doi:10.3772/j.issn.1006-6748.2018.02.003

Graph publishing method based on differential privacy protection[①]

Wang Junli (王俊丽)[②], Yang Li, Wu Yuxi, Guan Min
(Department of Computer Science and Technology, Tongji University, Shanghai 201804, P. R. China)

Abstract

There are growing concerns surrounding the data security of social networks because large amount of user information and sensitive data are collected. Differential privacy is an effective method for privacy protection that can provide rigorous and quantitative protection. Concerning the application of differential privacy in social networks, this paper analyzes current trends of research and provides some background information including privacy protection standards and noise mechanisms. Focusing on the privacy protection of social network data publishing, a graph-publishing model is designed to provide differential privacy in social networks via three steps: Firstly, according to the features of social network where two nodes that possess certain common properties are associated with a higher probability, a raw graph is divided into several disconnected sub-graphs, and correspondingly dense adjacent matrixes and the number of bridges are obtained. Secondly, taking the advantage of quad-trees, dense region exploration of the adjacent matrixes is conducted. Finally, using an exponential mechanism and leaf nodes of quad-trees, an adjacent matrix of the sanitized graph is reconstructed. In addition, a set of experiments is conducted to evaluate its feasibility, availability and strengths using three analysis techniques: degree distribution, shortest path, and clustering coefficients.

Key words: differential privacy, social network, data publication

0 Introduction

In recent years, information networks have experienced a rapid growth in various fields including social networks and the internet of things, in which the performance of social networks is especially striking. Facebook and Twitter now have more than one billion active users worldwide. Large amounts of network data are collected, including much personal information and sensitive data. Analyzing these data can offer significant potential benefits, but simultaneously users' privacy may be seriously threatened[1]. Therefore, greater attention is now being paid to the security of the network data.

There are a number of available privacy protection methods, including k-anonymity and l-diversity, which are based on restricted release. First, k-anonymity designed by Sweeney[2], guarantees that any record is indistinguishable with other $k-1$ records but is also vulnerable to consistency attack[3]. To address this problem, Machanavajjhala, et al.[4] proposed an l-diversity principle: for a dataset with k-anonymity, each sensitive property of the equivalent class has at least l values so that the risk of privacy disclosure is less than $1/l$. However, this method is also vulnerable to consistency and background knowledge attacks because of the lack of a strict attack model.

In order to solve those problems, Dwork, et al.[5] proposed differential privacy, which is a relatively new notion of privacy and has become the de facto standard for a security-controlled privacy guarantee[6]. If an attacker can access the information of all other records in addition to the target record in a dataset, the sum of those messages is regarded as the maximum background knowledge that the attacker masters. Under this assumption, differential privacy can defend against a background knowledge attack. Differential privacy is also built on solid mathematical foundations which greatly ensure the availability of data by giving quantitative representation and proof to the risks of privacy disclosure. With these advantages, differential privacy has been widely applied in various fields. When applied to network data, there are two common privacy

[①] Supported by the National Natural Science Foundation of China (No. 61105047), the National High Technology Research and Development Program of China (No. 2015IM030300), the Science and Technology Committee of Shanghai Support Project (No. 14JC1405800) and the Project of the Central Universities Fundamental Research of Tongji University.

[②] To whom correspondence should be addressed. E-mail: junliwang@tongji.edu.cn

Received on Mar. 20, 2017

protection standards. One is node privacy[7], which adds or removes an arbitrary node from the original graph and all edges are connected to it. Hence, an attacker cannot determine whether an individual node is in the graph. Thus, node privacy is able to completely protect all individuals. However, it is unfeasible in many cases because it imposes extreme server restrictions on queries. The other protection standard is edge privacy, which adds or removes an edge from the original graph. An attacker cannot determine whether a relationship exists between the individual nodes though it is a high probability. It offers a weaker guarantee than node privacy but is sufficient for many applications. Edge privacy is regarded as a more reasonable approach when combining differential privacy with network data, and has been widely applied[8-11]. Task, et al. proposed out-link privacy[12], which added or removed an arbitrary node and all of its out-links, which enables many queries to be unfeasible under both node and edge privacy. Gehrke, et al. created zero-knowledge privacy using cryptography information[13], which could provide better applications in social networks.

