Hindawi
Security and Communication Networks
Volume 2017, Article ID 8408253, 14 pages
https://doi.org/10.1155/2017/8408253

Research Article

An Edge Correlation Based Differentially Private Network Data Release Method

Junling Lu,[1,2] Zhipeng Cai,[3] Xiaoming Wang,[1,2] Lichen Zhang,[1,2] and Zhuojun Duan[3]

[1]*Key Laboratory for Modern Teaching Technology, Ministry of Education, Xi'an 710062, China*
[2]*School of Computer Science, Shaanxi Normal University, Xi'an 710119, China*
[3]*Department of Computer Science, Georgia State University, Atlanta, GA 30303, USA*

Correspondence should be addressed to Zhipeng Cai; zcai@gsu.edu

Received 16 August 2017; Accepted 16 October 2017; Published 13 November 2017

Academic Editor: Houbing Song

Differential privacy (DP) provides a rigorous and provable privacy guarantee and assumes adversaries' arbitrary background knowledge, which makes it distinct from prior work in privacy preserving. However, DP cannot achieve claimed privacy guarantees over datasets with correlated tuples. Aiming to protect whether two individuals have a close relationship in a correlated dataset corresponding to a weighted network, we propose a differentially private network data release method, based on edge correlation, to gain the tradeoff between privacy and utility. Specifically, we first extracted the Edge Profile (PF) of an edge from a graph, which is transformed from a raw correlated dataset. Then, edge correlation is defined based on the PFs of both edges via Jenson-Shannon Divergence (JS-Divergence). Secondly, we transform a raw weighted dataset into an indicated dataset by adopting a weight threshold, to satisfy specific real need and decrease query sensitivity. Furthermore, we propose ϵ-correlated edge differential privacy (CEDP), by combining the correlation analysis and the correlated parameter with traditional DP. Finally, we propose network data release (NDR) algorithm based on the ϵ-CEDP model and discuss its privacy and utility. Extensive experiments over real and synthetic network datasets show the proposed releasing method provides better utilities while maintaining privacy guarantee.

1. Introduction

Recently, social networking such as cooperation networks, online/mobile social networks, and software defined vehicular network [1] is becoming increasingly prevalent. Accompanied with the growth of the networks, mass of network data is released for analytical decisions or scientific researches. However, direct publication of these data, including sensitive information, leads to privacy leakage of individuals. For example, whether two individuals in a social network have a close relationship may be expected to be kept a secret. Therefore, privacy concerns have been raised in increasingly emerging technologies [2–9].

In general, a dataset corresponding to such a network, usually modeled as a graph, is considered as correlated data; that is, tuples in this dataset are dependent. Clearly, privacy preserving in such correlated settings is more difficult because an adversary can infer the relationship of two individuals from their associated friends. Accordingly, our concern is preventing, whether the relationship of two individuals appears in a network dataset, from being unveiled.

Differential privacy (DP), a privacy preserving model originated from statistical database, has currently drawn considerable attentions in research communities [10–17] due to (i) its rigorous and provable privacy guarantee and (ii) its assumption of adversaries' arbitrary background knowledge. However, DP actually assumes that the tuples in databases are independent [18]. In other words, DP cannot provide claimed privacy guarantees over correlated (nonindependent) data [19]. Therefore, the application of DP over correlated data is a challenge, and how to achieve a differentially private correlated data release method deserves to be further explored.

The focus of our work is on hiding the affinity degree of two individuals in a correlated dataset corresponding to a weighted network, that is, protecting whether the affinity degree of two individuals exceeds a given weight threshold, in a differentially private manner. Toward this end, we first transform a weighted network dataset into a corresponding

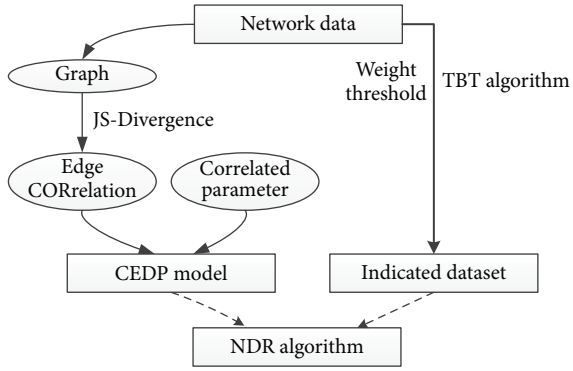


Figure 1: The framework of our solution.

weighted graph and define the correlation of both edges via Jenson-Shannon Divergence (JS-Divergence). For satisfying specific query need in spite of some utility loss, we utilize Threshold Based Transformation (TBT) algorithm to transform a weighted dataset, by adopting a weight threshold, into an indicated dataset, which also decreases query sensitivity. Finally, we present the notion of ϵ-correlated edge differential privacy (CEDP), by combining the correlation analysis and the correlated parameter, that is, the maximal number of correlated tuples, with traditional DP, and design differentially private network data release (NDR) algorithm to obtain better utilities while maintaining DP guarantee. Experimental results over real and synthetic network datasets also show the advantages of the proposed method. The framework of our solution is shown in Figure 1.

The contributions of our work are as follows.

First, we extract the Edge Profile (PF) vectors of edges in a weighted graph corresponding to a network dataset and then define the correlation of both edges via JS-Divergence. The inferred correlation analysis is more reasonable for datasets corresponding to such networks, since the typical Pearson correlation coefficient assumes that sample data follows normal distribution; however, the degree and weight distributions in such networks are often not so.

Second, we propose the ϵ-CEDP model based on our result of correlation analysis and the introduction of the correlated parameter, which makes DP over correlated datasets applicable and flexible. Furthermore, the NDR algorithm, based on correlated sensitivity and Laplace mechanism, is proposed, which also satisfies ϵ-CEDP and achieves the tradeoff of privacy and utility.

Third, we utilize TBT algorithm to transform a raw weighted dataset into an indicated dataset; that is, a weight value is equal to 1 or 0, by adopting a weight threshold, to satisfy specific real need and decrease query sensitivity. Admittedly, some utility loss exists in such transformation. However, many queries in real world only need Boolean values indicated by one and zero instead of accurate numeric answers. Therefore, this solution provides a feasible way for decreasing query sensitivity while maintaining real query need.

The rest of this paper is organized as follows. Section 2 discusses related literature. Section 3 provides the preliminaries. In Section 4, correlation analysis of both edges in a weighted graph is presented, and the ϵ-CEDP model and sensitivity calculation are proposed. Furthermore, a differentially private NDR algorithm, including TBT algorithm, to obtain the tradeoff between privacy and utility over correlated data is proposed in Section 5. The extensive experiments are illustrated in Section 6. Finally, Section 7 concludes the paper.

2. Related Work

Compared with previous works in privacy preserving, DP proposed by Dwork [20] provides a probabilistic formulation, which represents that adversaries learn little from both databases differing in one tuple even if adversaries know about all tuples except the target one. In other words, the inference abilities of adversaries about the presence or absence of a tuple are bounded regardless of adversaries' knowledge; that is, the presence or absence of a tuple is probabilistically indistinguishable for adversaries.

Currently, DP has drawn much attention in privacy preserving work needed in many fields. Wang et al. [10] considered a unified privacy distortion framework, where the distortion is defined to be the expected Hamming distance between the input and output databases, and investigated the relation between three different notions of privacy: identifiability, differential privacy, and mutual-information privacy. To provide personalized recommendation in big data resulting from social networks and maintain user privacy, a cloud-assisted differentially private video recommendation system based on distributed online learning was proposed [11]. The work in [12] proposed a new privacy preserving smart metering scheme for smart grid, which supports data aggregation, differential privacy, fault tolerance, and range-based filtering simultaneously. To et al. [13] introduced a novel privacy-aware framework for spatial crowdsourcing, which enables the participation of workers without compromising their location privacy. Focusing on the privacy protection of sensitive information in body area networks, the authors in [14, 15] proposed different privacy preserving schemes, based on differential privacy model, via a tree structure and dynamic noise thresholds, respectively. The work in [16] proposed a novel differentially private frequent sequence mining algorithm by leveraging a sampling-based candidate pruning technique, which satisfies ϵ-differential privacy and can privately find frequent sequences with high accuracy. In order to protect users' privacy in ridesharing services, a jointly differentially private scheduling protocol has been proposed [17], which aims to protect riders' location information and minimize the total additional vehicle mileage in the ridesharing system.

However, existing works have found that DP provides weaker privacy guarantee over nonindependent data; that is, DP needs more noise added to the output query result to cancel out the impact of correlations among tuples on privacy guarantee. Undoubtedly, how to analyze correlations among tuples and apply them into DP are desired to be further explored. For example, Kifer and Machanavajjhala [19] first

explicitly doubted the privacy guarantee of DP in correlated settings, for example, social networks, and then adopted the subsequently proposed privacy framework, that is, Pufferfish, to formalize and prove that DP assumes independence between tuples [18]. Inspired by the Pufferfish framework, Blowfish privacy [21] was proposed to achieve the tradeoff between privacy and utility using policies specifying secrets and constraints. Similarly, the authors in [22] proposed Bayesian DP to evaluate the level of private information leakage even when data is correlated and prior knowledge is incomplete. The work in [23] regarded the correlation among tuples as complete correlation and multiplied the query sensitivity with the number of correlated tuples in publishing correlated network data, which leaves room for fine-grained correlation analysis in the following work. Aiming to decrease the noise amount, Zhu et al. [24] depicted the correlation between tuples via Pearson correlation coefficient, including complete correlation, partial correlation, and independence. Liu et al. [25] inferred the dependence coefficient, distributed in interval $[0, 1]$, to evaluate the probabilistic correlation between two tuples in a more fine-grained manner, thus reducing the query sensitivity which results in less noise. Considering temporal correlations of a moving user's locations, the work in [26] leveraged a hidden Markov model to establish a location set and proposed a variant of DP to protect location privacy. Wu et al. [27] proposed the definition of correlated differential privacy to evaluate the real privacy level of a single dataset influenced by the other datasets when multiple datasets are correlated. The work in [28] formalized the privacy preservation problem to an optimization problem by modeling the temporal correlations among contexts and further proposed an efficient context-aware privacy preserving algorithm. Cao et al. [29] modeled the temporal correlations using Markov model and investigated the privacy leakage of a traditional DP mechanism under temporal correlations in the context of continuous data release. The work in [30] quantified the location correlation between two users through the similarity measurement of two hidden Markov models and applied differential privacy via private candidate sets to achieve the multiuser location correlation protection.

