TSINGHUA SCIENCE AND TECHNOLOGY
ISSN 1007-0214 02/18 pp235–243
DOI: 10.26599/TST.2021.9010018
Volume 27, Number 2, April 2022

Efficient Publication of Distributed and Overlapping Graph Data Under Differential Privacy

Xu Zheng, Lizong Zhang*, Kaiyang Li, and Xi Zeng

Abstract: Graph data publication has been considered as an important step for data analysis and mining. Graph data, which provide knowledge on interactions among entities, can be locally generated and held by distributed data owners. These data are usually sensitive and private, because they may be related to owners' personal activities and can be hijacked by adversaries to conduct inference attacks. Current solutions either consider private graph data as centralized contents or disregard the overlapping of graphs in distributed manners. Therefore, this work proposes a novel framework for distributed graph publication. In this framework, differential privacy is applied to justify the safety of the published contents. It includes four phases, i.e., graph combination, plan construction sharing, data perturbation, and graph reconstruction. The published graph selection is guided by one data coordinator, and each graph is perturbed carefully with the Laplace mechanism. The problem of graph selection is formulated and proven to be NP-complete. Then, a heuristic algorithm is proposed for selection. The correctness of the combined graph and the differential privacy on all edges are analyzed. This study also discusses a scenario without a data coordinator and proposes some insights into graph publication.

Key words: graph data; distributed data publication; differential privacy

1 Introduction

Diversity and capabilities on distributed data collection have remarkably increased[1]. Advanced techniques, such as Internet of Things (IoTs) and mobile social networks, contribute to such trends. Among these contents, graph data have constituted an imperative component because of their capability to capture both semantic and structural information[2]. Typical instances of graph data include social interactions[3, 4], bioinformatic contents[5, 6], semantic webs[7], road networks, and topological structures in a physical world[8]. With the publication and sharing of these data[9], the functionality of data analysis and mining can be significantly extended and enhanced[10, 11]. For example, the awareness of an organization can be enhanced by merging the social contacts among a group of people, thereby creating services, such as social recommendation and demographic data mining. Different companies may also combine their knowledge graphs to provide more comprehensive and intelligent services for their customers.

However, in addition to dramatic benefits, sensitivity and privacy for data owners are observed in graph data[12–14]. Individuals may come in contact with persons who should not be recognized by others, or companies can resist sharing knowledge in semantic webs related to their business secret[15]. Therefore, the arbitrary publication of graph data may pose severe threats to data owners. Adversaries can use data for various inference attacks and increase losses in diverse domains[16].

• Xu Zheng, Lizong Zhang, and Kaiyang Li are with the School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China. E-mail: {xzheng, lzhang}@uestc.edu.cn; kaiyang.li@outlook.com.
• Xi Zeng is with China Electronics Technology Cyber Security Co. Ltd., Chengdu 610000, China. E-mail: zxmm2@163.com.
∗ To whom correspondence should be addressed.
Manuscript received: 2020-12-26; revised: 2021-02-19; accepted: 2021-02-26

Moreover, graph data are divergent from typical rational data[17] and may include information covering related neighbors. Therefore, contents can partially overlap, i.e., social contacts or even their close friends are recorded by both ends. Publishing graph data is considerably challenging because one private content can be kept by multiple owners. Data owners must ensure that sensitive information, such as social ties, is globally preserved during publication.

Current efforts for publishing private graph data are unsuccessful. The mainstream of existing solutions either considers graph data as centralized data, or assumes that distributed graph data are non-overlapping. For example, in graph anonymization[18, 19], the uniqueness of each vertex is concealed, such that adversaries cannot distinguish vertices according to node distribution. Differential privacy[20] is also combined with graph publication as a standard of privacy preservation. A graph is usually encoded into sequences and perturbed accordingly. However, with these strategies, data should be centrally stored, and the distributed environment is not covered. In another domain of studies, the privacy-preserving publication of locally stored graphs is considered[21]. Local differential privacy is applied, and patterns, such as frequent graph structures, are aggregated by requestors. Nevertheless, in these methods, local graphs are assumed to be disjoint, or privacy issues underlying the overlaps are disregarded. Therefore, the privacy-preserving publication of overlapped and distributed graphs remains unresolved.

In this study, a novel framework for the publication of distributed and overlapped graph data with privacy-preserving is proposed on the basis of all related factors. In system settings, data owners locally hold their graphs, and a requestor expects to derive the combination of all these graphs. The edges in the graphs are assumed to be potentially sensitive; for example, graphs can record the social contacts of a data owner, and private interactions are sensitive in the graph. Differential privacy is adopted by our framework, which does not require any limit on the capability of adversaries and guarantees that no significant knowledge is learned from outputs. Moreover, the graphs are partially overlapped, i.e., the same vertices and edges may be stored by multiple owners. The publication mechanism must carefully consider the overlapping. Therefore, the framework mainly aims to derive the combined graph statistics, while strictly preserving the privacy of each data owner.

