IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received July 19, 2019, accepted July 31, 2019, date of publication August 5, 2019, date of current version August 20, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2933151

Large-Scale Dynamic Social Network Directed Graph K-In&Out-Degree Anonymity Algorithm for Protecting Community Structure

XIAOLIN ZHANG[1], JIAO LIU[1], JIAN LI[1], AND LIXIN LIU[1,2]
[1]School of Information Engineering, Inner Mongolia University of Science and Technology, Baotou 014010, China
[2]School of Information, Renmin University of China, Beijing 100872, China

Corresponding author: Xiaolin Zhang (2784899426@qq.com)

This work was supported in part by the Natural Science Foundation of China under Grant 61562065, and in part by the Research on Large-Scale Social Network Privacy Protection Technology based on Cloud Computing.

ABSTRACT Social network data publishing is dynamic, and attackers can perform association attacks based on social network directed graph data at different times. The existing social network privacy protection technology has low performance in dealing with large-scale dynamic social network directed graph data, and anonymous data publishing does not meet the needs of community structure analysis. A Dynamic Social Network Directed Graph K-In&Out-Degree Anonymity (DSNDG-KIODA) method to protect community structure is proposed. The method is based on the dynamic grouping anonymity rule to anonymize the dynamic K-in&out-degree sequence, and the virtual node distribution is added in parallel to construct an anonymous graph. The node information is transmitted based on the GraphX, and the virtual node pairs are selected and deleted according to the change of the directed graph modularity to reduce information loss. The experimental results show that the DSNDG-KIODA method improves the efficiency of processing large-scale dynamic social network directed graph data, and ensures the availability of community structure analysis when data is released.

INDEX TERMS Directed graph modularity, dynamic grouping anonymous rule, dynamic social network directed graph, GraphX, K-in&out-degree anonymity.

I. INTRODUCTION

With the rapid development of the Internet, online social networks(OSNs) have become more and more mature, and more and more people are involved. According to the 43rd Statistical Report on the Development of Internet in China, as of December 2018, the number of Internet users in China was 829 million, and 56.53 million new Internet users were added throughout the year. The Internet penetration rate reached 59.6%, an increase of 3.8% from the end of 2017. When large-scale users use OSNs, some users with the same hobbies or attributes will form a community. However, as the number of people using the network increases, serious privacy problems of users' information and community structure information may occur. The increase of network users means that the number of nodes in the social network is constantly changing, and the social network is dynamic. Bhagat *et al.* [1] pointed out that the attacker can easily identify the target users of the social network through the node information or community structure information of the social network data at different times, which leads to the leakage of personal information or community information. Therefore, the privacy protection of large-scale dynamic social directed graph has important theoretical significance and practical value. Dynamic social networks often form a community structure that is unique to a complex network. Researchers can analyze the community structure of large-scale dynamic social networks with directed graphs, which has important research significance in similar group discovery and group behavior pattern discovery [2].

For large-scale dynamic social network directed graph, privacy protection is implemented in parallel based on GraphX, which ensures the availability of community structure analysis when data is published anonymously. The main work and contributions are as follows:

(1) Aiming at the directed graph of large-scale dynamic social network, a new dynamic accessibility attack

The associate editor coordinating the review of this manuscript and approving it for publication was Osama Sohaib.

model is proposed. Based on the protection of community structure, a dynamic social network directed graph K-in&out-degree anonymity model is proposed.

(2) A dynamic social network directed graph K-in&out-degree anonymity (DSNDG-KIODA) algorithm is proposed to protect the community structure of large-scale social network directed graph whereas ensuring that anonymous graphs satisfy K-in&out-degree anonymity at different times.

(3) The experiments on the real data set show that the DSNDG-KIODA algorithm improves the processing efficiency of the directed graph privacy protection on large-scale dynamic social networks, and ensures the high availability of community structure analysis in data publishing.

The organization of the rest of this paper is as follows. In Section II, we describe the related work to dynamic social network privacy protection and community structure protection. Section III introduces relevant preliminary knowledge and gives definitions of research questions. Section IV introduces the DSNDG-KIODA algorithm that protects the community structure. Section V is based on real dynamic social network datasets and experimentally tested in terms of algorithm performance, information loss, and data availability. Section VI concludes the full text and gives the future work.

II. RELATED WORK

Social network privacy information mainly includes node privacy, edge privacy, and graph structure privacy [3]. Attackers can attack and mine the privacy information of social network, which leads to the leak of users' personal information. At present, the privacy protection of large-scale social network data is mainly divided into static social network privacy protection method and dynamic social network privacy protection method.

For the privacy protection of static social networks, the concept of k-degree anonymity [4] of undirected graphs is proposed. It is required that any node in the graph has at least k-1 nodes with the same degree, and the greedy strategy is added to achieve anonymity against node degree attribute attacks. Salas and Torra [5] proposed a k-degree anonymity method, which guaranteed that each approximation has at least k elements in the degree sequence. When the sum of the degrees of the anonymity sequence is even, the optimal solution of k-degree anonymity is obtained by using Euclidean distance. Li *et al.* [6] proposed a random sparse and random perturbation methods, and the social network undirected graph is randomly modified using the probability method. Casas-Roma *et al.* [7] proposed the UMGA algorithm, which generated anonymity sequence by greedy algorithm and exhaustive method, and modifies undirected graph by random edge selection and neighbor centrality edge selection to achieve k-degree anonymity. Sun *et al.* [8] proposed a novel anonymizing approach, called splitting anonymization. This approach avoids the low utility caused by the enforced noises on knowledge that is already known to the attackers. Social network processed by splitting anonymization can refuse direct attack. Macwan and Patel [9] proposed an improved k-degree anonymity model that retain the social network structural properties and also to provide privacy to the individuals. Utility measurement approach for community based graph model is used to verify the performance of the proposed technique. Dongran *et al.* [10] proposed a technique of hierarchical k-anonymity for graphlet structural perception. The method considers the degree of social network nodes according to the Characteristics of the power-law distribution. The nodes are divided according to the degrees, and the method analyzes the graphlet structural features of the graph in the privacy process and adjusts the privacy-processing strategies of the edges according to the graphlet structural features. The literature [4]–[10] only addresses the static social network protection of social network undirected graphs, ignoring the direction of social network graph data.