As seen from the above discussions, correlation analysis plays an important role in privacy preserving mechanisms, which directly influences the tradeoff between privacy protection and service utility. Obviously, the more accurate the correlation analysis, the better the balance of both aspects. Therefore, we attribute the underestimated privacy guarantee of DP over correlated data to the lack of data knowledge, and our work starts from data correlation analysis.

In this paper, we focus on correlated datasets corresponding to weighted cooperation networks. Different from the existing methods of correlation analysis, for example, simple multiplication in [23], Pearson correlation coefficient in [24], and the maximal information coefficient in [31], we extract the PF vectors of edges in a weighted graph corresponding to a correlated dataset and then define the correlation of both edges via JS-Divergence, which is more accurate and reasonable. Specifically, the work in [23] assumes both tuples are completely correlated; however, our proposed correlation results lie in interval $[0, 1]$ representing multiple correlation including complete correlation. In addition, the work in [24] assumes sample data follows normal distribution, while our method is not the case. Also, the maximal information coefficient proposed in [31] satisfies two heuristic properties including generality and equitability, and we will consider it in our future work.

3. Preliminaries

3.1. Differential Privacy. Differential privacy provides the privacy guarantee for an individual in the probabilistic sense [20]. It is defined as follows.

Definition 1 (ϵ-differential privacy). A randomized mechanism $\mathscr{A}$ satisfies ϵ-differential privacy if, for any pair of databases D and D' differing in only one tuple and for any output $S \in O(\mathscr{A})$ representing the possible output set of $\mathscr{A}$,

$$\Pr\left[\mathscr{A}(D) = S\right] \leq \exp(\epsilon) \cdot \Pr\left[\mathscr{A}\left(D'\right) = S\right], \tag{1}$$

where ϵ is the privacy budget depicting the probabilistic difference between the same outputs of $\mathscr{A}$ over D and D'.

Generally, DP is achieved via two mechanisms: Laplace mechanism [32] and exponential mechanism [33]. Both mechanisms include a concept of global sensitivity [20], which reveals DP's preferable choice of protecting the extreme case.

Definition 2 (global sensitivity). For any query function $f : D \to \mathbb{R}^d$, where D is a dataset and $\mathbb{R}^d$ is a d-dimension real-valued vector, the global sensitivity of f is defined as

$$\Delta f = \max_{D,D'} \left\| f(D) - \mathrm{f}\left(D'\right) \right\|_1, \tag{2}$$

where D and D' denote any pair of databases differing in only one tuple and $\| \cdot \|_1$ denotes l_1 norm.

Laplace mechanism, used in this paper, is formally presented as follows.

Theorem 3 (Laplace mechanism). *Given any query function $f : D \to \mathbb{R}^d$, where D is a dataset and $\mathbb{R}^d$ is a d-dimension real-valued vector, the global sensitivity Δf of f, and privacy budget ϵ, a randomized mechanism $\mathscr{A}$*

$$\mathscr{A}(D) = f(D) + Laplace\left(\frac{\Delta f}{\epsilon}\right) \tag{3}$$

provides the ϵ-differential privacy, where Laplace$(\cdot)$ denotes Laplace noise.

3.2. Weighted Adjacency Matrix. In this paper, we model a correlated dataset as a weighted undirected simple graph $G = (V, E, W)$, where $V = \{v_1, \ldots, v_n\}$ is the set of vertices and $n = |V|$ is the number of vertices, $E = \{e_{ij}\}$ is the set of edges and $e_{ij} = (v_i, v_j)$, $v_i \in V$, $i = 1, \ldots, n$, $v_j \in V$, $j = 1, \ldots, n$, and $W = \{w_{ij}\}$ is the set of weights where weight w_{ij} corresponds

Table 1: A raw weighted dataset.

V_i	V_j	w_{ij}
1	2	2
2	3	4
2	4	8
2	5	1
4	5	5
4	6	3

Note. V_i and V_j denote two individuals, and weight w_{ij} denotes the relation strength between them.

with edge e_{ij}. Then, the weighted adjacency matrix A^w of G can be denoted as

$$A_{ij}^w = \begin{cases} w_{ij}, & e_{ij} \in E \\ 0, & \text{otherwise}, \end{cases} \tag{4}$$

where w_{ij} represents the affinity degree between two individuals. Obviously, the weighted adjacency matrix A_{ij}^w is symmetric.

Example 4. Suppose a raw weighted dataset D_W is listed in Table 1. Then, the corresponding weighted adjacency matrix A^w of D_W can be denoted as

$$\begin{pmatrix} 0 & 2 & 0 & 0 & 0 & 0 \\ 2 & 0 & 4 & 8 & 1 & 0 \\ 0 & 4 & 0 & 0 & 0 & 0 \\ 0 & 8 & 0 & 0 & 5 & 3 \\ 0 & 1 & 0 & 5 & 0 & 0 \\ 0 & 0 & 0 & 3 & 0 & 0 \end{pmatrix}. \tag{5}$$

3.3. Correlation Metric. Motivated by the entropy in information theory, we adopt JS-Divergence, inferred from Kullback-Leibler Divergence (KL-Divergence) [23], to depict the difference of two distributions, which can be transformed to depict the correlation of two tuples in a correlated dataset.

Definition 5 (KL-Divergence). Suppose $P = \{p_1, p_2, \ldots, p_n\}$ and $Q = \{q_1, q_2, \ldots, q_n\}$ are the probability distributions of random variables $X = \{x_1, x_2, \ldots, x_n\}$ and $Y = \{y_1, y_2, \ldots, y_n\}$; then the KL-Divergence of P and Q is defined as follows:

$$\mathrm{KLD}(P \parallel Q) = \sum_{i=1}^{n} p_i \ln\left(\frac{p_i}{q_i}\right). \tag{6}$$

Here $0 \log 0 = 0$ is required. Based on KL-Divergence, we can obtain JS-Divergence as follows.

Definition 6 (JS-Divergence). Suppose $P = \{p_1, p_2, \ldots, p_n\}$ and $Q = \{q_1, q_2, \ldots, q_n\}$ are the probability distributions of random variables $X = \{x_1, x_2, \ldots, x_n\}$ and $Y = \{y_1, y_2, \ldots, y_n\}$, and $M = (1/2)(P + Q)$; then the JS-Divergence of P and Q is defined as follows:

$$\begin{aligned} \mathrm{JSD}(P \parallel Q) &= \frac{1}{2}\left(\mathrm{KLD}(P \parallel M) + \mathrm{KLD}(Q \parallel M)\right) \\ &= \frac{1}{2}\sum_{i=1}^{n}\left(p_i \ln\frac{2p_i}{p_i + q_i} + q_i \ln\frac{2q_i}{p_i + q_i}\right). \end{aligned} \tag{7}$$

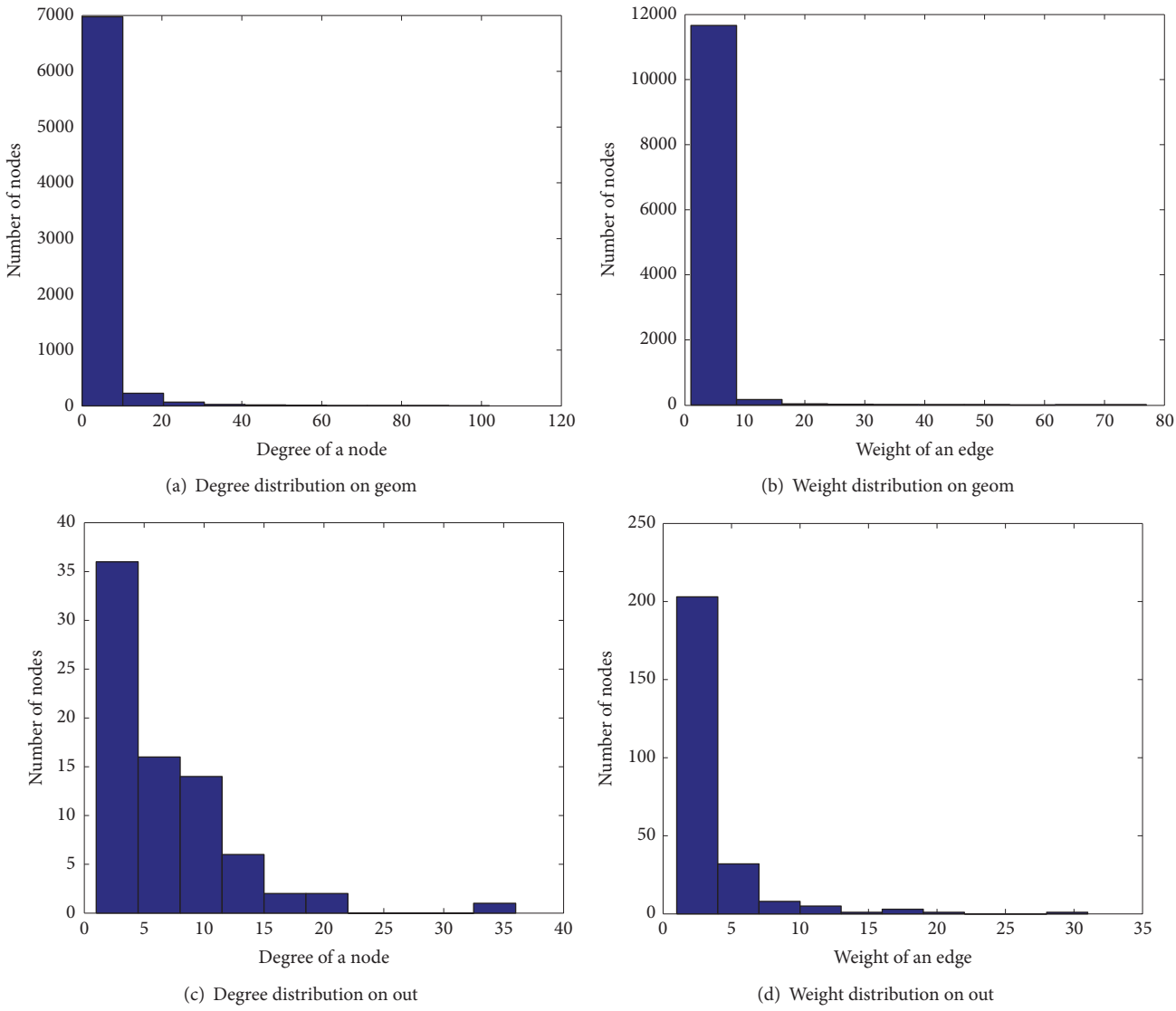
4. Correlation Analysis of Weighted Edges