In our work, two scenarios are investigated. In the first case, a data coordinator schedules the data publication, and data owners publish graphs. In the second case, a data coordinator is absent. As such, data owners must locally evaluate and publish their contents. As for the first case, a four-phase mechanism is proposed for graph publication: graph combination, plan construction sharing, data perturbation, and graph reconstruction. Specifically, the problem of plan construction sharing is proven to be NP-complete, and a heuristic algorithm is designed. The correctness, the property of differential privacy, and the efficiency of the proposed mechanism are all proven and analyzed. As for the second case, graph publication is thoroughly discussed. The disadvantages of baseline solutions are demonstrated. Some assumptions and insights into the desired solutions are proposed. This study is the first to focus on the publication of distributed and overlapped graph data under differential privacy. The following contributions are provided:

- A novel framework of the publication of distributed and overlapped graph under differential privacy.
- A heuristic algorithm for graph selection, which applies Minimum Membership Set Covering (MMSC).
- Theoretical analysis on the correctness, privacy preservation, and efficiency of the proposed framework.
- Thorough discussion on graph publication in a non-coalition scenario.

The remaining parts of this paper are organized as follows. Section 2 reviews the related literature. Section 3 proposes the problem formulation and some preliminaries. Sections 4 and 5 describe solutions for two different scenarios. Section 6 concludes the paper.

2 Related Work

The publication of private graph data has been considered as a fundamental problem of graph data collection and processing. Local differential privacy[22, 23], which is considered as the *de facto* standard of privacy preservation in distributed environments, has also been applied to graph data publication. Qin et al.[21] applied the idea of graph clustering to first group users into different clusters, and then derived statistics for data analysis. Graph publication follows an incremental manner through which data owners locally perturb and share their graphs. Sun et al.[24] proposed a framework of subgraph counting in a distributed manner; in this framework, noises are recursively injected, and countings are

published. Zhang et al.[25] provided a two-phase algorithm to generate synthetic graphs under local differential privacy. Gao et al.[26] proposed a novel mechanism by combining a hierarchical random graph model, and further reduced the scales of noises. Graph reconstruction is one of the major challenges of graph data management, such as direction discovery on undirected ties[27] and direction quantification on bidirectional ties[28]. However, these studies have not provided a convincing solution for general information publication on overlapped graphs. The problem of owner selection which optimizes the utility of the published graph remains unresolved.

Studies have focused on the centralized publication of private graphs[29]. Gao and Li[30] provided a novel algorithm of privacy-preserving graph sketching publication to defend seed and subgraph based de-anonymization attacks. Guo et al.[31] applied privacy-preserving graph publication to social recommendation, such that adversaries cannot learn sensitive knowledge from the recommended results. Zhang et al.[32] developed a novel algorithm of graph encryption that can be used to estimate the shortest distance. Other studies have also explored privacy-preserving graph embedding; in this process, graphs are embedded into vectors and perturbed before they are published[33, 34]. However, these solutions should be extended to a distributed environment[35], especially when graphs are overlapping and privacy issues should be globally considered.

3 Problem Formulation

First, the considered network model is described. Then, the adversarial model and some preliminaries are introduced. Lastly, the problem formulation for the distributed graph publication is provided.

3.1 Network model

Three types of roles are involved in data publication. M ***data owners*** $\{O_1, O_2, \ldots, O_M\}$ hold one graph each and share it with other participants. For example, data owners can be regular individuals who keep their own social connections on devices. One ***data requestor*** arrives in the system and expects to gain a combination of graphs held by all participants. In our framework, a data requestor is any of the third parties, such as service providers and local governments, which make decision and provide service based on the graph data, like the social-based content recommendation and demographic data mining. A ***data coordinator*** may act as an intermediate platform that arranges the graph publication between owners and requestors. In this case, the data coordinator holds the graph data for all owners and carefully publishes them to the requestor. However, the existence of a data coordinator is not mandatory in our framework. Data owners should directly publish their contents to the requestor.

In the system setting, each data owner O_i locally holds one graph $G_i = \{V_i, E_i\}$, where $V_i = \{v_{i1}, v_{i2}, \ldots, v_{iK_i}\}$, and $E_i = \{(v_{ij}, v_{ik})\}$. This setting indicates social connections or activities. For example, vertices in V_i refer to the social neighbors of the owner, and edges refer to the recent social interactions among neighbors. All the individual graphs can be combined and constitute an integrated graph, which is denoted as $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$. Specifically, $\mathcal{V} = \bigcup_{i=1}^{M} V_i$, and $\mathcal{E} = \bigcup_{i=1}^{M} E_i$. For simplicity, we assume $\mathcal{V} = \{v_1, v_2, \ldots, v_K\}$, and $\mathcal{E} = \{e_1, e_2, \ldots, e_H\}$.

During the procedure of graph data publication, the requestor first posts a query in the system. Then, the owners share their graphs with the coordinator, who processes and generates a graph sharing plan with the requestor. The requestor fuses the received data from data owners and generates the combined graph. In this case, a reliable third party, such as a service provider, exists. In an alternative setting, i.e., the data coordinator is absent, the owners directly process and publish their data, and the requestor accordingly generates the general graph.