In terms of graph mining, Hay, et al. queried private degree distribution using differential privacy[7,14]. Nissim, et al. proposed the concept of smooth sensitivity to count Laplace noise and calculate the number of triangles, k-star, and k-triangle while providing differential privacy[15]. Wang, et al. looked at clustering coefficients and applied the concept of "divide and conquer" to realize private queries[16]. There is also a number of studies regarding graph publishing. Chen, et al.[17] showed that differential privacy could provide provable privacy guarantees in a correlated setting, and proposed a method of density exploration and reconstruction (DER): a quad-tree was used to divide the adjacent matrix of the original graph into several regions, and then a sanitized graph was reconstructed with differential privacy. Sala, et al.[9] made use of a dk-model to build sanitized graph to maintain structural similarity with the original graph.

According to previous studies, when applying differential privacy to graph publishing, it is important to ensure the usefulness of the released data. Concerning differential privacy applied to social network data publication, the features of social networks are analyzed. Taking Facebook as an example, it is more likely that users will be friends with users in a common interest. Based on this consideration, a graph-publishing model called classification-based graph-publishing model (CGM) is proposed.

In detail, the work includes the following aspects:

1) According to the property of social networks and the application of differential privacy in a correlated dataset, the original graph is first divided into multiple subgraphs on the basis of node properties. Thus, the connection between edges will be stronger inside a subgraph and weaker between subgraphs.

2) For each sub-graph, a quad-tree is used to explore dense regions, and according to the leaf nodes of the quad-tree, corresponding new sub-graphs are reconstructed which are included in the whole sanitized graph.

3) Degree distribution, shortest path, and clustering coefficients are used as measurement methods. Several experiments are conducted to analyze the sanitized graph's structural consistency.

1 Social network graph-publishing problem statement

1.1 Differential privacy protection model

In this study, adjacent matrix A of graph G is used to abstractly represent a social network. Furthermore, $G = (V, E)$, $A = \begin{cases} 1 & \text{if } (v_i, v_j) \in E \\ 0 & \text{otherwise} \end{cases}$, V is a set of nodes and E is a set of edges. In a region $R \subseteq A$ and $|R| = m \times l$, the density of R is $den(R) = \sum_{i=1}^{m} \sum_{j=i}^{l} R_{ij}/ml$.

The definition of differential privacy is based on the concept of neighboring databases. Thus, for database D, D' is its neighboring database, if and only if $|D \Delta D'| = 1$ where $D \Delta D'$ denotes symmetric difference.

Definition 1[18] (ε-differential privacy) Mechanism M is ε-differential privacy for any two neighboring databases D and D'. P_M is the outputs of Mechanism M and S_M is any subset of outputs P_M. The following holds:

$$P_r[M(D) \in S_M] \leqslant \exp(\varepsilon) \times P_r[M(D') \in S_M] \qquad (1)$$

where probability P_r is the randomness of M and ε is the privacy budget.

A mechanism can satisfy differential privacy by adding appropriate noises to the results of the query functions. Two common noisy techniques are the Laplace mechanism and the exponential mechanism. A fundamental concept applied to count noise is global sensitivity.

Definition 2[19] (Global sensitivity) For function $f: D \rightarrow R^d$, the input is database D and output is real vector R^d. For any two neighboring databases D and D', the global sensitivity of f is

$$GS_f = \max_{D,D'} \| f(D) - f(D') \|_1 \qquad (2)$$

A. Laplace Mechanism

This is applied to the privacy of numerical results. Furthermore, $Lap(\sigma)$ denotes the Laplace distribution where the mean is zero and the scale parameter is σ. Its probability density function is $f(x \mid 0,\sigma) = \frac{1}{2\sigma} e^{-|x|/\sigma}$, where σ is decided by global sensitivity GS_f and privacy budget ε.

Theorem 1[20] For database D, for any function f: $D \rightarrow R^d$, global sensitivity GS_f, the mechanism M gives ε-differential privacy if it satisfies:

$$M(D) = f(D) + Lap(GS_f/\varepsilon) \tag{3}$$

B. Exponential Mechanism

This is applied to the privacy of entity objects. The input is database D and output is an entity $r \in$ *Range*. Furthermore, $q(D, r)$ denotes the utility function of r used to evaluate the result's merits, and GS_q denotes the global sensitivity of $q(D, r)$.

Theorem 2[21] For a database D, its utility function is $q(D, r)$. If mechanism M selects result r as the output from *Range* with a probability proportional to $\exp\left(\frac{\varepsilon q(D, r)}{2GS_q}\right)$, M gives ε-differential privacy.

Using the sequential composition of differential privacy, McSherry[22] divides the original graph into several subgraphs, each of which provides differential privacy. Combining these subgraphs can also provide differential privacy.

