 167–172, 2013.

[177] B. C. M. Fung, K. Wang, R. Chen, and P. S. Yu. Privacy-preserving data publishing: A survey of recent developments. *ACM Computing Surveys*, 42(4):1–53, 2010.

[178] S. Sharma, P. Gupta, and V. Bhatnagar. Anonymisation in social network: A literature survey and classification. *International Journal of Social Network Mining*, 1(1):51–66, 2012.

[179] L. Xu, C. Jiang, J. Wang, J. Yuan, and Y. Ren. Information security in big data: Privacy and data mining. *IEEE Access*, 2:1149–1176, 2014.



Shouling Ji is a ZJU 100-Young Professor in the College of Computer Science and Technology at Zhejiang University and a Research Faculty in the School of Electrical and Computer Engineering at Georgia Institute of Technology. He received a Ph.D. in Electrical and Computer Engineering from Georgia Institute of Technology, a Ph.D. in Computer Science from Georgia State University, and B.S. (with Honors) and M.S. degrees both in Computer Science from Heilongjiang University. His current research interests include Big Data Security and Privacy, Big Data Driven Security and Privacy, Differential Privacy, Password Security, and Machine Learning Security and Privacy. He also has interests in Graph Theory and Algorithms and Wireless Networks. He is a member of IEEE and ACM and was the Membership Chair of the IEEE Student Branch at Georgia State (2012-2013).

Prateek Mittal is an assistant professor in the Department of Electrical Engineering at Princeton University. His research interests include the domains of privacy enhancing technologies, trustworthy social systems, and Internet/network security. His work has influenced the design of several widely used anonymity systems, and he is the recipient of several awards, including an ACM CCS outstanding paper. He served as the program co-chair for the HotPETs workshop in 2013 and 2014. Prior to joining Princeton University, he was a postdoctoral scholar at University of California, Berkeley. He obtained his Ph.D. in Electrical and Computer Engineering from University of Illinois at Urbana-Champaign in 2012

Raheem Beyah is the Motorola Foundation Professor and Associate Chair in the School of Electrical and Computer Engineering at Georgia Tech, where he leads the Communications Assurance and Performance Group (CAP) and is a member of the Communications Systems Center (CSC). Prior to returning to Georgia Tech, Dr. Beyah was an Assistant Professor in the Department of Computer Science at Georgia State University, a research faculty member with the Georgia Tech CSC, and a consultant in Andersen Consulting's (now Accenture) Network Solutions Group. He received his Bachelor of Science in Electrical Engineering from North Carolina A&T State University in 1998. He received his Masters and Ph.D. in Electrical and Computer Engineering from Georgia Tech in 1999 and 2003, respectively. His research interests include network security, wireless networks, network traffic characterization and performance, and critical infrastructure security. He received the National Science Foundation CAREER award in 2009 and was selected for DARPA's Computer Science Study Panel in 2010. He is a member of AAAS and ASEE, is a lifetime member of NSBE, and is a senior member of ACM and IEEE.