For the privacy protection of dynamic social networks, Bhagat *et al.* [11] proposed that static social networks have limitations. Social networks evolve and a single instance is inadequate for analyzing the evolution of the social network or for performing any longitudinal data analysis. Publishing multiple instances of the same network independently has privacy risks, since stitching the information together may allow an adversary to identify users in the networks. So, it is of great significance to study dynamic social networks. Bhagat *et al.* [1] proposed a link prediction algorithm, which predicts the future structure through the current state of social network. Ding *et al.* [12] proposed a De-anonymity algorithm to anonymize dynamic social networks, combining structural knowledge with node attributes to defend against complex attacks by attackers. Wang *et al.* [13] proposed the DMRA algorithm on privacy preserving for releasing multiple time-series social network graphs based on the Time-Series Class Safety Condition(TSCSC) to achieve the K-degree anonymity. Rossi *et al.* [14] proposed a dynamic k-degree anonymity algorithm, which makes the time-degree sequence of each node indistinguishable from the temporal sequence of at least other k-1 nodes. Kiabod *et al.* [15] introduced a time-saving k-degree anonymization method in social network that anonymizes the social network graph. The anonymized degree sequence of the graph is computed based on the tree and partitions the graph bottom-up nodes based on the anonymization levels to realize the dynamic changes in privacy protection levels. The existing dynamic privacy protection methods aim at dynamic social network undirected graph. Although the information loss of the graph modification is reduced, the network connectivity of the original graph is changed. In the process of anonymity, the community structure of dynamic social network graph is not considered, which affects the analysis of the nature of community structure and reduces the value of data publishing.

The large-scale dynamic social network directed graph has a community structure, so the privacy protection of the community structure of the social network map has become

the focus of researchers. Campan *et al.* [16] used the community detection algorithm based on graph segmentation theory. By calculating the Laplacian matrix, the spectral constraint conditions are set, and the constraints of adding and deleting edges are calculated. Wang *et al.* [17] proposed a novel local-perturbation technique by combining the clustering technique with the randomly reconstructing technique that can reach the privacy requirement of k-anonymity, while minimizing the impact on community structure. Kumar and Kumar [18] used the concept of upper approximation of rough sets to divide communities and anonymously preserve the community structure properties of graphs after anonymity. Rousseau *et al.* [19] proposed a novel edge modification technique that better preserves the communities of a graph while anonymizing it. By maintaining the core number sequence of a graph, its community structure is guaranteed to remain unchanged. Macwan and Patel [20] proposes a fast privacy protection method for large-scale social networks based on heuristic analysis. Firstly, the community structure is divided, and personalized K-Degree anonymity is implemented for small communities in a distributed way. Then the generalized community structure forms a cohesive network, and the nodes are homogeneously processed to form a final anonymous network.

At present, most social network privacy protection technologies have limitations in dealing with large-scale dynamic social network directed graphs. The algorithm only protected the private information of individuals against dynamic social network undirected graphs, and ignored the protection of the community structure of social network directed graphs. Aiming at large-scale dynamic social network directed graph, a dynamic social network directed graph K-in&out-degree anonymity (DSNDG-KIODA) method to protect community structure is proposed. Method improves the efficiency of processing large-scale dynamic social network directed graph, and ensure the availability of community structure analysis when data is published.

III. PRELIMINARY KNOWLEDGE AND PROBLEM DEFINITION

Definition 1 (Dynamic Social Network Directed Graph): The dynamic social network directed graph is also called the social network incremental graph, which is expressed as $G = \{G_0, G_1, G_2, \ldots, G_t\}$ $(t = 0, 1, 2, \ldots, t)$ and the anonymous dynamic social network directed graph is represented as $G^* = \{G_0{}^*, G_1{}^*, G_2{}^*, \ldots, G_t{}^*\}$. Where $G_t = (V_t, E_t)$, V_t and E_t represent the node set and edge set of the graph G_t respectively. The dynamic social network directed graph is an incremental sequence, $V_{t-1} \subseteq V_t, E_{t-1} \subseteq E_t$. The edge $\langle u, v\rangle$ indicates that one edge from the node u points to v. The edge $\langle u, v\rangle$ is called the out-edge of u and the in-edge of v. The number of in-edges of node u is the in-degree of u, which is denoted as $d_{in}(u)$. The number of out-edges of node u is the out-degree of u, denoted as $d_{out}(u)$. The in&out-degree of node u at time t is represented by $d^t(u) = (d_{in}{}^t(u), d_{out}{}^t(u))$.

Definition 2 (Dynamic K-In&Out-Degree Attack): Suppose the attacker knows the in-degree and out-degrees of the target node at time t_1 and time t_2. The attacker can uniquely identify the target node through this background knowledge, which is called K-in&out-degree attack.

As shown in Figure. 1, it is assumed that the attacker knows that the in&out-degree of the target node u at $t = 0$ is (1,3), then Alice and Bob can be identified according to Figure. 1(a), and the target node cannot be uniquely identified. However, the attacker also knows that the in&out-degree of the target node u at $t = 1$ is (2,3) as shown in Figure. 1(b), so that the target node is uniquely identified as Alice, which leads to the disclosure of the privacy information of the Alice.