Theorem 3[22] For mechanisms $M_1, M_2, \cdots, M_n$, the corresponding privacy budget is $\varepsilon_1, \varepsilon_2, \cdots, \varepsilon_n$, and when applying to the same database D, a sequence of $M(M_1(D), M_2(D), \cdots, M_n(D))$ gives $(\sum_{i=1}^{n} \varepsilon_i)$-differential privacy.

1.2 Graph-publishing problem statement

The goal is to generate a sanitized graph that maintains structural consistency with the raw graph so that the published data is useful. Several analysis technologies are used to assess its utility, as detailed below.

A. Degree distribution

Degree distribution is a critical concept in network theory, and is a widely studied feature of graphs. Given graph $G = G(V, E)$, $V = \{v_1, v_2, \cdots, v_n\}$ is the set of nodes and $E = \{e_{i,j} \mid i = 1,2,\cdots,n, j = 1,2,\cdots, n\}$ is the set of edges. Then the degree of node v_i is the number of all edges connected to it.

Definition 3 (Degree frequency sequence) Given graph G, $d(v)$ represents the degree of node v and $f(G)$ denotes the degree frequency sequence of G. The i-th value in $f(G)$ is $\frac{|\{v_i \in v: d(v_i) = i\}|}{|V|}$.

Given the degree frequency sequence of the original graph G and sanitized graph G', $f(G)$ and $f(G')$, their difference is measured with Kullback - Leibler divergence which is also called relative entropy, measuring the difference between two probability distributions of the same event space.

$$D_{KL}(f(G), f'(G)) = \sum_{i=0}^{|V|-1} f(G)[i] \log \frac{f(G)[i]}{f'(G)[i]} \tag{4}$$

B. Shortest path

In graph G, if node v_i and v_j are reachable, a path between the two nodes can be defined as a sequence $(v_i, v_{i+1}, \cdots, v_{j-1}, v_j) \in V \times V \times \cdots \times V$, where $(v_i, v_{i+1}) \in E$. The length of the path is the number of edges involved in its sequence. Then, the shortest path between two nodes v_i and v_j is the minimum length of all paths between them. If nodes v_i and v_j are not reachable, the length of the shortest path is denoted as ∞.

C. Clustering coefficient

This coefficient indicates the nodes' degree of aggregation in a graph. The formula is

$$C_i = \frac{N_\Delta(i)}{N_3(i)} = \frac{N_\Delta(i)}{d_i(d_i - 1)/2} \tag{5}$$

where C_i denotes the clustering coefficient of node i, $N_\triangle(i)$ denotes the number of triangles involved in node i, $N_3(i)$ is the number of connected triples with node i in the center, and d_i shows the degree of node i.

Given the clustering coefficient of the original graph and sanitized graph, $C(G)$ and $C(G')$, the utility loss over G' is measured by its relative error:

$$error(G(G')) = \frac{|C(G) - C(G')|}{C(G)} \tag{6}$$

D. Betweenness centrality

Betweenness reflects the role and influence of respective nodes or edges in the network, which contains edge betweenness and node betweenness. Edge betweenness is defined as the proportion between the number of shortest paths through the edge and the total number of shortest paths in the network. Node betweenness is defined as the proportion between the number of shortest paths through the node and the total number of shortest paths in the network. Given the edge and node betweenness of the original graph and sanitized graph, $B(G)$ and $B(G')$, the similarity is measured by its Euclidean distance.

$$sim(B(G), B(G')) = \frac{1}{1 + E(B(G), B(G'))} \tag{7}$$

where $E(B(G), B(G'))$ denote Euclidean distance of $B(G)$, $B(G')$. The formula is:

$$E(B(G), B(G')) = \sqrt{(\sum (B(G)_i - B(G')_i)^2)} \tag{8}$$

2 Classification-based graph publishing model and algorithm

This study investigates the application of differential privacy in a correlated data setting. Most data in a social network are related. A database with correlation parameter k means that any record in the database is correlated to at most $k-1$ other records. With privacy budget ε, at most k records are divided into a group. Taking a "group" as a unit, it is sufficient to eliminate the effect of data correlation on any computation. Then it is possible to provide ε/k-differential privacy. The CGM algorithm is shown as Algorithm 1. The algorithm procedure of CGM is composed of three steps:

1) Graph classification: In this procedure, the goal is an adjacent matrix which is formed of dense clusters. Regarding a social network, two nodes with a certain characteristic are more likely to be associated. According to this feature and the concept of "bridge", the raw graph is classified into multiple subgraphs. It is then more likely that the adjacent matrix of the subgraph will be dense. The number of bridges should also be recorded at the same time.