3.2 Adversarial model

In our system, the data requestor is assumed to be semi-honest. It follows the standard procedure to derive the combined graph, and consequently gains knowledge on the sketching of the graph. However, the data requestor also infers the social links among users. In our framework, the detailed links are assumed to be sensitive because it may refer to interactions among users. For example, social links may indicate whether target users are social neighbors or have communicated recently. Data requestors with this knowledge can conduct a series of attacks, such as the direct inference on social connections or a subsequent inference based on neighbors' profiles.

The typical differential privacy is applied to privacy preservation. Specifically, differential privacy does not request any limits on the background knowledge of adversaries. The malicious requestor can determine all the vertices and edges except one target edge.

Differential privacy guarantees that no significant knowledge on the existence of the edge is learned. The formal definition of differential privacy is shown in Definition 1.

Definition 1 (Differential privacy[20]) Assume the set of contents to be D_i, and D_i' is another content set that differs from D_i on just one transaction. Then an algorithm or a data publication mechanism A for the accumulated number of contents satisfies ϵ-Differential Privacy (ϵ-DP), where ϵ is denoted as the privacy budget for differential privacy, and $\epsilon \geqslant 0$, if and only if

$$\forall y \in \text{Range}(A): \Pr[A(D_i) = y] \leqslant e^{\epsilon} \Pr[A(D_i') = y],$$

where Range(A) denotes the set of all possible outputs of A, and $\Pr[A(D_i) = y]$ indicates the probability where $A(D_i)$ equals y.

Differential privacy allows an adversary to know all the contents held by a contributor in any category except the target one. Conversely, the published statistics do not disclose significant knowledge on whether the contributor owns the content. A larger ϵ indicates that the contributor is less sensitive to their personal information and may contribute more accurate results. Differential privacy also follows the sequential and parallel compositional properties.

Theorem 1 (Sequential composition[36]) Let F_1, $F_2, \ldots, F_i, \ldots, F_K$ be a set of algorithms each providing ϵ_i-DP, then running in sequence all algorithms can provide $\sum \epsilon_i$-DP.

Theorem 2 (Parallel composition[36]) If G_1, $G_2, \ldots, G_i, \ldots, G_L$ are disjoint subsets of the original graph G, and $F_1, F_2, \ldots, F_i, \ldots, F_L$ are a set of algorithms each providing ϵ_i-differential privacy. Then, applying all F_i to their corresponding subset G_i can guarantee a $\max\{\epsilon_i\}$-DP for the whole graph.

The Laplace mechanism is a typical method designed for differential privacy on numerical values[20].

Theorem 3 For any function $h(\cdot): G \to R$, the randomized function $f(\cdot)$ provides ϵ-DP when

$$f(\cdot) = h(\cdot) + \text{Lap}\left(\frac{\Delta h}{\epsilon}\right) \tag{1}$$

where $\text{Lap}\left(\frac{\Delta h}{\epsilon}\right)$ follows Laplace distribution with scaling factor $\frac{\Delta h}{\epsilon}$, and Δh refers to the global sensitivity of function $h(\cdot)$: $\max |h(G_i) - h(G_i')|, \forall G_i$, where G_i and G_i' are two graphs with the same vertex set and different on just one edge.

Based on the definition, the published graph statistics should not be significantly diverse on neighboring graphs. Our framework considers link neighboring, which means that two graphs $\mathcal{G}$ and $\mathcal{G}'$ are neighboring graphs when they only diverge on one edge. The parameter ϵ indicates the degree of differential privacy, with a larger ϵ, the data owner is less sensitive, and the published graph is closer to ground truth.

3.3 Optimization goal

In our system, the combined graph is derived from all contributors by a data requestor. The distribution of node degrees is used as a reference by our framework. Such information can provide the data requestor with an overview of the network. However, only an approximate distribution is received by the data requestor because of privacy concerns. Therefore, the utility of the data requestor is expressed in the following:

$$\sum_{i=1}^{K} \frac{|\deg(v_i)' - \deg(v_i)|}{|\deg(v_i)|} \tag{2}$$

where $\deg(v_i)$ and $\deg(v_i)'$ denote the degree of v_i in the original and observed graph, respectively. Formula (2) means that the data requestor expects the observed degree of each vertex to be close to its true value.

Meanwhile, the platform or the contributors must carefully publish their contents to preserve the privacy of each single link. The platform selects a set of graphs to publish, namely, $I_j = 1$, where G_j is published; otherwise, $I_j = 0$. Each graph is published as an integrated body.