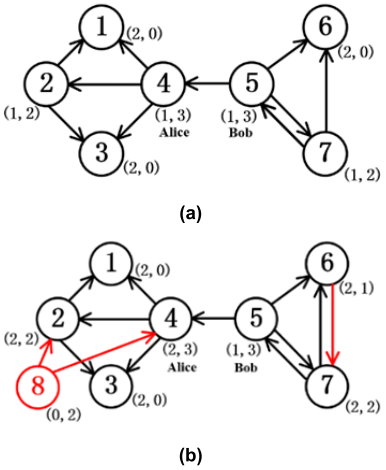

FIGURE 1. Dynamic social network directed graph. (a) Social network directed graph G_0 at t = 0. (b) Social network directed graph G_1 at t = 1. (Where the red arrow indicates the node/edge added at t = 1).

Definition 3 (K-In&Out-Degree Anonymity): Given a dynamic social network directed graph $G_t = (V_t, E_t)$ and the positive integer K. For any node $v \in V(G_t)$ in the directed graph, there are m ($m < k - 1$) other nodes satisfying the same in&out-degree as node v, i.e. $d^t_{in}(v) = d^t_{in}(v_i)$, $d^t_{out}(v) = d^t_{out}(v_i)$ $(1 \le i \le m)$. It is said that the dynamic social network directed graph G_t is a K-in&out-degree anonymity graph.

Definition 4 (Cost of In&Out-Degree Increase): For the nodes $u \in V_{t+1}$ and $u \notin V_t$, the cost of in&out-degree increase of judging the grouping of node u belongs to is expressed by $\Delta d(u)$. The $\Delta d(u)$ is calculated as follows:

$$\Delta d(u) = \left|\text{goal}^t\ (in\_deg) - d_{in}^{t+1}\ (u)\right| + \left|\text{goal}^t\ (out\_deg) - d_{out}^{t+1}\ (u)\right| \quad (1)$$

where goal ^{t} (in_deg, out_deg)) is the goal in&out-degree of g^t_n $(n = 1, 2, \ldots, n)$.

Definition 5 (Dynamic Grouping Anonymous Rule): For the anonymous sequence $d_t^{\sim} = \{g^t_1, g^t_2, \ldots, g^t_n\}$ at time t, the anonymity rule for the in&out-degree sequence at time $t + 1$ is as follows:

(1) For the newly added node, determine the group to which the node belongs by calculating the $\Delta d(u)$;

(2) For the in&out-degree changes of node u in the original graph, first determine whether there is the same in&out-degree as $d^{t+1}(u)$ in the target of the anonymous group at time t. If it exists, determine whether the number of group which node u belongs to is greater than or equal to $k+1$, and if it is satisfied, merge node u into the group. Otherwise, the grouping of the node u is unchanged, and the target in&out-degree of the group at time $t+1$ is calculated.

Definition 6 (Dynamic Social Network Directed Graph K-In&Out-Degree Anonymity Model): Given a dynamic social network directed graph $G = \{G_0, G_1, G_2, \ldots, G_t\}$ and the positive integer K.

(1) K-in&out-degree anonymous sequence is obtained by anonymous in&out-degree sequence of nodes at different times based on dynamic grouping anonymous rule.

(2) Anonymous graphs are constructed in parallel according to K-in&out-degree anonymous sequence distribution, and information between nodes is transmitted based on GraphX. Virtual nodes are selected for merging and deleting according to the change of modularity of directed graphs for many iterations, improving the availability of community structure analysis in data publishing.

The anonymous dynamic social network directed graph $G^* = \{G_0^*, G_1^*, G_2^*, \ldots, G_t^*\}$ obtained by satisfying these two conditions conforms to the dynamic social network directed graph K-in&out-degree anonymity model.

The anonymity process of the dynamic social network directed graph is shown in Figure. 2, where $K = 3$. The original social network directed graph G_t does not satisfy K-in&out-degree anonymity at time t. Adding edge $\langle 10, 11\rangle$ to get the anonymous social network directed graph G_t^*. Node 12 is added and edges $\langle 5, 7\rangle$, $\langle 9, 12\rangle$, $\langle 12, 10\rangle$, $\langle 12, 11\rangle$ are added to obtain the original social network directed graph G_{t+1} at time $t+1$. The directed graph G_t is anonymously obtained to obtain a 3-degree anonymity social network directed graph G_{t+1}^*, which satisfies the K-in&out-degree anonymity.

IV. DSNDG-KIODA ALGORITHM

Dynamic social network directed graph K-in&out-degree anonymity (DSNDG-KIODA) algorithm for protecting community structure is combined with Spark, a computational engine for large-scale data processing. The algorithm is executed in the distributed parallel environment, and the privacy protection strategy is implemented in parallel for the large-scale dynamic social network directed graph data. The algorithm symbols are described in Table I.

A. INITIAL SEQUENCE PARTITION ALGORITHM

Initial Sequence Partition(G,k) algorithm is used to group and anonymize the original social network directed graph K-in&out-degree sequence at the initial moment. The in-degree(out-degree) of the goal degree in the same group is the maximum value of all in-degree(out-degree) in the group, that is, goal(in_deg,out_deg)=(max{the in-degree of all elements in the group, max{the out-degree of all elements in the group}).