2) Exploration of the dense region: For each subgraph, a standard quad-tree is used and the dense region is explored to reconstruct a sanitized adjacent matrix with greater accuracy. This procedure produces a noisy quad-tree where the nodes of the tree indicate region size and a noisy count of 1s. The main challenge is to determine the height of the tree and select the splitting point.

Algorithm 1 CGM algorithm

```
Input: Raw graph $G$ , Privacy budget $\varepsilon$ , Correlation parameter $k$
Output: Sanitized graph $G'$
1. $\frac{\varepsilon}{k} = \varepsilon_c + \varepsilon_E + \varepsilon_R$
2. Subgraphs $C_i \leftarrow$ Graph classification ( $G, \varepsilon_C$ )
3. Generate the adjacent matrix $A_i$ from $C_i$
4. Noisy quad-tree $QT_i \leftarrow$ Explore dense region ( $A_i, \varepsilon_E$ )
5. Sanitized matrix $A_i{}' \urcorner$ Reconstruct adjacent matrix ( $QT_i$ , $A_i, \varepsilon_R$ )
6. Reconstruct sanitized graph $G'$ based on $A_i{}'$
7. Return $G'$
```

3) Reconstruction of the adjacent matrix: According to the leaf nodes of the quad-tree, and the exponential mechanism is used to arrange the distribution of 1s in corresponding regions to construct the sanitized adjacent matrix.

The input of the model is graph G, privacy budget ε and correlation parameter k. The output is the sanitized graph G'. Throughout the whole procedure, the sum of the privacy budget is ε/k. It is divided into ε_C, ε_E, ε_R, corresponding to every step.

2.1 Graph classification

The different vertex labels of the graph will have different adjacent matrixes. In a social network individual users with common interests are more likely to have relationships. Thus, the idea of "classification" is used to divide the raw graph into multiple disconnected subgraphs. The adjacent matrix of a subgraph will be dense clusters of 1s.

The edges connecting the two subgraphs are "bridges". Given graph G, E' is the cut set of G if the following holds: ($E' \subseteq E$, $P(G - E') > P(G)$ and ($E'' \subseteq E'$, $P(G - E'') = P(G)$ where $P(G)$ denotes the number of connected components. If there exists $E' = \{e\}$, e is a bridge. Kosaraju algorithm is an algorithm calculating a strongly connected component, which can find the strongly connected component in a time complexity of $O(V + E)$. The process is as follows: First, it conducts a depth-first search in graph G and count time T when each node has been searched. Second, it conducts a depth-first search in the inverse graph GT. At this moment, the searching order is decided by the values of T, not by the vertex labels. Finally, the obtained forest from the inverse graph GT is the corresponding connected component.

While classifying the raw graph, the noisy count of bridges is recorded. The basic flow of "graph classification" is as follows.

Procedure 1 Graph classification procedure

```
Input: Raw graph $G$, Privacy budget $\varepsilon_C$
Output: Subgraph $C_i$, $Br'(G)$
  1. Apply Kosaraju algorithm to $G$
  2. Generate Subgraph $C_i$ and the corresponding bridge $Br(G)$
  3. Get $Br'(G)$ with $\varepsilon_C$
  4. Return $C_i$ and $Br'(G)$
```

When calculating the number of bridges $Br'(G)$, Laplace noise is added to provide differential privacy. Furthermore, $GS(Br)$ denotes the global sensitivity of $Br(G)$, and satisfies $GS(Br) = 1$.

$$Br'(G) = Br(G) + Lap(GS(Br) / \varepsilon_C) \tag{9}$$

Taking Fig. 1 as an example, it ' s the original graph. After the processing, a sanitized graph will be generated, and the feasibility of this model will be evaluated through further experiments.

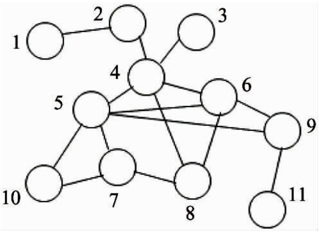


Fig. 1 A sample graph

Example 1 Considers the graph in Fig. 1. The possible classifying results are as follows $C_1 = \{1,2\}$, $C_2 = \{3\}$, $C_3 = \{4,5,6,7,8,9,10,11\}$, the noisy count of bridges is 3 which may includes edge (4, 3).