Each graph is perturbed with differential privacy to guarantee privacy. Therefore, the total differential privacy should be maintained when one edge is published multiple times,

$$\sum \epsilon_k \leqslant \epsilon_0, \ \forall k, e_j \in G_k, I_k = 1 \tag{3}$$

Generally, this procedure aims to select a subset of graphs, such that the privacy on each edge is strictly preserved under given budgets, and the accuracy of the outputting degrees is optimized,

$$\min \quad \sum_{i=1}^{K} \frac{|\deg(v_i)' - \deg(v_i)|}{|\deg(v_i)|} \tag{4}$$

s.t.

$$\sum \epsilon_k \leqslant \epsilon_0, \forall k, e_j \in G_k, I_k = 1 \tag{5}$$

$$I_j \in \{0, 1\}, \forall j \in \{1, 2, \ldots, M\} \tag{6}$$

4 Solutions for Data Publication with a Data Coordinator

First, the complexity of the proposed problem is

analyzed. Then, the whole procedure of data publication is given. Lastly, the performance of the proposed framework is examined in detail.

4.1 Complexity analysis

The degree of each edge is perturbed for privacy preservation in accordance with the scheme of graph publication. Specifically, the edge set that belongs to one graph G_i is perturbed with one single budget ϵ_i to ensure global performance. Therefore, $e_l \in E_i$ is assumed to appear N times in the published graph sets. Then, the privacy budget for e_l in each graph G_i should not be larger than ϵ_0/N according to the sequential properties of differential privacy.

As the graph is published as an integrated body, the privacy budget is determined in terms of its lowest bound,

$$\epsilon_{G_i} = \frac{\epsilon_0}{\mathrm{MAX}} \tag{7}$$

where

$$\mathrm{MAX} = \max ||\{G_k\}|e_l \in G_i, e_l \in G_k, I_k = 1||, \forall e_l \tag{8}$$

Therefore, this procedure aims to maximize the privacy budget for all edges because the variance is determined by ϵ_{e_l}.

We can further prove that the variance is minimized when the global budget is equally partitioned for multiple graphs. Generally, the problem is the same as the following to achieve the optimal accuracy: a set of G_i is selected, such that each edge is included in at least one graph, and the maximum appearance of any edge in all the selected graphs is globally minimized. The problem is formulated as follows:

$$\min \max \sum_i \mathrm{Mem}_e(e_l, G_j) \cdot I_j \tag{9}$$

s.t.

$$I_j \in \{0, 1\}, \forall j \in \{1, 2, \ldots, M\} \tag{10}$$

$$\sum_i \mathrm{Mem}_e(e_l, G_j) \cdot I_j \geqslant 1, \forall e_j \in \mathcal{G} \tag{11}$$

where $\mathrm{Mem}_e(e_l, G_j)$ means the edge e_l exists in graph G_j.

The problem is reduced from the MMSC, which is verified as NP-complete. Each graph G_i is considered as an individual set, where each edge is an element of the set. Then, $\mathcal{G}$ is the union of all sets. The problem is converted to search for a group of individual sets covering $\mathcal{G}$, while the membership of all edges in the selected group is minimized. This modified problem is the same as the definition of MMSC. Therefore, the proposed problem is NP-complete.

Theorem 4 The problem of selecting optimal set of graphs for publication is NP-complete.

4.2 Novel framework for privacy-preserving overlapped graph publication

The overview of the framework is initially introduced. Then, each component is described in detail.

4.2.1 Overview

The whole process of data publication is composed of four major steps. In the first step, all contributors upload their graph data to the data coordinator, who fuses these graphs and generates the combined $\mathcal{G}$. In the second step, the data coordinator carefully selects a set of graphs for publication and guarantees that the published graph sequence covers all edges in $\mathcal{G}$. The total budget spent on an arbitrary edge is less than the budget ϵ_0. Moreover, it tries to minimize the membership of each edge. In the third step, the data owners perturb and encode the published graph. They subsequently share the outputs with the data requestor. Lastly, the data requestor generates the graphs based on the received sequences.

4.2.2 Data fusion

In the first step, the data coordinator aggregates graphs from all contributors and generates the combined graph $\mathcal{G}$. The collected graph is assumed as a set of $\{G_1, G_2, \ldots, G_M\}$. In our framework, the data coordinator serves as a trusted service provider, so G_i is identical to the one locally held by O_i. The data coordinator then produces the combined graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ in two steps. It first combines and creates the vertices set $\mathcal{V}$,

$$\mathcal{V} = V_1 \bigcup V_2 \bigcup \cdots \bigcup V_M.$$

Then, the edge set $\mathcal{E}$ is constructed accordingly as follows:

$$\mathcal{E} = E_1 \bigcup E_2 \bigcup \cdots \bigcup E_M,$$

where the interaction among two arbitrary vertices exists in the constructed edge set once it is included in at least one of the graphs.

4.2.3 Graph selection

In the second step, the data coordinator carefully determines the set of graphs that will be published to the data requestor. The selection aims to cover all edges in $\mathcal{E}$, and their maximum membership is globally minimized.

A heuristic algorithm named **Overlapped Graph Covering Selection** (OGCS) is proposed, considering the original problem to be NP-complete. OGCS initially solves the linear programming version of the problem

formulated in Formula (9). It is achieved by relaxing the constraint in Formula (10) to $I_j \in [0,1], \forall j \in \{1,2,\ldots,M\}$.