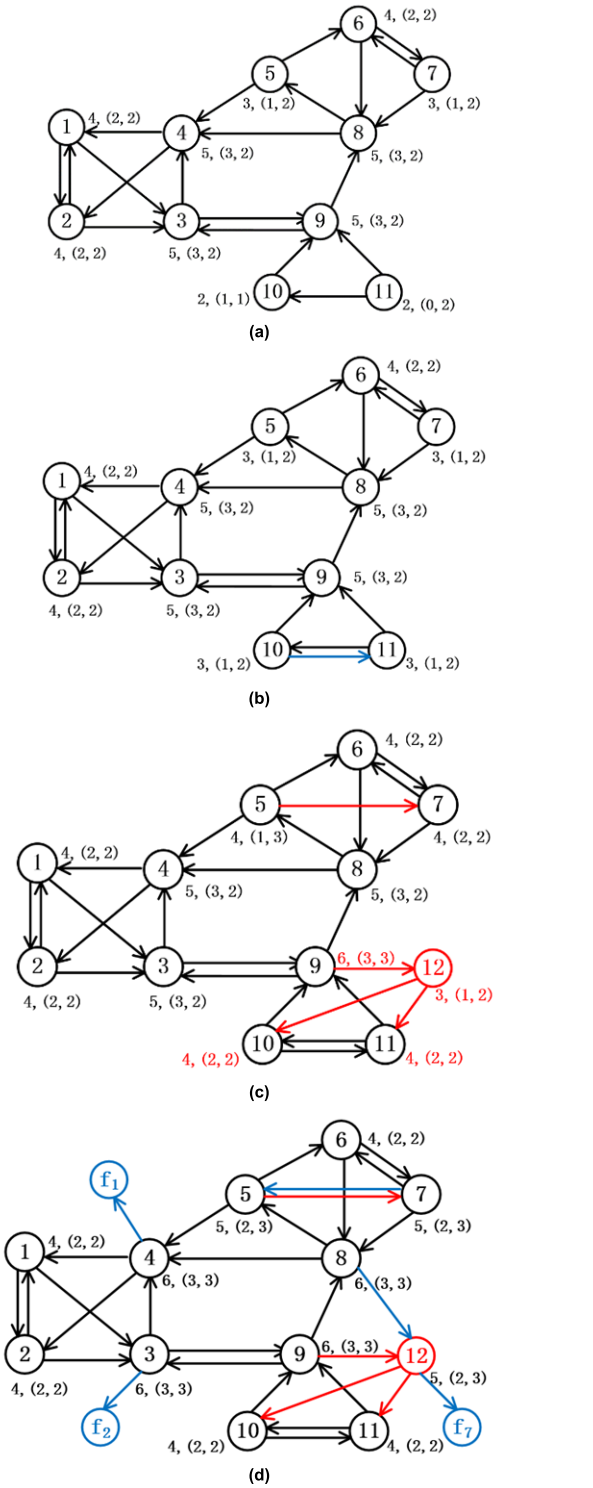

FIGURE 2. Anonymous process of dynamic social network directed graph (K = 3). (a) Original directed graph G_t at time t. (b) Anonymous directed graph G_t^* at time t. (c) Original directed graph G_{t+1} at time $t+1$. (d) Anonymous directed graph G_{t+1}^* at time $t+1$. (Where the red arrow indicates the node(edge) added at time $t+1$, the blue arrow indicates the node/edgeS added for anonymization at time $t+1$).

TABLE 1. The algorithm symbols description.

Symbols	Description
k	Anonymous parameter.
$G_0=\{V_0,E_0\}$	Directed graph $G_0=\{V_0,E_0\}$ at time t=0, which is composed of the node set V_0 and the edge set E_0.
$G_t=\{V_t,E_t\}$	Directed graph $G_t=\{V_t,E_t\}$ at time t, which is composed of the node set V_t and the edge set E_t.
$d_0^{\sim}=\{g_1^0, g_2^0, \ldots, g_n^0\}$	K-in&out-degree anonymity sequence at time t=0.
$d_t^{\sim}=\{g_1^t, g_2^t, \ldots, g_n^t\}$	K-in&out-degree anonymity sequence at time t.
CandidateSet	Candidate virtual node set.
(f_w, f_x)	Virtual node pair.
$G=\{G_0,G_1,G_2,\ldots,G_t\}$	Dynamic social network directed graph.
$G^*=\{G_0^*,G_1^*,G_2^*,\ldots,G_t^*\}$	Dynamic social network anonymous directed graph.

The Initial Sequence Partition algorithm is as follows:

Algorithm 1 FInitial Sequence Partition(G,k)

Input: $G_0 = \{V_0, E_0\}$, k
Output: $d_0^{\sim} = \{g_1^0, g_2^0, \ldots, g_n^0\}$
1 MF_Seq=[];
2 **for** each node ⟨in_deg,out_deg⟩ **do**
3 Insert in MF_Seq;
4 **end for**
5 Sort(MF_Seq) by ⟨in_deg,out_deg⟩;
6 last_partition_index=0;
7 **for** $v_i \in$MF_Seq **do**
8 **for** l=last to i-1 **do**
9 goal_1(in_deg,out_deg)= max(in_deg[i],out_deg[i]);
10 **end for**
11 **for** m=i to i+k **do**
12 goal_2(in_deg, out_deg)= max(in_deg[m+1],out_deg[m+1]);
13 goal_3(in_deg, out_deg)= max(in_deg[m],out_deg[l]);
14 **end for**
15 SPC1=goal_1-MF_Seq[i], SPC2=0;
16 **for** j=i+1 to i+k **do**
17 SPC1=SPC1+goal_3–deg[j-1];
18 SPC2=SPC2+goal_2–deg[j];
19 **end for**
20 **if** SPC2<SPC1 **then**
21 last_partition_index=i;
22 i=i+k;
23 **else**
24 i++;
25 **end if**
26 **end for**

Initial Sequence Partition(G,k) algorithm shows the steps for the initial sequence partition. For the given directed graph $G_0 = \{V_0, E_0\}$, in each iteration, it calculates two partitions cost and makes a decision of optimal partitioning. For each node, it checks for further k nodes to decide whether to merge current nodes with current group or to initiate a new group. Hence, the time complexity of Algorithm 1 is O(k|V|).

Lines 2-14 of the Algorithm 1 are sorted according to the in&out-degree of node. Lines 16-25 determine the size of the SPC1 and SPC2 values for grouping. SPC1 represents the anonymization cost of the current element merged into the previous group, and SPC2 represents the cost of forming a new group with the following k-1 element. Therefore, if SPC2<SPC1, then the element forms a new group.