2.2 Reconstruction of adjacent matrix

After carrying out the first step, the corresponding adjacent matrix of each sub-graph is obtained. Next, those regions that are the most dense or sparse using a quad-tree are explored. To create a quad-tree, a two-dimensional space is divided into four subspaces iteratively. The process is as follows. First, according to the condition of leaf nodes, calculate the height of the tree. Regarding those non-leaf regions, select the best splitting point using the exponential mechanism, making use of the following maximal density contrast function $q(R,\ p) = \max_{\forall R' \in \Re}(den(R')) - \min_{\forall R' \in \Re}(den(R'))$ where $R \subseteq A$, p is one splitting point over R and R is the set of subregions of R resulting from p. The nodes in the tree are made up of two segments: the region R and the noisy count of 1s within the region, denoted by $\tilde{c}$.

Example 2 Considers the subgraph $C_3 = \{4,5,6,7,8,9,10,11\}$ of the graph in Fig. 1 and its adjacency matrix in Table 1. Suppose the height of quad-tree is calculated to 2. The first possible parting result is $R_1 = [4:7,4:7]$, $R_2 = [4:7,8:11]$, $R_3 = [8:11,4:7]$, $R_4 = [8:11,8:11]$. According to the noisy count, CGM model needs to further partition R_2. After that, the corresponding quad-tree is illustrated in Fig. 2.

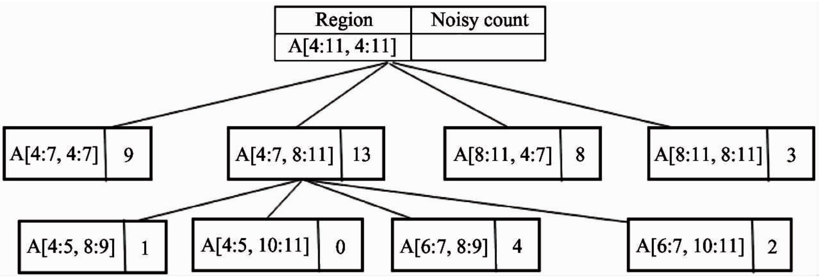


Fig. 2 A sample quad-tree of C_3

Table 1 The adjacency matrix of C_3

	4	5	6	7	8	9	10	11
4	0	1	1	0	1	0	0	0
5	1	0	1	1	0	1	1	0
6	1	1	0	0	1	1	0	0
7	0	1	0	0	1	0	1	0
8	1	0	1	1	0	0	0	0
9	0	1	1	0	0	0	0	1
10	0	1	0	1	0	0	0	0
11	0	0	0	0	0	1	0	0

In the first step, adopt global sensitivity to compute the noise of the bridges to provide differential privacy. The process of proof-satisfying differential privacy in the second and third step is based on Chen, et al.[17]. Thus, the procedure of constructing every sanitized graph conforms to the condition of differential privacy. Finally, according to the sequential composition, combine each sanitized subgraph into the whole sanitized graph. The noisy bridges are also randomly added to the graph. Then the entire process of constructing a sanitized graph is completed. Considering the graph in Fig. 1, After using CGM model ,the possible sanitized graph may be Fig. 3.

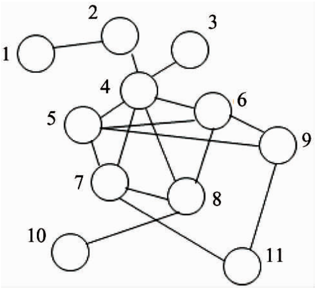


Fig. 3 The sanitized graph

3 Experimental evaluation

All experiments in this paper are simulated on PC (Intel (R) Core (TM) i7-8700 CPU, 8GHz, 16GRAM), the operating system is Windows 10 with Visual C + +6.0 coding environment.

The CGM ' s feasibility is evaluated via experiments using degree distribution, shortest path and clustering coefficients, and compared with the DER method to assess their advantages or disadvantages. The datasets used in the experiments are ego-Gplus, ego-Facebook, Wiki-vote and soc-Epinions1, which are from the Stanford Network Analysis Project. Ego-Gplus consists of "circles" from Google +, where data were collected from users who had manually shared their circles using the "share circle" feature. Ego-Facebook consists of "circles" or "friends lists" from Facebook, where data were collected from survey participants using the Facebook app. Wiki-vote contains the Wikipedia voting data from the inception of Wikipedia. Nodes in the network represent Wikipedia users and a directed edge from node *i* to node *j* represents that user *i* voted on user *j*. The data of soc-Epinions1 comes from a general consumer review site about "who-trust-whom". All the trust relationships of members in the site interact and form the Web of Trust. According to the Trust Web, reviews that are shown to the users could be determined through review rating.