After the results of the linear programming problem are derived, the selected graph set should also be derived. Ideally, many rounding algorithms can be applied to the selection, and approximation ratios can be guaranteed. However, they cannot ensure that each edge deterministically belongs to the published set. Therefore, OGCS applies a heuristic strategy for selection. First, all graphs are sorted in descending order based on the values of $I_1, I_2, \ldots, I_M$. Graphs are iteratively selected by OGCS from the beginning of the list and moved the selected list. The iteration is terminated once all edges in $\mathcal{E}$ are included by at least one selected graph. Lastly, the selected graph set $\mathcal{L} = \{G_{s1}, G_{s2}, \ldots, G_{sP}\}$ is constructed and fed to the third phase, where P stands for the number of selected graphs.

4.2.4 Graph perturbation

In the third step, OGCS determines the privacy budgets applied to each graph, accordingly perturbs the graph, and publishes the encoded graph to the data requestor.

OGCS initially estimates the privacy degree for each graph. The algorithm mainly aims to ensure that the total privacy budget spent on each edge is no more than ϵ_0; as such, OGCS first estimates the total appearance of each edge e_l in a published graph set $\mathcal{L}$ and derives Card $(e_l, \mathcal{L})$, which records the number of graphs in $\mathcal{L}$ containing e_l set. e_m is assumed to have the largest Card$(\cdot, \mathcal{L})$ among all edges in G_i. Then, OGCS sets the privacy budget for G_i as $\dfrac{\epsilon_0}{\text{Card}(e_m, \mathcal{L})}$.

In the encoding and perturbing step, OGCS first encodes G_i into its degree set $\{\deg(v_1), \deg(v_2), \ldots, \deg(v_{|V_i|})\}$. Then, the degree set is perturbed with the typical Laplace mechanism. Specifically, $\Delta \deg(\cdot) = 1$ as the change in one edge can modify the degree of a vertex by at most 1. The perturbed set is denoted as $\{\deg(v_1)', \deg(v_2)', \ldots, \deg(v_{|V_i|})'\}$. Lastly, the data owners publish the perturbed sequences to the data is requestor, i.e., $\{\deg(v_1)', \deg(v_2)', \ldots, \deg(v_{|V_i|})'\}$ is published for each selected graph G_i.

4.2.5 Graph construction

In the final step, the data requestor receives the published sets from the coordinator. As for each vertex, the final degree is estimated,

$$\deg_0(v_i) = \frac{\sum_{j=1}^{M} \deg(v_i') \times \text{Mem}_v(v_i, G_j) I_j}{\sum_{j=1}^{M} \text{Mem}_v(v_i, G_j) I_j} \quad (12)$$

where $\text{Mem}_v(v_i, G_j)$ indicates whether v_i belongs to graph G_j, once the estimated degrees of all vertices are derived, the data requestor can use various techniques to construct a graph or directly apply the degrees for analysis.

4.3 Performance analysis

In this section, the correctness and privacy preservation of the proposed framework are discussed.

4.3.1 Correctness

The proposed framework provides two major properties, namely, the completeness of the derived results and an unbiased estimation of the degree of each vertex. In the first property, OGCS ensures that its degree is published in at least one graph for each vertex. This property is essential because the data requestor expects to create the whole graph. The statistical analysis on the whole graph is meaningful and reliable. As for the second property, the proposed framework estimates the degree of each vertex based on the property of the Laplace mechanism. Generally, the following conclusion is obtained.

Theorem 5 The proposed framework ensures that the privacy-preserving degree is published at least once for each vertex, and the estimated degrees are unbiased for the data requestor.

4.3.2 Privacy preservation

The proposed framework provides differential privacy for the published graphs. First, each single vertex is preserved under differential privacy through the sequential and parallel properties shown in Theorems 1 and 2. Second, the privacy budget for an arbitrary should not be more than ϵ_0. This property is proven in Lemma 1.

Lemma 1 The total privacy budget spent on any edge e_l is no more than ϵ_0 in the proposed framework.

Proof Assume an edge e_l appears in published graph $\{G_1, G_2, \ldots, G_H\}$, and the corresponding privacy budgets are $\{\epsilon_1, \epsilon_2, \ldots, \epsilon_H\}$. Then we must have $\sum \epsilon_i / H \leqslant \epsilon_0 / H$, which means the average privacy budget is no more than ϵ_0 / H.

This proposition is proven by contradiction. Otherwise, there must be at least one G_i with $\epsilon_i > \epsilon_0 / H$. However, we have $\epsilon_i \leqslant \epsilon_0 / H$ as e_l belongs to G_i and appears H times in all published graphs. Therefore, the assumption is contradicted with the fact, which means $\sum \epsilon_i / H \leqslant \epsilon_0 / H$, i.e.,

$$\sum \epsilon_i \leqslant \epsilon_0.$$

■

Therefore, the proposed framework can preserve each contributor under the requested degree of differential privacy, which is concluded in Theorem 6.