In this section, we first discuss how to define the correlation of both edges in a weighted graph corresponding to a network dataset and then introduce why and how we conduct dataset transformation based on a given weight threshold. Finally, we define the ϵ-CEDP model and calculate the correlated sensitivity for smaller added noise.

4.1. Correlation Definition. For achieving the correlation of tuples in a raw weighted dataset D_W, we first obtain a weighted graph G, whose weighted adjacency matrix is denoted by A^w. Then, the correlation problem is changed to seeking the correlation of edges in G. To this end, we first describe the PF vector of a weighted edge from the perspectives of relational strength and network structure and then define the correlation of both edges via JS-Divergence instead of Pearson correlation coefficient.

For a weighted edge e_{ij}, suppose $V_i = \{v \mid (v, v_i) \in E, v \neq v_i\}$ represents the set of vertices connected with v_i and $V_j = \{v \mid (v, v_j) \in E, v \neq v_j\}$ represents the set of vertices connected with v_j; we extract the PF vector of e_{ij}, denoted by $\mathrm{PF}(e_{ij})$, from the perspectives of relational strength and network structure simultaneously. Specifically, we obtain $w_{ij}/\max_{e_{ij} \in E}(w_{ij}) \in [0, 1]$ from the global weights of all edges. In addition, we get $w_{ij}/\sum_{v \in V_i} w_{iv} \in [0, 1]$ and $w_{ij}/\sum_{v \in V_j} w_{jv} \in [0, 1]$ from the local weights of edge e_{ij}. On the other hand, similar to the representation of relational strength, $\mathrm{Deg}(v_i)/\max_{v \in V}(\mathrm{Deg}(v)) \in [0, 1]$ and $\mathrm{Deg}(v_j)/\max_{v \in V}(\mathrm{Deg}(v)) \in [0, 1]$ are constructed, by introducing the node degree $\mathrm{Deg}(\cdot)$, to depict the global active degree for both vertices of edge e_{ij}. Also, $|V_i \cap V_j|/|V_i \cup V_j| \in [0, 1]$ is adopted, via the set similarity, to depict the ratio of the number of common vertices connecting v_i and v_j to that of V_i and V_j. Meanwhile, $\mathrm{Deg}(v_i)/\sum_{v \in V_i} \mathrm{Deg}(v) \in [0, 1]$ and $\mathrm{Deg}(v_j)/\sum_{v \in V_j} \mathrm{Deg}(v) \in [0, 1]$ are used to depict the local active degree for both vertices of edge e_{ij}. Combining the above factors, we obtain $\mathrm{PF}(e_{ij})$ as follows:

$$\begin{aligned} \mathrm{PF}\left(e_{ij}\right) = \Bigg(& \frac{w_{ij}}{\max_{e_{ij} \in E}\left(w_{ij}\right)}, \frac{w_{ij}}{\sum_{v \in V_i} w_{iv}}, \frac{w_{ij}}{\sum_{v \in V_j} w_{jv}}, \\ & \frac{\mathrm{Deg}\left(v_i\right)}{\max_{v \in V}\left(\mathrm{Deg}\left(v\right)\right)}, \frac{\mathrm{Deg}\left(v_j\right)}{\max_{v \in V}\left(\mathrm{Deg}\left(v\right)\right)}, \frac{\left|V_i \cap V_j\right|}{\left|V_i \cup V_j\right|}, \\ & \frac{\mathrm{Deg}\left(v_i\right)}{\sum_{v \in V_i} \mathrm{Deg}\left(v\right)}, \frac{\mathrm{Deg}\left(v_j\right)}{\sum_{v \in V_j} \mathrm{Deg}\left(v\right)} \Bigg). \end{aligned} \tag{8}$$



(a) Degree distribution on geom

(b) Weight distribution on geom

(c) Degree distribution on out

(d) Weight distribution on out

Figure 2: Degree distribution and weight one on network data.

Similarly, for any other edge e_{mn}, according to (8), we define PF(e_{mn}) as follows:

$$\mathrm{PF}(e_{mn}) = \left(\frac{w_{mn}}{\max_{e_{mn}\in E}(w_{mn})}, \frac{w_{mn}}{\sum_{v\in V_m} w_{mv}}, \frac{w_{mn}}{\sum_{v\in V_n} w_{nv}}, \right.$$
$$\frac{\mathrm{Deg}(v_m)}{\max_{v\in V}(\mathrm{Deg}(v))}, \frac{\mathrm{Deg}(v_n)}{\max_{v\in V}(\mathrm{Deg}(v))}, \frac{|V_m \cap V_n|}{|V_m \cup V_n|}, \quad (9)$$
$$\left. \frac{\mathrm{Deg}(v_m)}{\sum_{v\in V_m}\mathrm{Deg}(v)}, \frac{\mathrm{Deg}(v_n)}{\sum_{v\in V_n}\mathrm{Deg}(v)} \right).$$

Note that Pearson correlation coefficient assumes that sample data follows normal distribution. However, in social networks, the weight and degree distributions do not follow such distribution, which is also verified by our experiments shown in Figure 2, where geom and out are the abbreviations of geom.net [34–36] and out.moreno_lesmis_lesmis [37–39] for simplicity. Specifically, (i) geom.net is the authors collaboration network in Computational Geometry based on the file geombib.bib, and the reduced simple network contains 7343 vertices and 11898 edges. Two authors are linked with an edge, iff they wrote a common work. The value of an edge is the number of common works. (ii) out.moreno_lesmis_lesmis is the characters cooccurrences network in Victor Hugo's novel "Les Misérables," and it contains 77 vertices and 254 edges. A node represents a character and an edge between two nodes shows that these two characters appeared in the same chapter of the book. The weight of each link indicates how often such a coappearance occurred.

Since our constructed PF vectors of edges do not satisfy the assumption of normal distribution, we adopt JS-Divergence, instead of Pearson correlation coefficient, to measure the CORrelation (COR) of any two edges e_{ij}, e_{mn} in a weighted graph. To this end, we normalize PF(e_{ij}) and

$\mathrm{PF}(e_{mn})$ as $\mathrm{PN}(e_{ij})$ and $\mathrm{PN}(e_{mn})$, which are two probability distributions. Therefore, we have

$$\mathrm{PN}\left(e_{ij}\right) = \frac{\mathrm{PF}\left(e_{ij}\right)}{S_{ij}}, \tag{10}$$

where $S_{ij} = w_{ij}/\max_{e_{ij}\in E}(w_{ij}) + w_{ij}/\sum_{v\in V_i} w_{iv} + w_{ij}/\sum_{v\in V_j} w_{jv} + \mathrm{Deg}(v_i)/\max_{v\in V}(\mathrm{Deg}(v)) + \mathrm{Deg}(v_j)/\max_{v\in V}(\mathrm{Deg}(v)) + |V_i \cap V_j|/|V_i \cup V_j| + \mathrm{Deg}(v_i)/\sum_{v\in V_i}\mathrm{Deg}(v) + \mathrm{Deg}(v_j)/\sum_{v\in V_j}\mathrm{Deg}(v)$ and

$$\mathrm{PN}\left(e_{mn}\right) = \frac{\mathrm{PF}\left(e_{mn}\right)}{S_{mn}}, \tag{11}$$

where $S_{mn} = w_{mn}/\max_{e_{mn}\in E}(w_{mn}) + w_{mn}/\sum_{v\in V_m} w_{mv} + w_{mn}/\sum_{v\in V_n} w_{nv} + \mathrm{Deg}(v_m)/\max_{v\in V}(\mathrm{Deg}(v)) + \mathrm{Deg}(v_n)/\max_{v\in V}(\mathrm{Deg}(v)) + |V_m \cap V_n|/|V_m \cup V_n| + \mathrm{Deg}(v_m)/\sum_{v\in V_m}\mathrm{Deg}(v) + \mathrm{Deg}(v_n)/\sum_{v\in V_n}\mathrm{Deg}(v)$.

Meanwhile, we consider the distance of both edges as follows.