Assuming K = 3, the grouping result of the original social network directed graph G_t at time t as shown in Figure. 2(a) are g_1^0 = {(3, 2), (3, 2), (3, 2), (3, 2)}, g_2^0 = {(2, 2), (2, 2), (2, 2)}, g_3^0 = {(1, 2), (1, 2), (1, 1), (0, 2)}. Thus, the K-in&out-degree anonymous sequence is $d_0^{\sim}$ = {(3, 2), (3, 2), (3, 2), (3, 2), (2, 2), (2, 2), (2, 2), (1, 2), (1, 2), (1, 2), (1, 2)}.

B. DYNAMIC SEQUENCE PARTITION ALGORITHM

The dynamic sequence partition groups the in&out-degree sequence at time t + 1 according to the dynamic grouping anonymous rule to obtain K-in&out-degree anonymous sequence. The Dynamic Sequence Partition algorithm is as follows:

Algorithm 2 FDynamic Sequence Partition ($d_t^{\sim}$, G_{t+1}, k)

Input: $d_t^{\sim} = \{g_1^t, g_2^t, \ldots, g_n^t\}$, k
Output: $d_{t+1}^{\sim} = \{g_1^{t+1}, g_2^{t+1}, \ldots, g_n^{t+1}\}$
1 ChangeNodeSet = ∅, AddNodeSet = V_{t+1}-V_t, Δd=0;
2 **for** u∈AddNodeSet **do**
3 Calculate the value of Δd(u) and select a group with a small Δd(u) value;
// Determining the group to which the node belongs by calculating the Δd(u)
4 **end for**
5 **for** v∈V_t **do**
6 **if** $d^t(v) \neq d^{t+1}(v)$ **then**
7 add node v to ChangeNodeSet;
8 **end if**
9 **end for**
10 **for** w ∈ ChangeNodeSet **do**
11 **for** i=1 to n **do**
12 **if** $d^{t+1}(w)$=goal(g_i^t) **then**
13 **if** w.group.size≥k+1 **then**
14 $g_i^{t+1} = g_i^t \cup w$;
15 **else**
16 $g_i^{t+1} = g_i^t$;
17 **end if**
18 **else**
19 $w.group^{t+1} = w.group^t$;
20 **end if**
21 calculate the goal of g_i^{t+1};
22 return g_i^{t+1};
23 **end for**

Dynamic Sequence Partition($d_t^{\sim}$, G_{t+1}, k) algorithm shows the steps of group anonymity of a dynamic k-in&out-degree sequence. For the directed graph G_t anonymous k-in&out-

degree sequence at time t + 1, according to the grouping result at time t, first determine the group to which the newly added node belongs (the number of newly added nodes is represented by m), then determine the change of the grouping of the changing node, and finally calculate the target access degree of each group. Therefore, the time complexity of Algorithm 2 is O(mn+|V|).

As shown in Figure. 2(c) at time t+1, AddNodeSet={12}, ChangeNodeSet={5, 7, 9, 10, 11}. First, determine the group to which the node 12 belongs, at time t goal(g_1^t) = (3, 2), goal(g_2^t) = (2, 2), goal(g_3^t) = (1, 2), $d^{t+1}(12)$ = (1, 2) = goal(g_3^t), so node 12 is divided into g_3. The in&out-degree of nodes 5, 7, 9, 10, 11 are all changed. According to lines 10-23 of algorithm 3, the sequence of anonymous K-in&out-degree at time t + 1 is $\tilde{d}_{t+1} = \{g_1^{t+1} = \{(3, 3), (3, 3), (3, 3), (3, 3)\}$, $g_2^{t+1} = \{(2, 2), (2, 2), (2, 2), (2, 2), (2, 2)\}$, $g_3^{t+1} = \{(2, 3), (2, 3), (2, 3)\}\}$.

C. MERGE_DELETE ALGORITHM

The anonymous graph is constructed by adding virtual nodes in parallel according to the dynamic K-in&out-degree anonymous sequence distribution. In order to reduce information loss, the virtual node pairs are merged and deleted through node information transfer based on the GraphX, which improves the availability of community structure analysis in data publishing.

The data structure of the information transfer is represented by a five-tuple (dstid, srcid, hops, community, tags), which is called a n-hops neighborhood table (HNT). As shown in TABLE II, each row of the HNT table is an HNTE (n-hops neighborhood table entry).

(1) stid: the node ID.
(2) srcid: initially, dstid is the node ID, and dstid is the node ID of the destination node during the information transfer.
(3) hops: the number of iterations that the source node transfers information to the destination node.
(4) community: the community to which the source node belongs.
(5) tags: initially tags=0, tags=1 means that the source node and the destination node are both virtual nodes.

TABLE 2. n-hops neighborhood table(HNT).

Node	dstid	scrid	hops	community	tags

Initially, the dstid and srcid of the node are the node ID, hops=0, tags=0. As shown in Figure. 2(c), node 2 has HNTE={2, 2, 0, 1, 0}.

Definition 7 (Virtual Node Pair Merge_Delete Condition): Exists edge ⟨u, f_w⟩ and edge ⟨f_x, v⟩, virtual node pair (f_w, f_x) can be merged and deleted, if and only if ∀(f_w, f_x) ∈ VirtualSet meets the following three conditions:

(1) f_w, f_x ∉ VirtualRDD.
(2) ⟨u, v⟩ ∉ EdgeRDD.
(3) u≠v.

Modularity also known as a modular metrics, is a commonly used method to measure the strength of network community structure, which was first proposed by NewMan [21]. The modularity is calculated as follows:

$$Q = \frac{1}{2m}\sum_{ij}\left[A_{ij} - \frac{k_i * k_j}{2m}\right]\delta(C_i, C_i) \tag{2}$$

where the matrix A_{ij} is an adjacency matrix, and A_{ij} represents the weight of the edge between the node i and the node j. When the network is an unweighted graph, the weight of all edges can be regarded as 1. $k_i(k_j)$ is the sum of the weights(degrees) of all the edges connected to node i. The $\delta(C_i,C_j)$ function means that if node i and node j in the same community return 1, otherwise return 0.