Applying CGM and DER methods to ego-Gplus, ego-Facebook, Wiki-vote and soc-Epinions1, corresponding sanitized graphs are obtained. Making use of those metrics in Section 2.2, the following experimental results are obtained where the correlation parameter is $k = 15$ and the values of privacy budget ε are 0.6, 0.7, 0.8, 0.9, and 1. Table 2 shows datasets in the experiments.

Table 2 Experimental datasets

Datasets	0.2|V|	0.4|V|	0.6|V|	0.8|V|	|V|
ego-Gplus	21522	43044	64566	86088	107614
ego-Facebook	807	1615	2421	3288	4039
Wiki-vote	1423	2846	4269	5692	7115
soc-Epinions1	15175	30351	45527	60703	75879

In the first set of experiments, the utility of the clustering coefficient is examined in terms of the average relative error. The related parameters are similar to the previous settings. Table 3 shows the experimental results in ego-Gplus, ego-Facebook, Wiki-vote and soc-Epinions1. With the increase of the privacy budget, the average relative error gradually decreases. In ego-Facebook Dataset, the error shows a steady downward trend, which indicates that CGM and DER approach work better. In Wiki-vote Dataset, the applicability is poor. When the privacy budget is larger, the query results of CGM model have a higher accuracy.

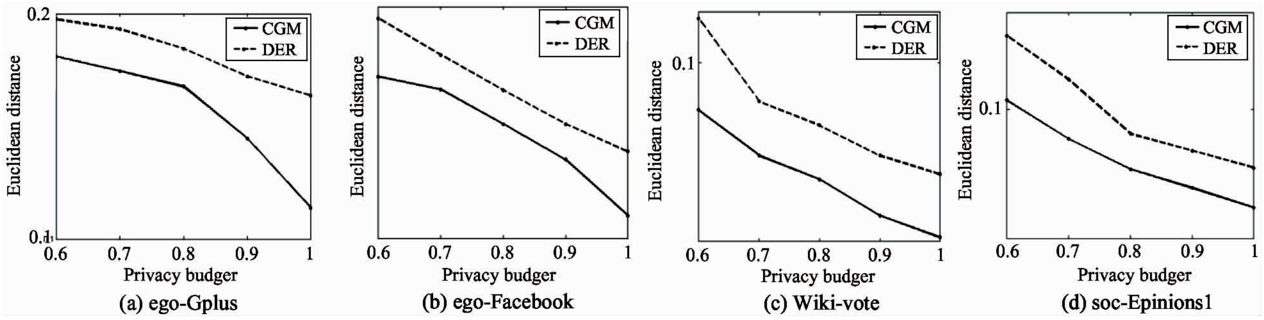
Table 3 Some experimental results

Measurement	privacy budget	DER (ego-Gplus)	CGM (ego-Gplus)	DER(ego-Facebook)	CGM(ego-Facebook)	DER (Wiki-vote)	CGM (Wiki-vote)	DER(soc-Epinions1)	CGM(soc-Epinions1)
Average relative error of clustering coefficient	0.6	0.58	0.51	0.56	0.44	0.68	0.61	0.49	0.45
	0.7	0.45	0.35	0.42	0.35	0.6	0.52	0.45	0.37
	0.8	0.35	0.3	0.3	0.28	0.52	0.48	0.4	0.29
	0.9	0.22	0.18	0.23	0.17	0.5	0.3	0.35	0.2
	1	0.16	0.05	0.12	0.06	0.45	0.2	0.25	0.15
Average relative error of the shortest path	0.6	0.8	0.7	0.6	0.59	0.68	0.61	0.58	0.42
	0.7	0.63	0.6	0.5	0.4	0.65	0.55	0.5	0.4
	0.8	0.4	0.35	0.4	0.32	0.59	0.5	0.43	0.3
	0.9	0.35	0.24	0.25	0.21	0.5	0.4	0.32	0.22
	1	0.18	0.21	0.18	0.1	0.43	0.35	0.3	0.15
KL-divergence of degree distribution	0.6	0.94	0.81	0.72	0.67	0.82	0.75	0.83	0.6
	0.7	0.85	0.78	0.6	0.56	0.8	0.66	0.78	0.57
	0.8	0.69	0.62	0.57	0.43	0.75	0.58	0.7	0.44
	0.9	0.62	0.6	0.48	0.37	0.69	0.49	0.59	0.36
	1	0.52	0.5	0.36	0.3	0.6	0.39	0.51	0.3

In the second set of experiments, the utility of the shortest path of the sanitized graph is demonstrated by the average relative error. From Table 3 it can be seen that the CGM ' s error value shows a steady declining trend. And compared with DER method, the proposed method produces a lower error values and has stronger usefulness in Wiki-vote and soc-Epinions1.