Definition 7 (edge distance). Suppose $e_{ij} = (v_i, v_j)$ and $e_{mn} = (v_m, v_n)$ are two edges in graph G, $d(v_1, v_2)$ denotes the length of the shortest path between nodes v_1 and v_2, and I is the index of the smallest value in vector $[d(v_i, v_m), d(v_i, v_n), d(v_j, v_m), d(v_j, v_n)]$; then the distance of e_{ij} and e_{mn} is defined as follows:

$$\mathrm{dis}\left(e_{ij}, e_{mn}\right) = \begin{cases} d\left(v_i, v_m\right) + d\left(v_j, v_n\right), & I = 1 \\ d\left(v_i, v_n\right) + d\left(v_j, v_m\right), & I = 2 \\ d\left(v_j, v_m\right) + d\left(v_i, v_n\right), & I = 3 \\ d\left(v_j, v_n\right) + d\left(v_i, v_m\right), & I = 4. \end{cases} \tag{12}$$

Specifically, we first calculate the distances including $d(v_i, v_m)$, $d(v_i, v_n)$, $d(v_j, v_m)$, and $d(v_j, v_n)$ and then determine the smallest one among these distances and its index I, complete the calculation of the distance of another pair of nodes, and finally obtain the distance of two edges e_{ij} and e_{mn}.

Based on Definitions 6 and 7, we define the CORrelation (COR) of two probability distributions via JS-Divergence as follows.

Definition 8 (CORrelation). Suppose $P = \{p_1, p_2, \ldots, p_n\}$ and $Q = \{q_1, q_2, \ldots, q_n\}$ are the probability distributions of random variables $X = \{x_1, x_2, \ldots, x_n\}$ and $Y = \{y_1, y_2, \ldots, y_n\}$; then the CORrelation of P and Q is defined as follows:

$$\mathrm{COR}\left(P, Q\right) = \frac{1 - \mathrm{JSD}\left(P \parallel Q\right)}{1 + \mathrm{dis}\left(e_{ij}, e_{mn}\right)}. \tag{13}$$

According to (10)–(13) and Definition 6, we adopt the normalized PN vectors of edges e_{ij}, e_{mn} to measure their correlation as

$$\begin{aligned} \mathrm{COR}\left(e_{ij}, e_{mn}\right) = \frac{1}{1 + \mathrm{dis}\left(e_{ij}, e_{mn}\right)} \Bigg(& 1 \\ & - \frac{1}{2}\sum_{k=1}^{8} \mathrm{PN}\left(e_{ij}\right)_k \ln \frac{\mathrm{PN}\left(e_{ij}\right)_k}{\mathrm{PN}\left(M\right)_k} \\ & - \frac{1}{2}\sum_{k=1}^{8} \mathrm{PN}\left(e_{mn}\right)_k \ln \frac{\mathrm{PN}\left(e_{mn}\right)_k}{\mathrm{PN}\left(\mathrm{M}\right)_k} \Bigg), \end{aligned} \tag{14}$$

where $\mathrm{PN}(\cdot)_k$ denotes the k element of vector $\mathrm{PN}(\cdot)$ and

$$\mathrm{PN}\left(M\right)_k = \frac{1}{2}\left(\mathrm{PN}\left(e_{ij}\right)_k + \mathrm{PN}\left(e_{mn}\right)_k\right). \tag{15}$$

Substituting (15) into (14), we obtain

$$\begin{aligned} \mathrm{COR}\left(e_{ij}, e_{mn}\right) = \frac{1}{1 + \mathrm{dis}\left(e_{ij}, e_{mn}\right)} \Bigg(& 1 \\ & - \frac{1}{2}\sum_{k=1}^{8} \mathrm{PN}\left(e_{ij}\right)_k \ln \frac{2\mathrm{PN}\left(e_{ij}\right)_k}{\mathrm{PN}\left(e_{ij}\right)_k + \mathrm{PN}\left(e_{mn}\right)_k} \\ & - \frac{1}{2}\sum_{k=1}^{8} \mathrm{PN}\left(e_{mn}\right)_k \ln \frac{2\mathrm{PN}\left(e_{mn}\right)_k}{\mathrm{PN}\left(e_{ij}\right)_k + \mathrm{PN}\left(e_{mn}\right)_k} \Bigg). \end{aligned} \tag{16}$$

In our opinion, the proposed correlation definition, extracted from two aspects of relational strength and network structure, is more reasonable. The rationale is (i) graph models, commonly abstracted from networks, reflect inherent dependent relations of individuals, which naturally form edge correlations and (ii) the weights of edges in weighted graphs describe the affinity degree of individuals' relations, which also influence the variances of edge correlations.

Example 9. Take e_{24} and e_{25} in D_W as an example to demonstrate the calculation of $\mathrm{COR}(e_{24}, e_{25})$. According to (8) and (9), we have

$$\begin{aligned} \mathrm{PF}\left(e_{24}\right) &= \left(\frac{8}{8}, \frac{8}{15}, \frac{8}{16}, \frac{4}{4}, \frac{3}{4}, \frac{1}{6}, \frac{4}{7}, \frac{3}{7}\right), \\ \mathrm{PF}\left(e_{25}\right) &= \left(\frac{1}{8}, \frac{1}{15}, \frac{1}{6}, \frac{4}{4}, \frac{2}{4}, \frac{1}{5}, \frac{4}{7}, \frac{2}{7}\right). \end{aligned} \tag{17}$$

Furthermore, according to (10) and (11), we get

$$\mathrm{PN}\left(e_{24}\right) = (0.2020, 0.1077, 0.1010, 0.2020, 0.1515, 0.0337, 0.1154, 0.0866), \tag{18}$$

$$\mathrm{PN}\left(e_{25}\right) = (0.0429, 0.0229, 0.0572, 0.3430, 0.1715, 0.0686, 0.1960, 0.0980). \tag{19}$$

Input: Weighted dataset D_W, weight threshold T.
Output: Indicated dataset D_I.
(1) **for** (Each tuple $(i, j) \in D_W$) **do**
(2) **if** $(w_{ij} > T)$ **then**
(3) $w_{ij} = 1$;
(4) **else**
(5) $w_{ij} = 0$;
(6) **end if**
(7) **end for**
(8) **return** D_I with indicated values.

Algorithm 1: TBT algorithm.

Finally, according to (16), we obtain

$$\mathrm{COR}\left(e_{24}, e_{25}\right) = 0.4679. \tag{20}$$

4.2. Dataset Transformation. We consider some real world situations that do not need exact query answers. For example, people sometimes only want to learn about whether two individuals have an intimate relationship or not, rather than the specific number of communication or cooperation. So the privacy concern at this time is to avoid the leakage of close relationship, that is, yes or no. Therefore, the first thing we focus on is to transform a weighted dataset D_W to an indicated dataset D_I, based on a given weight threshold T. In other words, we consider replacing query "Select SUM(weight) from D_W where $w_{ij} > T$" with query "Select COUNT(∗) from D_W where $w_{ij} > T$", which satisfies some specific situations and decreases the query sensitivity simultaneously. Note that this method aims to avoid the leakage of whether an edge satisfying the given threshold condition exists and not to avoid the weights of edges satisfying the one exposed. In our opinion, this solution is reasonable and suitable for achieving privacy protection via DP in spite of some utility loss.

To this end, we propose the TBT algorithm to modify raw weight values w_{ij} in D_W as an indicated value; that is, $w_{ij} = 1$ if $w_{ij} > T$; otherwise, $w_{ij} = 0$, thus transforming D_W to D_I. The TBT algorithm is presented in Algorithm 1.

4.3. Correlated Edge Differential Privacy. As discussed above, we only consider the situations: the query answers responding to a correlated weighted data are yes or no, which indicates whether two individuals have close relationship. That is, the privacy concern herein is to avoid the leakage of whether there is a close relationship between two individuals, in a weighted dataset whose at most z tuples are correlated, where z is the correlated parameter. To this end, we first define correlated neighboring databases as follows.

Definition 10 (correlated neighboring databases). Any pair of databases $D_{\mathrm{COR},z}$ and $D'_{\mathrm{COR},z}$ are correlated neighboring databases, if the weight change of a tuple in $D_{\mathrm{COR},z}$ results in the weight changes of at most $z - 1$ other correlated tuples in D' based on the correlation $\mathrm{COR}(\cdot,\cdot)$ of both tuples.

Note that the neighboring databases in Definition 10 are described by two parameters: the correlation $\mathrm{COR}(\cdot,\cdot)$ aforementioned and the correlated parameter z. Specifically, we have the following.

(i) Based on JS-Divergence, we have the following conclusion about the correlation $\mathrm{COR}(\cdot,\cdot)$.

Theorem 11. *For any two edges e_{ij} and e_{mn} in the weighted graph G corresponding to a network dataset D_W, $0 \le COR(e_{ij}, e_{mn}) \le 1$ holds.*

Proof. For the ease of exposition, we denote the last two items in the numerator of (16) as follows.

$$\begin{aligned}\mathrm{JSD}\left(e_{ij} \parallel e_{mn}\right) &= \frac{1}{2} \\ &\cdot \sum_{k=1}^{8} \left(\mathrm{PN}\left(e_{ij}\right)_k \ln \frac{2\mathrm{PN}\left(e_{ij}\right)_k}{\mathrm{PN}\left(e_{ij}\right)_k + \mathrm{PN}\left(e_{mn}\right)_k} \right. \\ &\left. + \mathrm{PN}\left(e_{mn}\right)_k \ln \frac{2\mathrm{PN}\left(e_{mn}\right)_k}{\mathrm{PN}\left(e_{ij}\right)_k + \mathrm{PN}\left(e_{mn}\right)_k} \right).\end{aligned} \tag{21}$$

Since $\mathrm{PN}(\cdot)$ denotes a probability distribution, we have

$$\begin{aligned}\sum_{k=1}^{8} \mathrm{PN}\left(e_{ij}\right)_k &= 1, \\ \sum_{k=1}^{8} \mathrm{PN}\left(e_{\mathrm{mn}}\right)_k &= 1;\end{aligned} \tag{22}$$

we consider two cases separately.