However, for the social networks directed graph, nodes have different in-degrees and out-degrees. Therefore, for the social network directed graph, the definition of the directed graph modularity is proposed.

Definition 8 (Directed Graph Modularity): For the social network directed graph, the directed graph modularity is represented by DQ. The DQ is calculated as follows:

$$DQ = \frac{1}{m}\sum_{ij}\left[A_{ij} - \frac{k_i^{out} * k_j^{in}}{m}\right]\delta(C_i, C_j) \tag{3}$$

where the matrix A_{ij} is an adjacency matrix of the directed graph. If there is an edge ⟨i, j⟩, there is an edge pointing from node i to node j, then element $a_{ij} = 1$, otherwise $a_{ij} = 0$. k_i^{out} and k_j^{in} represent the out-degree of node i and the in-degree of node j, respectively, and m is the total number of edges of the directed graph. The directed graph modularity can be simplified as:

$$\begin{aligned} \mathrm{DQ} &= \frac{1}{m}\sum_{ij}\left[A_{ij} - \frac{k_i^{out} * k_j^{in}}{m}\right]\delta\left(C_i, C_j\right) \\ &= \frac{1}{m}[\sum_{ij}A_{ij} - \frac{\sum_i k_i^{out} * \sum_j k_j^{in}}{m}]\delta(C_i, C_j) \\ &= \frac{1}{m}[\sum_c in - \frac{d_{out} * d_{in}}{m}] \\ &= \sum_c [\frac{l_c}{m} - \frac{d_{out} * d_{in}}{m^2}] \end{aligned} \tag{4}$$

lc is the number of edges in the community, m is the total number of edges of the directed graph, and $d_{out}(d_{in})$ is the sum of out-degrees(in-degrees) in the community.

Definition 9 (Change of Directed Graph Modularity): The change value of the directed graph modularity is expressed by ΔDQ, which measures the information loss caused by adding edge operation. The ΔDQ is calculated as follows:

$$\Delta DQ = \left|DQ\left(G_t\right) - DQ\left(G_t^{'}\right)\right| \tag{5}$$

$DQ(G_t)$ represents the directed graph modularity before adding edge, and $DQ(G'_t)$ represents the directed graph modularity after adding edge.

Definition 10 (Virtual Node Pair Merge_Delete Rule): Each iteration obtains virtual node pair set (VirtualSet) by transferring information between nodes, and the virtual node pair satisfying the VNMDC is placed in the candidate virtual node set (CandidateSet). The virtual node pair merge delete rule is as follows:

(1) The number of virtual node pairs is 1, directly merged and deleted;

(2) The number of virtual node pairs is more than 1:

   a. If the virtual node pair belongs to the same community, randomly select one of the virtual node pairs to merge and delete;

   b. If the virtual node pair belongs to different communities, calculate the value of DQ and select the virtual node pair with the small ΔDQ to merge and delete.

The Merge_Delete algorithm for virtual node pair as follows:

Algorithm 3 FMerge_Delete(CandidateSet)

Input: CandidateSet
Output: (f_w, f_x)
1 M = the number of same community in CandidateSet;
2 N = CandidateSet.size;
3 **if** (N >1) **then**
4 **if** (M==0) **then**
5 (f_w, f_x) =the min(ΔDQ) from CandidateSet;
6 return (f_w, f_x);
7 **end if**
8 **if** (M >1) **then**
9 randomly select (f_w, f_x);
10 return (f_w, f_x);
11 **end if**
12 **else**
13 return (f_w, f_x);
14 **end if**

Merge_Delete(CandidateSet) algorithm shows the steps to select the virtual node pair. For each virtual node pair in the CandidateSet, the virtual node pair that satisfies the condition is selected according to the VNMDC. Therefore, the time complexity of Algorithm 3 is $O(N\times(N-1))=O(N^2)$.

If the number of virtual node pairs in the CandidateSet is more than 1, the algorithm 3 is executed in lines 4-11. For virtual node pairs in different communities, a virtual node pair with a small ΔDQ value is selected for merge and delete. For virtual node pairs in the same community, randomly select one of the virtual node pairs to merge and delete.

D. DSNDG-KIODA ALGORITHM

The dynamic social network directed graph K-in&out-degree anonymity (DSNDG-KIODA) algorithm is as follows:

Algorithm 4 FDSNDG-KIODA

Input: $G = \{G_0, G_1, G_2, \ldots, G_t\}$, k
Output: $G^* = \{G_0^*, G_1^*, G_2^*, \ldots, G_t^*\}$
1 Initialize the dynamic social network directed graph G_0, $d_0^{\sim}$ =**Sequence Partition (G_0,k);**
2 **for** i=1 to t **do**
3 $d_t^{\sim}$ =**Dynamic Sequence Partition** ($d_t^{\sim}$,G_{t+1},**k);**
4 Add virtual nodes based on anonymous sequence to get the anonymous graph G_i';
5 Initialize the anonymous graph G_i', CandidateSet=Ø, VirtualRDD=Ø;
6 **for** SuperStep=1 to 6 **do**
7 Dst.Message ← Src.Message;
  //The source node sends the updated HNTE information to the destination node.
8 **for** (Message from Dst.HNTE) **do**
9 **if** (Message.Tags==1) **then**
10 Dst.VirtualSet ← Message;
11 **end if**
12 **end for**
13 **for** (Message from Dst.VirtualSet) **do**
14 f_w =Message.srcid;
15 f_x =Message.disid;
16 **if** (f_w, f_x) satify VNMDC **then**
17 CandidateSet ← (f_w, f_x);
18 **end if**
19 **end for**
20 **if** CandidateSet.size > 0 **then**
21 (f_w, f_x) =**Merge_Delete(CandidateSet)**;
22 G_i'.EdgeRDD.Remove $\langle u, f_w\rangle$;
23 G_i'.EdgeRDD.Remove $\langle f_x, v\rangle$;
24 G_i'.EdgeRDD.Add $\langle u, v\rangle$;
25 VirtualRDD.Add (f_w,f_x);
26 VoteToHalt (f_w, f_x);
27 **end if**
28 **end for**
29 **return** $\mathbf{d_{ti}^{\sim}}$**,** $\mathbf{G_i}$**';**
30 **end for**