Theorem 6 The proposed framework provides each data owner with ϵ_0-DP on his graph data.

5 Discussion on a Solution Without Data Coordinators

This section provides some insights into the graph publication without a trusted coordinator. It first proposes some assumptions on the problem and then gives ideas on data publication.

In more general cases, the existence of a data coordinator cannot be ensured to fuse and schedule the data publication. Therefore, data owners should locally decide if their graphs should be shared, and identify the degree of differential privacy that should be injected. Moreover, data owners own knowledge of their neighbors; therefore, their graph data are usually limited in depth. Based on these facts, the following assumptions are added to the case.

Assumption 1 Data owners know the contacts not only between themselves and their neighbors but also among their neighbors.

Assumption 2 Data owners have no idea about other owners who have no direct contact with them or their direct neighbors.

On the basis of these assumptions, each data owner locally processes and publishes their graphs, and the data requestor should reconstruct the combined graph according to these local graphs. This section investigates two types of strategies.

In the first strategy, each vertex simply publishes a unique degree in the graph, i.e., the output of an owner O_i is $\deg(v_i)$, where v_i indicates owner O_i is in the combined graph. Afterward, the data requestor collects all outputs from all owners and reconstructs the graph by using some existing methods.

The strategy has several advantages. For instance, owners evaluate their contents and publish their results without considering others' behaviors. Moreover, the privacy budget of each owner can be set as $\epsilon_0/2$, because one single edge is applied by two owners.

However, the strategy has some disadvantages. First, it requests the underlying graphs to be exactly tied to data owners, i.e., one vertex per owner. Then, the utility of the strategy is limited when the underlying graph is more sophisticated and composed of more vertices, which are similar to the knowledge graph. Second, simply publishing the single degree discloses the identity of the data owner, which is unexpected. Spamming attacks or knowledge on centrality is learned from the data owner even if the vertex can be anonymized.

A second strategy is proposed by considering both challenges. In this strategy, each data owner initially publishes subgraphs instead of a single degree. Specifically, the central vertices that can be determined by data owners are selected. Then, the subgraph is constructed by involving the central vertex, its direct neighbors, and all edges among these neighbors. The degrees of vertices are subsequently encoded and perturbed within the subgraph. As for privacy preservation, each edge is recognized by direct neighbors of both vertices (according to Assumption 1). Consequently, the privacy budget can be set as ϵ_0/T, where T is the number of the joint neighbors of two vertices and can be exchanged and shared via communication.

Lastly, the data requestor receives the perturbed sequences from data owners and reconstructs the combined graph through many approaches. For example, a data requestor can extend and reconstruct individual graphs one by one. The detailed processing and adjustment of the reconstruction will be described in our future study.

6 Conclusion

In this study, the problem of distributed graph publication under differential privacy is investigated. The graphs held by distributed data owners overlap and may include sensitive links. As such, a privacy-preserving framework for data publication is proposed. The framework initially sets a data coordinator to select from data owners and determine the published graphs. The selected graphs are then perturbed with differential privacy and shared with a data requestor. In the third step, the final combined graph is derived from all the received graphs by the data requestor. Graph selection is formulated as a problem of the MMSC, which is proven to be NP-complete. A heuristic algorithm is designed accordingly. The performance and correctness of the framework and the guarantee on differential privacy for edges among all data owners are extensively analyzed. Graph publication in the absence of coordinators is also discussed in terms of its ability to guide the local data publication. Some assumptions and insights into the design of the framework are proposed.

Acknowledgment

This work was supported by the National Natural Science Foundation of China (Nos. U19A2059 and 61802050) and Ministry of Science and Technology of Sichuan Province Program (Nos. 2021YFG0018 and 20ZDYF0343).

References

[1] M. S. Mahmud, J. Z. Huang, S. Salloum, T. Z. Emara, and K. Sadatdiynov, A survey of data partitioning and sampling methods to support big data analysis, *Big Data Mining and Analytics*, vol. 3, no. 2, pp. 85–101, 2020.

[2] U. Kang and C. Faloutsos, Big graph mining: Algorithms and discoveries, *ACM SIGKDD Explorat. Newsl.*, vol. 14, no. 2, pp. 29–36, 2013.

[3] J. Wu and N. Wang, Approximating special social influence maximization problems, *Tsinghua Science and Technology*, vol. 25, no. 6, pp. 703–711, 2020.

[4] Q. X. Hou, M. Han, and Z. P. Cai, Survey on data analysis in social media: A practical application aspect, *Big Data Science and Technology*, vol. 3, no. 4, pp. 259–279, 2020.

[5] M. Togninalli, E. Ghisu, F. Llinares-López, B. Rieck, and K. Borgwardt, Wasserstein Weisfeiler-Lehman graph kernels, in *Proc. 33rd Int. Conf. Neural Information Processing Systems*, Vancouver, Canada, 2019, pp. 6439–6449.