Case 1 ($\mathrm{COR}(e_{ij}, e_{mn}) \le 1$). Since $\ln(x) \le x - 1$ when $x > 0$, we have

$$\begin{aligned}&\sum_{k=1}^{8} \mathrm{PN}\left(e_{ij}\right)_k \ln \frac{2\mathrm{PN}\left(e_{ij}\right)_k}{\mathrm{PN}\left(e_{ij}\right)_k + \mathrm{PN}\left(e_{mn}\right)_k} \\ &\le \sum_{k=1}^{8} \mathrm{PN}\left(e_{ij}\right)_k \left(\frac{2\mathrm{PN}\left(e_{ij}\right)_k}{\mathrm{PN}\left(e_{ij}\right)_k + \mathrm{PN}\left(e_{mn}\right)_k} - 1 \right) \\ &= \sum_{k=1}^{8} \mathrm{PN}\left(e_{ij}\right)_k \frac{\mathrm{PN}\left(e_{ij}\right)_k - \mathrm{PN}\left(e_{mn}\right)_k}{\mathrm{PN}\left(e_{ij}\right)_k + \mathrm{PN}\left(e_{mn}\right)_k} \\ &\le \sum_{k=1}^{8} \mathrm{PN}\left(e_{ij}\right)_k .\end{aligned} \tag{23}$$

Substituting (22) into (23), we have

$$\sum_{k=1}^{8} \mathrm{PN}\left(e_{ij}\right)_k \ln \frac{2\mathrm{PN}\left(e_{ij}\right)_k}{\mathrm{PN}\left(e_{ij}\right)_k + \mathrm{PN}\left(e_{mn}\right)_k} \le 1. \tag{24}$$

Similarly, we obtain

$$\sum_{k=1}^{8} \mathrm{PN}\left(e_{mn}\right)_k \ln \frac{2\mathrm{PN}\left(e_{mn}\right)_k}{\mathrm{PN}\left(e_{ij}\right)_k + \mathrm{PN}\left(e_{mn}\right)_k} \le 1. \tag{25}$$

Combining (21), (24), and (25), we have

$$\mathrm{JSD}\left(e_{ij} \parallel e_{mn}\right) \leq 1. \tag{26}$$

Case 2 ($\mathrm{COR}(e_{ij}, e_{mn}) \geq 0$). Since $\ln(x) \geq 1 - 1/x$ when $x > 0$, we have

$$\begin{aligned}
&\sum_{k=1}^{8} \mathrm{PN}\left(e_{ij}\right)_k \ln \frac{2\mathrm{PN}\left(e_{ij}\right)_k}{\mathrm{PN}\left(e_{ij}\right)_k + \mathrm{PN}\left(e_{mn}\right)_k} \\
&\geq \sum_{k=1}^{8} \mathrm{PN}\left(e_{ij}\right)_k \left(1 - \frac{\mathrm{PN}\left(e_{ij}\right)_k + \mathrm{PN}\left(e_{mn}\right)_k}{2\mathrm{PN}\left(e_{ij}\right)_k}\right) \\
&= \sum_{k=1}^{8} \frac{\mathrm{PN}\left(e_{ij}\right)_k - \mathrm{PN}\left(e_{mn}\right)_k}{2}.
\end{aligned} \tag{27}$$

Similarly, we obtain

$$\begin{aligned}
&\sum_{k=1}^{8} \mathrm{PN}\left(e_{mn}\right)_k \ln \frac{2\mathrm{PN}\left(e_{mn}\right)_k}{\mathrm{PN}\left(e_{ij}\right)_k + \mathrm{PN}\left(e_{mn}\right)_k} \\
&\geq \sum_{k=1}^{8} \frac{\mathrm{PN}\left(e_{mn}\right)_k - \mathrm{PN}\left(e_{ij}\right)_k}{2}.
\end{aligned} \tag{28}$$

Combining (21), (27), and (28), we have

$$\mathrm{JSD}\left(e_{ij} \parallel e_{mn}\right) \geq 0. \tag{29}$$

Combining (26) with (29), then we have

$$0 \leq \mathrm{JSD}\left(e_{ij} \parallel e_{mn}\right) \leq 1. \tag{30}$$

Note that $\mathrm{dis}(e_{ij}, e_{mn}) \geq 0$ due to (12). Finally, according to (16) and (30), we have

$$0 \leq \mathrm{COR}\left(e_{ij}, e_{mn}\right) \leq 1. \tag{31}$$

□

Clearly, $\mathrm{COR}(e_{ij}, e_{mn}) = 0$ denotes e_{ij} is independent of e_{mn}; that is, the corresponding tuples in a dataset are independent. $\mathrm{COR}(e_{ij}, e_{mn}) = 1$ denotes e_{ij} is fully dependent to e_{mn}; that is, the corresponding tuples in a dataset are fully correlated. $0 < \mathrm{COR}(e_{ij}, e_{mn}) < 1$ denotes e_{ij} is partially dependent on e_{mn}; that is, the corresponding tuples in a dataset are partially correlated.

(ii) Similar to [23–25], we introduce the correlated parameter z representing that there are at most z correlated tuples in a dataset. In other words, a tuple is correlated with at most $z - 1$ other tuples; that is, an edge in a graph is correlated with at most $z - 1$ other edges. Obviously, $z = 1$ represents the independent case of tuples in a dataset, $z = n$ represents the fully correlated case of tuples in a dataset, and $1 < z < n$ represents the partially correlated case of tuples in a dataset. Therefore, the variance of z increases the flexibility of Definition 10.

Furthermore, we define the ϵ-CEDP model as follows.

Definition 12 (ϵ-correlated edge differential privacy). A randomized mechanism $\mathcal{M}$ satisfies ϵ-differential privacy if, for any neighboring databases $D_{\mathrm{COR},z}$ and $D'_{\mathrm{COR},z}$ and for any output $S \in O(\mathcal{M})$ representing the possible output set of $\mathcal{M}$,

$$\begin{aligned}
\Pr\left(\mathcal{M}\left(D_{\mathrm{COR},z}\right) = S\right) &\leq \exp(\epsilon) \\
&\cdot \Pr\left(\mathcal{M}\left(D'_{\mathrm{COR},z}\right) = S\right),
\end{aligned} \tag{32}$$

where ϵ is the privacy budget depicting the probabilistic difference between the same outputs of $\mathcal{M}$ over $D_{\mathrm{COR},z}$ and $D'_{\mathrm{COR},z}$ and $\mathrm{COR}(\cdot,\cdot)$ and z are the correlation of two tuples and the correlated parameter representing the maximal number of correlated tuples, respectively.

4.4. Sensitivity Calculation. After transforming weighted dataset D_W to indicated dataset D_I, we add Laplace noise to query answers based on the ϵ-CEDP model. Laplace noise is determined by two factors: privacy budget ϵ and the global sensitivity of a query, and the latter refers to the maximal change of query result due to the modification of only one tuple. Here, for a query f, assume the global sensitivity of f, resulting from the change of tuple t_j, in independent settings is Δf_j. Clearly, $\Delta f_j = 1$. However, for dataset D_I with n tuples where at most z tuples are correlated, the query sensitivity resulted from modifying tuple t_i, called Edge Sensitivity denoted by ES_i, is more complex. Specifically, (i) if $\mathrm{COR}(t_i, t_j) = 0$, that is, $z = 1$, denoting the independent case, $\mathrm{ES}_i = 1$, (ii) if $\mathrm{COR}(t_i, t_j) = 1$ and $2 \leq z \leq n$, denoting the fully correlated case, $\mathrm{ES}_i = z$, and (iii) if $0 < \mathrm{COR}(e_{ij}, e_{mn}) < 1$ and $2 \leq z \leq n$, denoting the partially correlated case, ES_i is defined as follows.

$$\mathrm{ES}_i = \sum_{j=1}^{n} \left|\mathrm{COR}\left(t_i, t_j\right)\right| \Delta f_j. \tag{33}$$

Since the change of a tuple only affects at most other $z - 1$ correlated tuples, ES_i can be rewritten as

$$\mathrm{ES}_i = \sum_{j=1}^{z} \left|\mathrm{COR}\left(t_i, t_j\right)\right| \Delta f_j. \tag{34}$$

Finally, we have the correlated sensitivity denoted by CS, that is, the maximal ES_i in dataset D_I, as follows:

$$\mathrm{CS} = \max_{i \in D_I} \mathrm{ES}_i. \tag{35}$$

Note that the CS is also suitable for the independent and fully correlated cases. Based on the CS, we can achieve ϵ-CEDP, which is shown as follows.

Theorem 13. *Given any query function $f : D_{\mathrm{COR},z} \to \mathbb{R}^d$, where $D_{\mathrm{COR},z}$ is a correlated dataset with the correlation definition $\mathrm{COR}(\cdot,\cdot)$ and the correlated parameter z and $\mathbb{R}^d$ is a d-dimension real-valued vector, the correlated sensitivity CS of f, and privacy budget ϵ, a randomized mechanism $\mathcal{M}$,*

$$\mathcal{M}\left(D_{\mathrm{COR},z}\right) = f\left(D_{\mathrm{COR},z}\right) + \mathit{Laplace}\left(\frac{\mathrm{CS}}{\epsilon}\right), \tag{36}$$

provides ϵ-CEDP, where $\mathit{Laplace}(\cdot)$ denotes Laplace noise.

Input: Original dataset D_W, privacy budget ϵ, correlated parameter z, threshold T and query set $\mathcal{Q}$.
Output: Noisy query result $\mathcal{M}(D_W)$.
(1) Calculate the correlation $\mathrm{COR}(\cdot,\cdot)$ of any two edges in D_W according to Eq. (16);
(2) Call Algorithm $\mathrm{TBT}(D_W, T)$, return D_I;
(3) Calculate the correlated sensitivity CS according to Eq. (35);
(4) **for** (Each $f \in \mathcal{Q}$) **do**
(5) $\quad \mathcal{M}(D_I) = f(D_I) + \mathrm{Laplace}\left(\frac{\mathrm{CS}}{\epsilon}\right)$;
(6) **end for**
(7) **return** Noisy query result $\mathcal{M}(D_I)$ as $\mathcal{M}(D_W)$.

Algorithm 2: NDR algorithm.