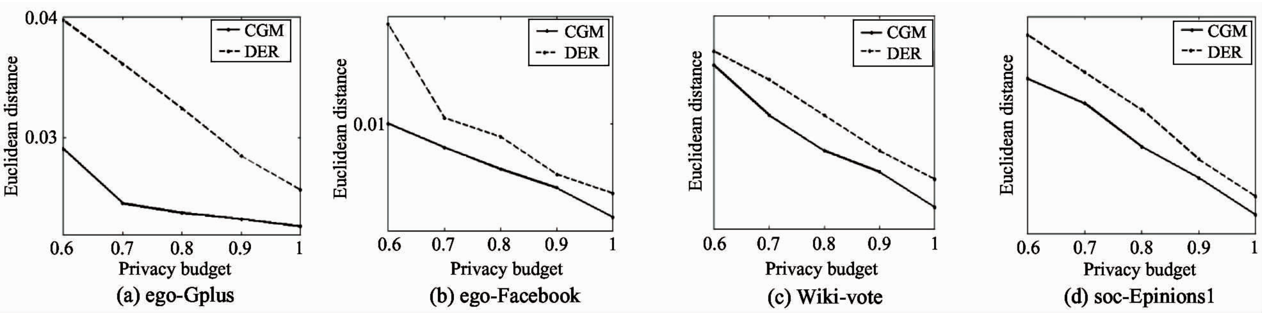
In the third set of experiments, the utility of the sanitized data for the degree distribution are measured by KL-divergence. In Table 3, it can be observed that the KL divergences of CGM are smaller than DER in all settings. Especially in soc-Epinions1 Dataset, with the increase of privacy budget, the distance value is generally smaller. Therefore, this approach is more suitable to preserve degree distribution.

Also, the structure consistency between published graph and original graph is measured by the number of edges and the number of nodes. The results of the experiment are shown in Fig. 4 and Fig. 5. It can be seen from the results that the Euclidean distance of the edge betweenness and the node betweenness is smaller, indicating the higher similarity. With the increase of the privacy budget, the distance values are steadily decreasing, and there is no obvious gap between the CGM and the DER methods. In the case of small budget value, the difference is obvious. With the increase of the budget value, the gap gradually decreases. Fig. 5(a) shows that the CGM method is more applicable on the ego-Gplus dataset.



(a) ego-Gplus (b) ego-Facebook (c) Wiki-vote (d) soc-Epinions1

Fig. 4 Euclidean distance of edge betweenness



(a) ego-Gplus (b) ego-Facebook (c) Wiki-vote (d) soc-Epinions1

Fig. 5 Euclidean distance of node betweenness

The above experimental results show that when CGM is applied in a social network, the relative error values for degree distribution, shortest path, clustering coefficient and betweenness centrality are relatively contained within a range. The value of Euclidean distance in calculating edge and node betweenness are much smaller. Thus, the sanitized graph has structural consistency with the raw graph. The results of the CGM show that the average relative error of the shortest path and clustering coefficient and the maximum difference of KL-divergence are better than that of the DER method. However, the accuracy of the query results could be improved. Future research will be conducted to improve space division and the privacy budget allocation.

4 Conclusions

In this paper a CGM model is designed to apply differential privacy to a social network data publication and to ensure the sanitized graph's utility. This procedure can be divided into three steps. First, according to the features of the social network, the original graph is classified into multiple disconnected subgraphs and the noisy count of bridges is recorded. Then for each sub-graph, the dense regions in the corresponding adjacent matrix are explored using a quad-tree in which nodes consist of the region's size and noisy count of 1s. Finally, with the leaf nodes of each quad-tree, the sanitized adjacent matrixes are reconstructed and combined into the whole sanitized graph using/through the noisy count of bridges. At the end of this paper, some experiments and results are analyzed using degree distribution, shortest path, clustering coefficient and betweenness centrality methods. According to the results, the proposed method maintains data utility and has certain advantages. Further research will be conducted to improve the accuracy and efficiency of the method.