[6] L. Y. Liu, B. C. Yu, M. Han, S. S. Yuan, and N. Wang, Mild cognitive impairment understanding: An empirical study by data-driven approach, *BMC Bioinformatics*, vol. 20, no. 15, p. 481, 2019.

[7] H. Kim, A. Bhattacharyya, and K. Anyanwu, Semantic query transformations for increased parallelization in distributed knowledge graph query processing, in *Proc. Int. Conf. for High Performance Computing, Networking, Storage and Analysis*, Denver, CO, USA, 2019, pp. 1–14.

[8] K. Ueta, X. Y. Xue, Y. Nakamoto, and S. Murakami, A distributed graph database for the data management of IoT systems, in *Proc.* 2016 *IEEE Int. Conf. Internet of Things* (*iThings*) *and IEEE Green Computing and Communications* (*GreenCom*) *and IEEE Cyber, Physical and Social Computing* (*CPSCom*) *and IEEE Smart Data* (SmartData), Chengdu, China, 2016, pp. 299–304.

[9] Z. P. Cai and X. Zheng, A private and efficient mechanism for data uploading in smart cyber-physical systems, *IEEE Trans. Netw. Sci. Eng.*, vol. 7, no. 2, pp. 766–775, 2020.

[10] K. Al Fararni, F. Nafis, B. Aghoutane, A. Yahyaouy, J. Riffi, and A. Sabri, Hybrid recommender system for tourism based on big data and AI: A conceptual framework, *Big Data Mining and Analytics*, vol. 4, no. 1, pp. 47–55, 2021.

[11] J. Li, X. Pei, X. J. Wang, D. Y. Yao, Y. Zhang, and Y. Yue, Transportation mode identification with GPS trajectory data and GIS information, *Tsinghua Science and Technology*, vol. 26, no. 4, pp. 403–416, 2021.

[12] X. Y. Li, C. H. Zhang, T. Jung, J. W. Qian, and L. L. Chen, Graph-based privacy-preserving data publication, in *Proc. 35th Ann. IEEE Int. Conf. Computer Communications*, San Francisco, CA, USA, 2016, pp. 1–9.

[13] W. Y. Day, N. H. Li, and M. Lyu, Publishing graph degree distribution with node differential privacy, in *Proc. 2016 Int. Conf. Management of Data*, San Francisco, CA, USA, 2016, pp. 123–138.

[14] Z. B. He and J. X. Zhou, Inference attacks on genomic data based on probabilistic graphical models, *Big Data Mining and Analytics*, vol. 3, no. 3, pp. 225–233, 2020.

[15] Z. P. Cai and Z. B. He, Trading private range counting over big IoT data, in *Proc. 39th Int. Conf. on Distributed Computing Systems*, Dallas, TX, USA, 2019, pp. 144–153.

[16] L. Muñoz-González, D. Sgandurra, A. Paudice, and E. C. Lupu, Efficient attack graph analysis through approximate inference, *ACM Transactions on Privacy and Security*, vol. 20, no. 3, pp. 1–30, 2017.

[17] J. Li, M. Siddula, X. Z. Cheng, W. Cheng, Z. Tian, and Y. S. Li, Approximate data aggregation in sensor equipped IoT networks, *Tsinghua Science and Technology*, vol. 25, no. 1, pp. 44–55, 2020.

[18] K. Liu and E. Terzi, Towards identity anonymization on graphs, in *Proc. 2008 ACM SIGMOD Int. Conf. Management of Data*, Vancouver, Canada, 2008, pp. 93–106.

[19] J. B. Wang, Z. P. Cai, and J. G. Yu, Achieving personalized k-anonymity-based content privacy for autonomous vehicles in CPS, *IEEE Trans. Ind. Inform.*, vol. 16, no. 6, pp. 4242–4251, 2020.

[20] C. Dwork, Differential privacy: A survey of results, in *Proc. 5th Int. Conf. Theory and Applications of Models of Computation*, Xi'an, China, 2008, pp. 1–19.

[21] Z. Qin, T. Yu, Y. Yang, I. Khalil, X. K. Xiao, and K. Ren, Generating synthetic decentralized social graphs with local differential privacy, in *Proc. ACM SIGSAC Conf. Computer and Communications Security*, Dallas, TX, USA, 2017, pp. 425–438.

[22] R. Bassily and A. Smith, Local, private, efficient protocols for succinct histograms, in *Proc. 47th Ann. ACM Symp. Theory of Computing*, Portland, OR, USA, 2015, pp. 127–135.

[23] X. Zheng and Z. P. Cai, Privacy-preserved data sharing towards multiple parties in industrial IoTs, *IEEE J. Sel. Areas Commun.*, vol. 38, no. 5, pp. 968–979, 2020.

[24] H. P. Sun, X. K. Xiao, I. Khalil, Y. Yang, Z. Qin, H. Wang, and T. Yu, Analyzing subgraph statistics from extended local views with decentralized differential privacy, in *Proc. 2019 ACM SIGSAC Conf. Computer and Communications Security*, London, UK, 2019, pp. 703–717.