Proof.

$$\begin{aligned}
&\frac{\Pr\left(\mathcal{M}\left(D_{\mathrm{COR},z}\right)=S\right)}{\Pr\left(\mathcal{M}\left(D'_{\mathrm{COR},z}\right)=S\right)} \\
&= \frac{\exp\left(-\epsilon\left|\mathcal{M}\left(D_{\mathrm{COR},z}\right)-f\left(D_{\mathrm{COR},z}\right)\right|/\mathrm{CS}\right)}{\exp\left(-\epsilon\left|\mathcal{M}\left(D'_{\mathrm{COR},z}\right)-f\left(D'_{\mathrm{COR},z}\right)\right|/CS\right)} \\
&\le \exp\left(\frac{\epsilon\left|f\left(D_{\mathrm{COR},z}\right)-f\left(D'_{\mathrm{COR},z}\right)\right|}{\mathrm{CS}}\right).
\end{aligned} \tag{37}$$

According to (35), the following holds:

$$\frac{\left|f\left(D_{\mathrm{COR},z}\right)-f\left(D'_{\mathrm{COR},z}\right)\right|}{\mathrm{CS}} \le 1. \tag{38}$$

Finally, combining (37) with (38), we have

$$\frac{\Pr\left(\mathcal{M}\left(D_{\mathrm{COR},z}\right)=S\right)}{\Pr\left(\mathcal{M}\left(D'_{\mathrm{COR},z}\right)=S\right)} \le \exp(\epsilon). \tag{39}$$

□

For indicated dataset D_I with weight $w_{ij} \in \{0,1\}$ and the correlated parameter z, we can easily infer the global sensitivity is equal to z. Due to $0 \le \mathrm{COR}(\cdot,\cdot) \le 1$, we have $\mathrm{CS} < z$. Therefore, CS is less than the global sensitivity. In other words, added noise via CS is less than that via the global sensitivity; hence the utility of the mechanism $\mathcal{M}$ based on CS is better.

5. Network Data Release Method

Based on indicated dataset D_I and the CS discussed in Section 4, we proposed a network data release method in special cases, which achieves the ϵ-CEDP model. Furthermore, the theoretical analysis of privacy and utility is elaborated.

5.1. NDR Algorithm. The goal of NDR algorithm is to achieve the tradeoff between privacy and utility under correlated settings. To this end, three phases are taken into account: (i) for achieving the correlation of two tuples in D_W, we transform dataset D_W into the corresponding graph and calculate the correlation $\mathrm{COR}(\cdot,\cdot)$ of both edges via the JS-Divergence, (ii) based on a given weight threshold T, we convert D_W into D_I via TBT algorithm. In other words, the sensitivity in independent settings is 1, irrelevant to the weights. Furthermore, we implement the calculation of CS, and (iii) combining the affordable privacy budget ϵ with CS, we calculate the added Laplace noise and finally obtain the noisy query result $\mathcal{M}(D_I)$ for query f in query set $\mathcal{Q}$. The NDR algorithm is presented in Algorithm 2.

5.2. Utility Analysis. Clearly, NDR algorithm satisfies the ϵ-CEDP model. To conduct utility analysis, we adopt the (α,δ)-useful definition in [40] to depict the utility of NDR as follows.

Definition 14. A mechanism NDR is (α,δ)-useful for a query f in all queries $\mathcal{Q}$, if, with probability at least $1-\delta$, for any query f and dataset D, NDR satisfies $\max_{f\in\mathcal{Q}}|\mathrm{NDR}(D) - f(D)| \le \alpha$.

Based on Definition 14, we obtain the following utility analysis.

Theorem 15. *For any query $f \in \mathcal{Q}$ and dataset D, a mechanism NDR satisfies (α,δ)-useful if NDR can obtain $\max_{f\in\mathcal{Q}}|NDR(D) - f(D)| \le \alpha$ with at least probability $1-\delta$ when $\delta \ge \exp(-\epsilon\alpha/CS)$.*

Proof. By Definition 14, we have

$$\begin{aligned}
&\Pr\left(\max_{f\in\mathcal{Q}}\left|\mathrm{NDR}(D) - f(D)\right| \le \alpha\right) \\
&= \Pr\left(\max_{f\in\mathcal{Q}}\left|\mathrm{Laplace}\left(\frac{\mathrm{CS}}{\epsilon}\right)\right| \le \alpha\right) \\
&= \int_{-\alpha}^{\alpha}\frac{\epsilon}{2\mathrm{CS}}\exp\left(-\frac{\epsilon x}{\mathrm{CS}}\right)dx \\
&= \int_{0}^{\alpha}\frac{\epsilon}{\mathrm{CS}}\exp\left(\frac{-\epsilon x}{\mathrm{CS}}\right)dx = 1-\exp\left(-\frac{\epsilon\alpha}{\mathrm{CS}}\right).
\end{aligned} \tag{40}$$

If $\delta \ge \exp(-\epsilon\alpha/\mathrm{CS})$, then the following holds:

$$1-\exp\left(-\frac{\epsilon\alpha}{\mathrm{CS}}\right) \le 1-\delta. \tag{41}$$

According to (40) and (41), we obtain

$$\Pr\left(\max_{f\in\mathcal{Q}}\left|\mathrm{NDR}\left(D\right)-f\left(D\right)\right|\leq\alpha\right)\leq 1-\delta. \quad (42)$$

Therefore, mechanism NDR satisfies (α,δ)-useful. □

6. Experiment

Generally, the goal of privacy preserving is to achieve maximal utilities while maintaining required privacy guarantees; that is, the tradeoff between privacy and utility is desired. In this section, we first present the better privacy guarantees and utilities of Algorithm NDR based on the definition of (α,δ)-useful and then further demonstrate its better utilities in terms of mean absolute error (MAE). Here the Baseline algorithm adopts the multiplication in [23] to handle with the correlated tuples in a network dataset. Considering the constraint of applying Pearson correlation coefficient, we do not adopt the method using Pearson correlation coefficient as comparison reference in the following experiments. To verify the advantages of Algorithm NDR concerning privacy and utility, we conduct NDR and Baseline algorithms on three datasets: geom, out explained in Section 4, and randomly generated dataset (rgd), which is a randomly generated weighted network containing 100 vertices and 1645 edges. The weight of each edge is uniformly distributed in interval $[1, 50]$. Such doing can also show the better adaption of the proposed correlation metric and algorithm over real world and synthetic datasets. Without loss of generality, threshold T here is set as 0, and the selection of its value is to be investigated in future work.

6.1. Privacy and Utility. We analyzed privacy and utility of NDR and Baseline algorithms in terms of (α,δ)-useful when the correlated parameter is set to the size of the whole dataset. In terms of privacy, we evaluate the consumption of privacy budget ϵ under the same accuracy α and the same possibility $1-\delta$. Clearly, the smaller the consumed privacy budget, the better the performance of algorithm.

Figures 3(a), 3(c), and 3(e) present the variation of privacy budget, consumed by algorithms NDR and Baseline based on datasets geom, out, and rgd, with the increase of α from 1 to 40 when δ equals 0.1 and 0.5, respectively. From Figures 3(a), 3(c), and 3(e), we can see that privacy budgets decrease in all cases with the increase of α. The reason is that, with the relaxation of α, larger noise can be allowed when δ stays fixed; therefore algorithm can consume smaller privacy budget. Meanwhile, we also see that privacy budgets decrease with the increase of δ from 0.1 to 0.5 when α stays fixed. Because the possibility of satisfying accuracy requirement decreases with the increase of δ, which means that algorithms can have more chances to add larger noise; that is, algorithms can use smaller privacy budget. Such advantage of both algorithms especially in the case of $\alpha = 1$ is more obvious than that of other ones. In fact, when δ stays constant, the higher the accuracy presented by α, the more the privacy budget needed by algorithms.

On the other hand, Figures 3(b), 3(d), and 3(f) demonstrate the variation of δ of algorithms NDR and Baseline based on datasets geom, out, and rgd, with the increase of α from 0 to 10000 when ϵ equals 0.1 and 1.0, respectively. We can see that δ decreases in all cases with the increase of α; that is, the possibility increases with the increase of α. Note that this trend varies from dataset to dataset; for example, the possibility and accuracy of algorithm NDR over datasets geom and out are evidently different when $\epsilon = 1$. Clearly, when ϵ is determined, the possibility increases with the relaxation of α. In addition, we find that algorithm NDR can have larger possibility, that is, smaller δ, than the Baseline algorithm to achieve the same accuracy α under the same level of privacy budget ϵ. Also, when ϵ increases from 0.1 to 1.0, algorithms also have possibility to achieve the same accuracy α, which is easily understood from (40).