References

[1] Guan H, Yang H J, Wang J. An ontology-based approach to security pattern selection[J]. *International Journal of Automation and Computing*, 2016, 13(2): 168-182

[2] Sweeney L. k-anonymity: a model for protecting privacy [J]. *International Journal of Uncertainty, Fuzziness and Knowledge based Systems*, 2002, 10(5):557-570

[3] Li N H, Li T C, Venkatasubramanian S. t-closeness: Privacy beyond k-anonymity and l-diversity[C]. In: Proceedings of the IEEE International Conference on Data Engineering, Istanbul, Turkey, 2007. 106-115

[4] Machanavajjhala A, Gehrke J, Kifer D, et al. l-diversity: privacy beyond k-anonymity[C]. In: Proceedings of the 22nd International Conference on Data Engineering. Atlanta, USA, 2006. 24-35

[5] Dwork C, McSherry F, Nissim K, et al. Calibrating noise to sensitivity in private data analysis[C]. In: Proceedings of the 3rd Conference on Theory of Cryptography. New York, USA, 2006. 265-284

[6] Wang J, Liu S B, Li Y K. A review of differential privacy in individual data release[J]. *International Journal of Distributed Sensor Networks*, 2015, doi:10.1155/2015/259682

[7] Hay M, Li C, Miklau G, et al. Accurate estimation of the degree distribution of private networks[C]. In: Proceedings of the 9th IEEE International Conference on Data Mining, Miami, USA, 2009. 169-178

[8] Wang J L, Guan M, Wei S C. A survey on differential privacy research for social network analysis[J]. *Chinese High Technology Letters*, 2015, 25(3):239-248 (in Chinese)

[9] C Li, G Miklau, M Hay, et al. The matrix mechanism: optimizing linear counting queries under differential privacy[J]. *The VLDB Journal*, 2015, 24(6):757-781

[10] K Nissim, U Stemmer. On the generalization properties of differential privacy[J]. *Computer Science*, 2015, arXiv: 1504.05800

[11] Oneto L, Ridella S, Anguita D. Differential privacy and generalization: Sharper bounds with applications [J]. *Pattern Recognition Letters*, 2017, 89(4): 31-38

[12] Task C, Clifton C. A guide to differential privacy theory in social network analysis[C]. In: Proceedings of the IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, Los Alamitos, USA, 2012. 411-417

[13] Gehrke J, Lui E, Pass R. Towards privacy for social networks: a zero-knowledge based definition of privacy[C]. In: Proceedings of the Theory of Cryptography Conference, Providence, USA, 2011. 432-449

[14] Hay M, Rastogi V, Miklau G, et al. Boosting the accuracy of differentially private histograms through consistency [J]. *Proceedings of the VLDB Endowment*, 2010, 3(1-2):1021-1032

[15] Nissim K, Raskhodnikova S, Smith A. Smooth sensitivity and sampling in private data analysis[C]. In: Proceedings of the 39th Annual ACM Symposium on Theory of Computing, SanDiego, USA, 2007. 75-84

[16] Wang Y, Wu X, Zhu J, et al. On learning cluster coefficient of private networks[J]. *Social Network Analysis and Mining*, 2013, 3(4): 925-938

[17] Chen R, Fung B C M, Philip S Y, et al. Correlated network data publication via differential privacy [J]. *The VLDB Journal*, 2014, 23(4):653-676

[18] Dwork C. A firm foundation for private data analysis[J]. *Communications of the ACM*, 2011, 54(1):86-95

[19] Dwork C. Differential privacy[C]. In: Proceedings of the 33rd International Colloquium on Automata, Languages and Programming, Venice, Italy, 2006. 1-12

[20] Dwork C, Rothblum G N. Concentrated Differential Privacy, CoRR[S], abs/1603.01887, 2016

[21] McSherry F, Talwar K. Mechanism design via differential privacy[C]. In: Proceedings of the 48th Annual IEEE Symposium on Foundations of Computer Science, Providence, USA, 2007. 94-103

[22] McSherry F. Privacy integrated queries: an extensible platform for privacy-preserving data analysis[J]. *Communications of the ACM*, 2010, 53(9):89-97

Wang Junli, born in 1978. She received her Ph. D degrees in Computer Science Department of Tongji University in 2007. She also received her B. S. degree from Jilin University in 2000. Her research interests include privacy protection, deep learning, and network data analysis.