[25] Y. X. Zhang, J. H. Wei, X. J. Zhang, X. X. Hu, and W. F. Liu, A two-phase algorithm for generating synthetic graph under local differential privacy, in *Proc. 8th Int. Conf. Communication and Network Security*, Qingdao, China, 2018, pp. 84–89.

[26] T. C. Gao, F. Li, Y. Chen, and X. K. Zou, Local differential privately anonymizing online social networks under HRG-based model, *IIEEE Trans. Comput. Soc. Syst.*, vol. 5, no. 4, pp. 1009–1020, 2018.

[27] J. Zhang, C. K. Wang, J. M. Wang, J. X. Yu, J. Chen, and C. P. Wang, Inferring directions of undirected social ties, *IEEE Trans. Knowl. Data Eng.*, vol. 28, no. 12, pp. 3276–3292, 2016.

[28] C. K. Wang, C. P. Wang, Z. Wang, X. J. Ye, J. X. Yu, and B. Wang, DeepDirect: Learning directions of social ties with edge-based network embedding, *IEEE Trans. Knowl. Data Eng.*, vol. 31, no. 12, pp. 2277–2291, 2019.

[29] X. Zheng, G. C. Luo, and Z. P. Cai, A fair mechanism for private data publication in online social networks, *IEEE Trans. Netw. Sci. Eng.*, vol. 7, no. 2, pp. 880–891, 2020.

[30] T. C. Gao and F. Li, Privacy-preserving sketching for online social network data publication, in *Proc. 16th Ann. IEEE Int. Conf. Sensing, Communication, and Networking*, Boston, MA, USA, 2019, pp. 1–9.

[31] T. L. Guo, J. Z. Luo, K. Dong, and M. Yang, Differentially private graph-link analysis based social recommendation, *Inf. Sci.*, vols. 463&464, pp. 214–226, 2018.

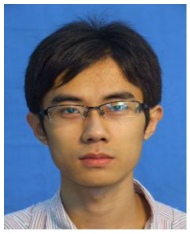
[32] C. Zhang, L. H. Zhu, C. Xu, K. Sharif, C. Zhang, and X. M. Liu, PGAS: Privacy-preserving graph encryption for accurate constrained shortest distance queries, *Inf. Sci.*, vol. 506, pp. 325–345, 2020.

[33] D. P. Xu, S. H. Yuan, X. T. Wu, and H. Phan, DPNE: Differentially private network embedding, in *Proc. 22nd Pacific-Asia Conf. Knowledge Discovery and Data Mining*, Melbourne, Australia, 2018, pp. 235–246.

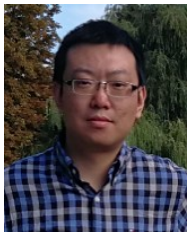
[34] S. Zhang and W. W. Ni, Graph embedding matrix sharing with differential privacy, *IEEE Access*, vol. 7, pp. 89390–89399, 2019.

[35] Z. P. Cai, X. Zheng, and J. G. Yu, A differential-private framework for urban traffic flows estimation via taxi companies, *IEEE Trans. Ind. Inform.*, vol. 15, no. 12, pp. 6492–6499, 2019.

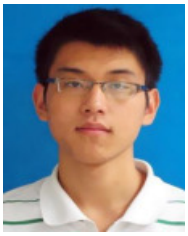
[36] F. McSherry and I. Mironov, Differentially private recommender systems: Building privacy into the netflix prize contenders, in *Proc. 15th ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, Paris, France, 2009, pp. 627–636.



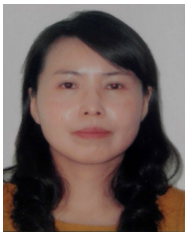
Xu Zheng received the BEng, MEng, and PhD degrees from Harbin Institute of Technology in 2010, 2012, and 2017, respectively. He received the second PhD degree from Georgia State University, USA in 2018. He is currently an assistant professor at the School of Computer Science and Engineering, University of Electronic Science and Technology of China. His research areas focus on IoTs and data security.



Lizong Zhang received the MEng and PhD degrees from Staffordshire University, USA in 2008 and 2013, respectively. He is currently an associate professor in computing science at the School of Computer Science and Engineering, University of Electronic Sciences and Technology of China. His main research interests include machine learning, knowledge management, and computer vision. His other research interests include big data, cloud computing, and artificial intelligence systems.



Kaiyang Li received the MEng degree from University of Electronic Science and Technology of China, Chengdu, China in 2016. He is currently a PhD candidate at the School of Computer Science and Engineering, University of Electronic Science and Technology of China. His research interests include data privacy and machine learning.



Xi Zeng received the BEng and MEng degrees from University of Electronic Science and Technology of China, Chengdu, China in 2006 and 2008, respectively. She is currently a research fellow at China Electronics Technology Cyber Security Co. Ltd. Her research interests include cyber security and network communication.