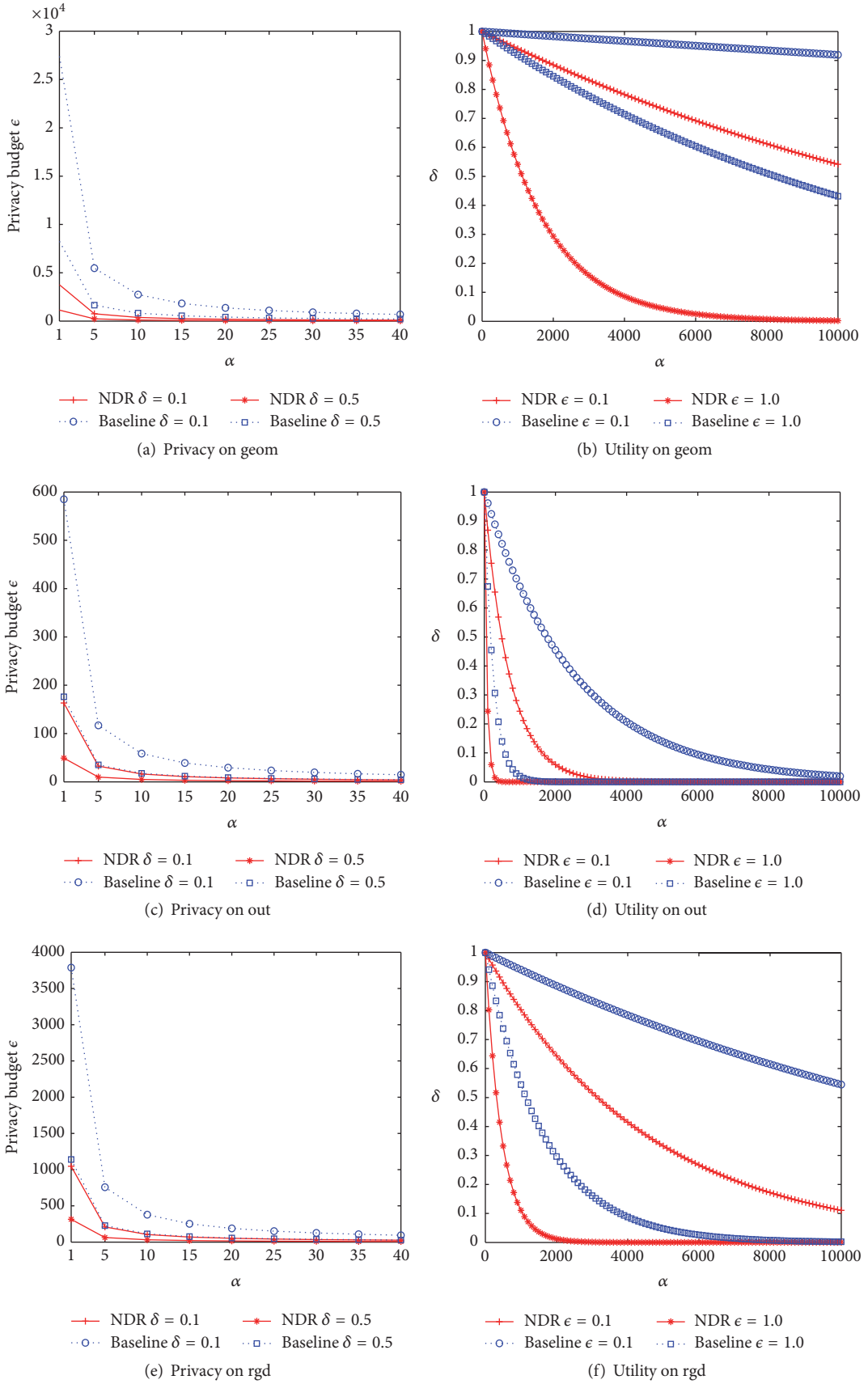
6.2. Utility. We adopt MAE, that is, $(1/|\mathcal{Q}|)\sum_{f\in\mathcal{Q}}|\widehat{f}(t)-f(t)|$, to depict the performance of algorithms NDR and Baseline. Obviously, the smaller the MAE value, the better the utility. For each dataset, 10000 queries are randomly generated, and each query result ranges from 0 to the maximal number of tuples.

Figures 4(a), 4(c), and 4(e) show the variances of MAEs of NDR and Baseline algorithms, over datasets geom, out, and rgd, under various privacy budgets when the correlated parameter z is 10. From Figures 4(a), 4(c), and 4(e), we can see that the MAEs of both algorithms decrease with the increase of privacy budget ϵ from 0.1 to 1. Because larger privacy budget leads to smaller noise added to raw data, the downtrends always hold. More importantly, algorithm NDR can obtain better accuracy, that is, smaller MAE, under various ϵ. Furthermore, the smaller the privacy budget ϵ, the more obvious such advantage. The reason is that algorithm NDR adopts the more reasonable correlation metric compared with the Baseline algorithm.

Figures 4(b), 4(d), and 4(f) show the variances of MAEs of NDR and Baseline algorithms, over datasets geom, out, and rgd, under various correlated parameters when privacy budget ϵ is 0.5. In Figures 4(b), 4(d), and 4(f), we find that the MAEs of both algorithms increase with the increase of correlated parameter z from 1 to 40. Undoubtedly, with the increase of the number of correlated tuples in a dataset, larger noise needs to be injected to eliminate the effect of tuple correlation, which necessarily results in the increase of MAE. In addition, we also note that algorithm NDR can obtain better accuracy, that is, smaller MAE, under various correlated parameters compared with the Baseline algorithm. Also, the larger the correlated parameter, the larger such advantage. All these advantages are due to the more reasonable correlation metric, which is proposed in Section 4 and adopted by algorithm NDR.

7. Conclusion

In this paper, we focus on adopting differential privacy model to avoid the leakage of close relationship between two individuals in a network. To this end, we first extract the PF vector from both aspects of node degree and edge weight to depict an edge in a network dataset and then design the correlation metric of two edges via JS-Divergence to avoid the



(a) Privacy on geom

(b) Utility on geom

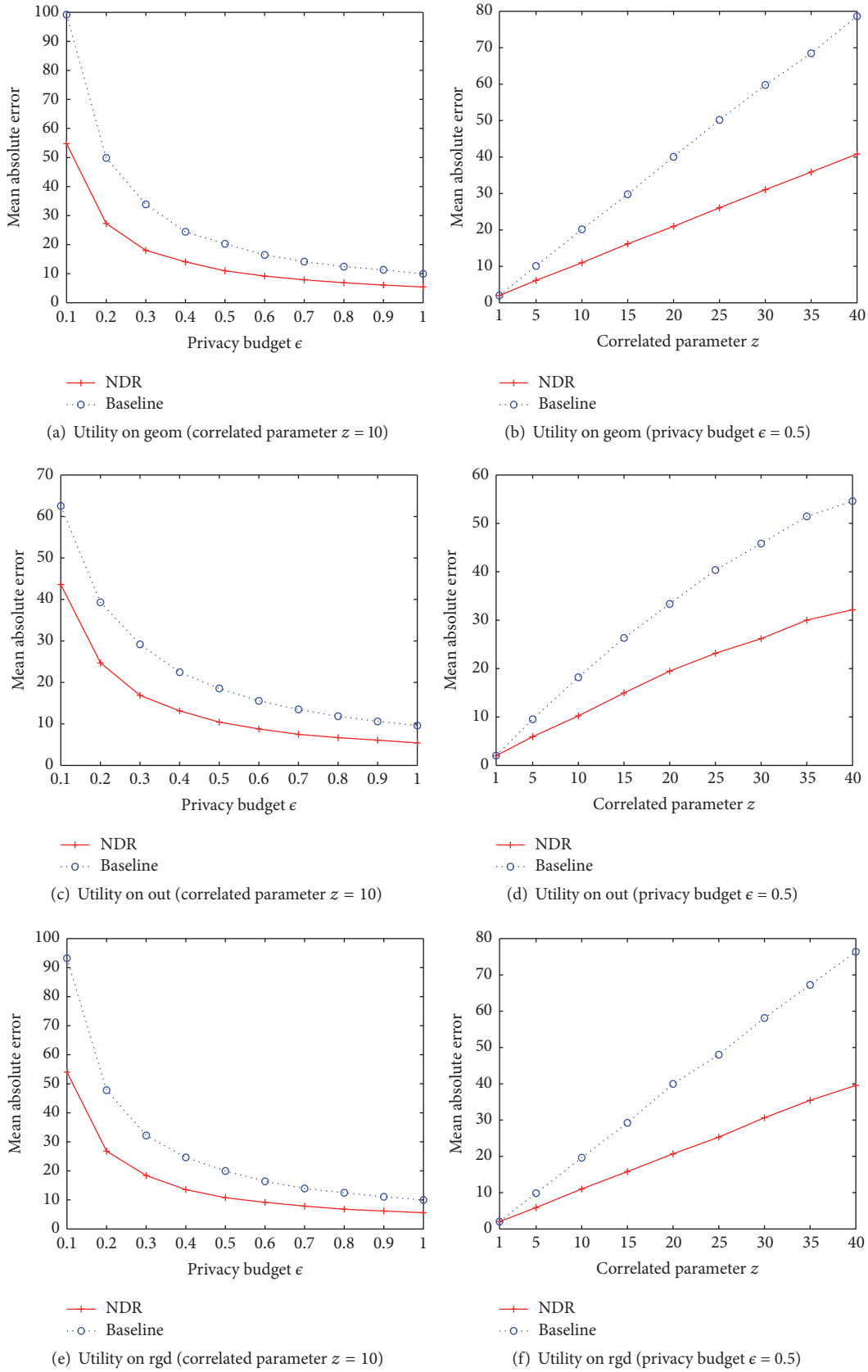
(c) Privacy on out

(d) Utility on out

(e) Privacy on rgd

(f) Utility on rgd

Figure 3: Comparison of privacy and utility on network data in terms of (α, δ)-useful.



(a) Utility on geom (correlated parameter $z = 10$)

(b) Utility on geom (privacy budget $\epsilon = 0.5$)

(c) Utility on out (correlated parameter $z = 10$)

(d) Utility on out (privacy budget $\epsilon = 0.5$)

(e) Utility on rgd (correlated parameter $z = 10$)

(f) Utility on rgd (privacy budget $\epsilon = 0.5$)

Figure 4: Comparison of utility on network data in terms of mean absolute error.

constraint of adopting Pearson correlation coefficient. Next, we proposed the ϵ-CEDP model to deal with the correlated dataset by introducing two parameters including our correlation metric and the correlated parameter. Furthermore, we present the NDR algorithm based on the ϵ-CEDP and discuss its privacy and utility in terms of the definition of (α, δ)-useful. Extensive experiments on real and synthetic network datasets verify the advantages of our proposed privacy preserving model and algorithm concerning privacy and utility. Admittedly, the proposed solution is currently appropriate for weighted network datasets, and other datasets are out of the scope of this paper. In future work, we will discuss the impacts of choosing weight threshold on algorithm performances, explore more appropriate correlation metrics, and investigate privacy preserving algorithms in different applications.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work is partly supported by the Fundamental Research Funds for the Central Universities of China under Grant no. GK201703061, the National Natural Science Foundation of China under Grants nos. 61402273 and 61373083, the National Science Foundation (NSF) under Grants nos. CNS-1252292, 1741277, and 1704287, and the Natural Science Basic Research Plan in Shaanxi Province of China under Grants nos. 2017JM6060 and 2017JM6103.