DSNDG-KIODA algorithm shows the step of the dynamic social network directed graph K-in&out-degree anonymity. The time complexity analysis of Algorithm 4 is $O(k|V|+(mn+|V|)+N^2+L)$(where L is the number of virtual node pairs that are selected for merging and deleting).

The dynamic social network directed graph $G=\{G_t, G_{t+1}\}$ as shown in Figure. 2, the specific steps of the DSNDG-KIODA algorithm are as follows:

1. Perform algorithm 1 to obtain the anonymous K-in& out-degree sequence $d_t^{\sim}$ of the initial directed graph at time t, and perform algorithm 2 to obtain the dynamic anonymous K-in&out-degree sequence $d_{t+1}^{\sim}$ at time t+1. The anonymous graph $G_{t+1}^{\sim}$ is constructed by adding virtual node in parallel, as shown in Figure. 3.
2. The anonymous graph is constructed in parallel according to the dynamic K-in&out-degree anonymous

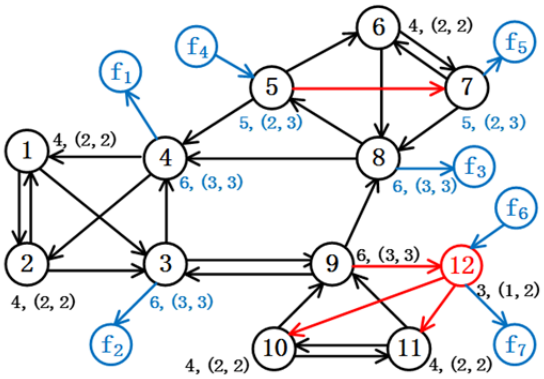

FIGURE 3. Anonymous directed graph $G^{\sim}_{t+1}$ at time t + 1.

sequence, and the virtual node pair is merged and deleted based on the GraphX. The iteration process at time t + 1 is as follows (nodes 1-4 belong to community 1, nodes 5-8 belong to community 2, and nodes 9-12 belong to community 3):

(1) Superstep=0, the node initialization gets the initial EdgeRDD.
(2) Superstep=1, the node receives its own 1-hop neighborhood information and generates a 1-hop neighborhood table. The first iteration tags are all 0, VirtualSet= Ø, CandidateSet=Ø.
(3) Superstep=2, the 2-hop neighbor table is shown in TABLE III (only lists some virtual nodes HNTE), and check if there is tags=1. Iteratively obtains VirtualSet={(f_6,f_7)}. Virtual nodes f_2 and f_1 are connected to node 12 which unsatisfy VNMDC. The operation of merge and delete virtual nodes cannot be performed, CandidateSet=Ø.

TABLE 3. 2-hop neighborhood table.

Node	dstid	scrid	hops	community	tags
f_1	f_1	3	2	1	0
	f_1	5	2	2	0
	f_1	8	2	2	0
f_2	f_2	1	2	1	0
	f_2	2	2	1	0
	f_2	9	2	3	0
f_3	f_3	6	2	2	0
	f_3	7	2	2	0
	f_3	9	2	3	0
f_5	f_5	5	2	2	0
	f_5	7	2	2	0
f_7	**f_7**	**f_6**	**2**	**3**	**1**
	f_7	9	2	3	0

(4) Superstep=3, VirtualSet={(f_1,f_4), (f_5,f_4)}. The virtual nodes f_5 and f_4 belong to the same community, directly merged and deleted, that is, adding edges ⟨6,5⟩ and deleting edges ⟨f_4,5⟩, ⟨6,f_5⟩.
(5) Superstep=4, VirtualSet= Ø.
(6) Superstep=5, VirtualSet={(f_2,f_6), (f_3,f_6)}. The virtual nodes f_2, f_3, and f_6 belong to different communities, calculate the DQ value, and choose the virtual node pair with a small ΔDQ to be merged and deleted. It is calculated that $DQ(G^{\sim}_{t+1})$ =0.5060, DQ((f_2,f_6)) = 0.4735, ΔDQ((f_2,f_6)) = 0.0325, DQ((f_3,f_6)) = 0.4745, ΔDQ ((f_3,f_6)) = 0.0315. Therefore, choose the virtual node pairs (f_3, f_6) to merge and delete, add edges ⟨8, 12⟩ and delete edges ⟨f_6, 12⟩, ⟨8, f_3⟩. The third iteration stops and the result is shown in Figure. 4.

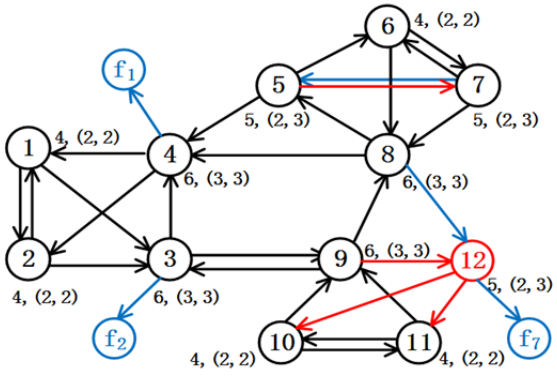

FIGURE 4. The third iteration result of directed graph $G^{\sim}_{t+1}$.