References

[1] J. Liu, J. Wan, B. Zeng, Q. Wang, H. Song, and M. Qiu, "A scalable and quick-response software defined vehicular network assisted by mobile edge computing," *IEEE Communications Magazine*, vol. 55, no. 7, pp. 94–100, 2017.

[2] P. Hu, H. Ning, T. Qiu, H. Song, Y. Wang, and X. Yao, "Security and privacy preservation scheme of face identification and resolution framework using fog computing in internet of things," *IEEE Internet of Things Journal*, vol. 4, no. 5, pp. 1143–1155, 2017.

[3] Z. He, Z. Cai, and J. Yu, "Latent-data privacy preserving with customized data utility for social network data," *IEEE Transactions on Vehicular Technology*, vol. PP, no. 99, pp. 1-1, 2017.

[4] X. Zheng, Z. Cai, J. Yu, C. Wang, and Y. Li, "Follow but no track: privacy preserved profile publishing in cyber-physical social systems," *IEEE Internet of Things Journal*, vol. PP, no. 99, pp. 1-1, 2017.

[5] Z. Cai, Z. He, X. Guan, and Y. Li, "Collective data-sanitization for preventing sensitive information inference attacks in social networks," *IEEE Transactions on Dependable and Secure Computing*, vol. 99, pp. 1-1, 2017.

[6] Y. Liang, Z. Cai, Q. Han, and Y. Li, "Location privacy leakage through sensory data," *Security and Communication Networks*, vol. 2017, Article ID 7576307, 12 pages, 2017.

[7] X. Zheng, Z. Cai, J. Li, and H. Gao, "Location-privacy-aware review publication mechanism for local business service systems," in *Proceedings of the IEEE INFOCOM 2017 - IEEE Conference on Computer Communications*, pp. 1–9, Atlanta, GA, USA, May 2017.

[8] L. Zhang, Z. Cai, and X. Wang, "FakeMask: a novel privacy preserving approach for smartphones," *IEEE Transactions on Network and Service Management*, vol. 13, no. 2, pp. 335–348, 2016.

[9] Y. Wang, Z. Cai, G. Yin, Y. Gao, X. Tong, and G. Wu, "An incentive mechanism with privacy protection in mobile crowdsourcing systems," *Computer Networks*, vol. 102, Supplement C, pp. 157–171, June 2016.

[10] W. Wang, L. Ying, and J. Zhang, "On the relation between identifiability, differential privacy, and mutual-information privacy," *Institute of Electrical and Electronics Engineers Transactions on Information Theory*, vol. 62, no. 9, pp. 5018–5029, 2016.

[11] P. Zhou, Y. Zhou, D. Wu, and H. Jin, "Differentially Private Online Learning for Cloud-Based Video Recommendation with Multimedia Big Data in Social Networks," *IEEE Transactions on Multimedia*, vol. 18, no. 6, pp. 1217–1229, 2016.

[12] J. Ni, K. Zhang, K. Alharbi, X. Lin, N. Zhang, and X. S. Shen, "Differentially private smart metering with fault tolerance and range-based filtering," *IEEE Transactions on Smart Grid*, vol. 8, no. 5, pp. 2483–2493, 2017.

[13] H. To, G. Ghinita, L. Fan, and C. Shahabi, "Differentially private location protection for worker datasets in spatial crowdsourcing," *IEEE Transactions on Mobile Computing*, vol. 16, no. 4, pp. 934–949, 2017.

[14] C. Lin, P. Wang, H. Song, Y. Zhou, Q. Liu, and G. Wu, "A differential privacy protection scheme for sensitive big data in body sensor networks," *Annals of Telecommunications-Annales des Télécommunications*, vol. 71, no. 9-10, pp. 465–475, 2016.

[15] C. Lin, Z. Song, H. Song, Y. Zhou, Y. Wang, and G. Wu, "Differential privacy preserving in big data analytics for connected health," *Journal of Medical Systems*, vol. 40, no. 4, article no. 97, pp. 1–9, 2016.

[16] S. Xu, S. Su, X. Cheng, K. Xiao, and L. Xiong, "Differentially private frequent sequence mining," *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 11, pp. 2910–2926, 2016.

[17] W. Tong, J. Hua, and S. Zhong, "A jointly differentially private scheduling protocol for ridesharing services," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 10, pp. 2444–2456, 2017.

[18] D. Kifer and A. Machanavajjhala, "A rigorous and customizable framework for privacy," in *Proceedings of the 31st Symposium on Principles of Database Systems (PODS '12)*, pp. 77–88, Scottsdale, Arizona, USA, May 2012.

[19] D. Kifer and A. Machanavajjhala, "No free lunch in data privacy," in *Proceedings of the 2011 ACM SIGMOD and 30th PODS 2011 Conference on Management of Data (SIGMOD)*, pp. 193–204, Athens, Greece, June 2011.

[20] C. Dwork, "Differential privacy," in *Proceedings of the 33rd International Conference on Automata, Languages and Programming - Volume Part II (ICALP '06)*, pp. 1–12, Venice, Italy, 2006.

[21] X. He, A. Machanavajjhala, and B. Ding, "Blowfish privacy: Tuning privacy-utility trade-offs using policies," in *Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data, (SIGMOD '14)*, pp. 1447–1458, Snowbird, Utah, USA, June 2014.

[22] B. Yang, I. Sato, and H. Nakagawa, "Bayesian differential privacy on correlated data," in *Proceedings of the ACM SIGMOD International Conference on Management of Data, (SIGMOD '15)*, pp. 747–762, Melbourne, Victoria, Australia, June 2015.

[23] R. Chen, B. C. M. Fung, P. S. Yu, and B. C. Desai, "Correlated network data publication via differential privacy," *The VLDB Journal*, vol. 23, no. 4, pp. 653–676, 2014.

[24] T. Zhu, P. Xiong, G. Li, and W. Zhou, "Correlated differential privacy: hiding information in Non-IID data set," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 2, article no. A2, pp. 229–242, 2015.

[25] C. Liu, S. Chakraborty, and P. Mittal, "Dependence makes you vulnerable: differential privacy under dependent tuples," in *Proceedings of the Network and Distributed System Security Symposium (NDSS '16)*, San Diego, Calif, USA.

[26] Y. Xiao and L. Xiong, "Protecting locations with differential privacy under temporal correlations," in *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security, (CCS '15)*, pp. 1298–1309, Denver, Colorado, USA, October 2015.

[27] X. Wu, T. Wu, M. Khan, Q. Ni, and W. Dou, "Game theory based correlated privacy preserving analysis in big data," *IEEE Transactions on Big Data*, vol. PP, no. 99, pp. 1-1, 2017.

[28] L. Zhang, Y. Li, L. Wang, J. Lu, P. Li, and X. Wang, "An efficient context-aware privacy preserving approach for smartphones," *Security & Communication Networks*, vol. 2017, no. 2, pp. 1–11, 2017.

[29] Y. Cao, M. Yoshikawa, Y. Xiao, and L. Xiong, "Quantifying differential privacy under temporal correlations," in *Proceeding of the IEEE 33rd International Conference on Data Engineering (ICDE '17)*, pp. 821–832, San Diego, CA, USA, 2017.

[30] L. Ou, Z. Qin, Y. Liu, H. Yin, Y. Hu, and H. Chen, "Multi-user location correlation protection with differential privacy," in *Proceedings of the 2016 IEEE 22nd International Conference on Parallel and Distributed Systems (ICPADS '16)*, pp. 422–429, Wuhan, China, December 2016.

[31] D. N. Reshef, Y. A. Reshef, H. K. Finucane et al., "Detecting novel associations in large data sets," *Science*, vol. 334, no. 6062, pp. 1518–1524, 2011.

[32] C. Dwork, F. McSherry, K. Nissim, and A. Smith, "Calibrating noise to sensitivity in private data analysis," in *Proceedings of the Third Conference on Theory of Cryptography (TCC '06)*, vol. 3876, pp. 265–284, New York, NY, USA, March 2006.

[33] F. McSherry and K. Talwar, "Mechanism design via differential privacy," in *Proceedings of the 48th Annual Symposium on Foundations of Computer Science (FOCS '07)*, pp. 94–103, Providence, RI, USA, October 2007.

[34] V. Batagelj and A. Mrvar, "Pajek datasets," http://vlado.fmf.uni-lj.si/pub/networks/data/.

[35] N. H. F. Beebe and H. F. Nelson, "Beebe's bibliographies page," http://www.math.utah.edu/~beebe/bibliographies.html.

[36] B. Jones, "Computational geometry database," http://jeffe.cs.illinois.edu/compgeom/biblios.html.

[37] *Les misérables network dataset – KONECT*, 2016, http://konect.uni-koblenz.de/networks/moreno_lesmis.

[38] D. E. Knuth, *The stanford graph base: a platform for combinatorial computing*, vol. 37, Addison-Wesley Reading, Boston, Mass, USA, 1993.

[39] J. Kunegis, "KONECT — The koblenz network collection," in *Proceeding of the 22nd International Conference on World Wide Web Companion (WWW '13)*, pp. 1343–1350, Rio de Janeiro, Brazil, May 2013.

[40] A. Blum, K. Ligett, and A. Roth, "A learning theory approach to non-interactive database privacy," in *Proceedings of the Fortieth Annual ACM Symposium on Theory of Computing (STOC '08)*, pp. 609–618, ACM, New York, NY, USA, 2008.