The DSNDG-KIODA algorithm iterates six times and the virtual node pair stops merging and deleting. The anonymous social network directed graph $G_{t+1}*$ as shown in Figure. 2(d).

V. EXPERIMENTAL ANALYSIS

The DSNDG-KIODA algorithm is compared with the dynamic K-degree anonymity algorithm proposed by Rossi [14] and the dynamic k^w-structure diversity anonymity (k^w-SDA) algorithm proposed by Tai [22]. Distributed environment: GraphX, 15 computing nodes, CPU 1.8GHz, 16GB RAM, Hadoop 2.7.2, Spark 2.2.0, Scala 2.11.12.

A. EXPERIMENTAL SETUP

The experiment is tested using five real social network directed graph datasets published by Stanford University: (1) Email-Eu-core; (2) CollegeMsg; (3) Wiki-talk; (4) Stack Overflow; (5) CAIDA AS.

Email-Eu-core dataset consists of email data from a large European research institution, and the directed edge (u,v,t) represents that the user u sent an email to the user v at time t. CollegeMsg dataset consists of private messages sent on the online social network of UCI, the directed edge (u,v,t) represents the user u send a private message to the user v at time t. Wiki-talk dataset is where Wikipedia users edit each other's conversation pages, the directed edge (u,v,t) represents that user u has edited user v's conversation page at time t. Stack Overflow dataset is a time interactive network on the Stack Exchange website Stack Overflow. The directed edge (u,v,t) represents that the user u answered the user v at time t. The statistics related to the dataset as shown in Table IV.

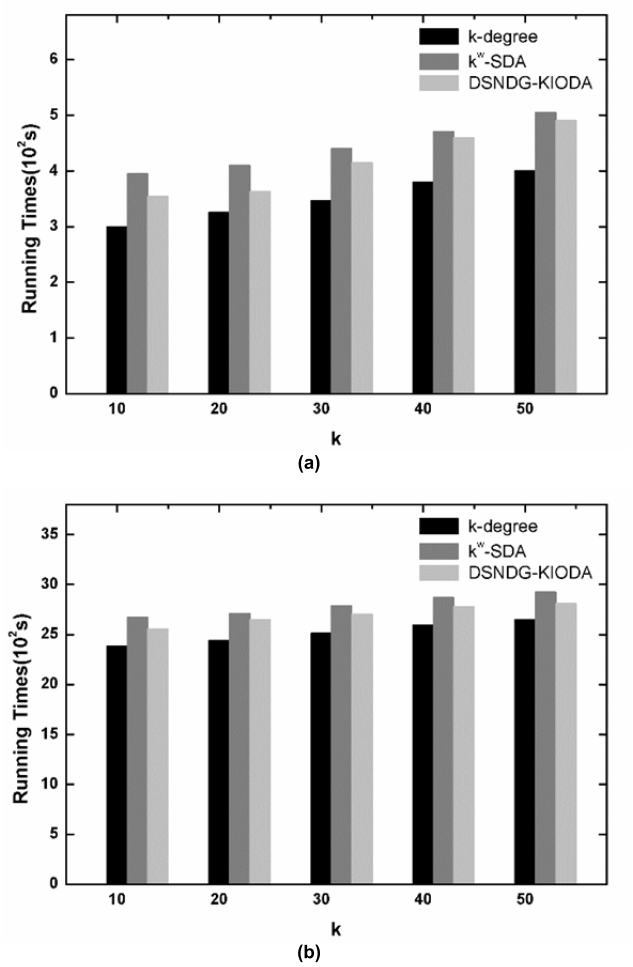

FIGURE 5. Running time. (a) CollegeMsg. (b) Wiki-talk.

TABLE 4. The statistics related to the dataset.

Dataset	Nodes	Temporal Edges	Time span
Email-Eu-core	986	332,334	803 days
CollegeMsg	1,899	59,835	193 days
CAIDA AS	26,475	106,762	122 graphs
Wiki-talk	1,140,149	7,833,140	2,320 days
Stack Overflow	2,464,606	17,823,525	2,774 days

CAIDA AS dataset contains 122 CAIDA AS graphs from January 2004 to November 2007. The five timestamps are selected as shown in Table V.

TABLE 5. CAIDA AS dataset.

t	Date	Nodes	Edges	Nodes Added	Edges Added
1	2005.12.05	20,889	83,640	20,889	83,640
2	2006.05.29	22,191	90,172	1,302	6,532
3	2006.12.25	23,918	98,178	1,727	8,006
4	2007.05.28	25,158	102,468	1,240	4,290
5	2007.11.12	26,475	106,762	1,317	4,294

B. ALGORITHM PERFORMANCE ANALYSIS

Figure. 5 shows the running time on CollegeMsg and Wiki-talk dataset. It can be seen from Figure. 5 that the running time increases with the value of k increases. The running time of k^w-SDA and DSNDG-KIODA algorithm is similar, however, the k-degree algorithm has the shortest running time. This is because k^w-SDA and DSNDG-KIODA algorithm consider the community structure of the social network directed graph in dynamic anonymity. The DSNDG-KIODA algorithm needs to merge and delete more virtual node pairs as the k value increases. Therefore, the running time is slightly larger than the k-degree algorithm, but overall the DSNDG-KIODA algorithm does not run very long.

C. INFORMATION LOSS ANALYSIS

In order to measure the information loss of the graph structure in the anonymity process, the change rate of the graph structure properties on Email-Eu-core and Stack Overflow dataset is tested, i.e. average clustering coefficient (ACC) and eigenvector centrality (EC). G represents the value of ACC(EC) before anonymity, G^* represents the value of ACC(EC) after anonymity.

$$Change\ ratio = |G^* - G|/G \tag{6}$$

Figure. 6 shows the change rate of ACC after anonymity. Figure. 7 shows the change rate of EC after anonymity. The

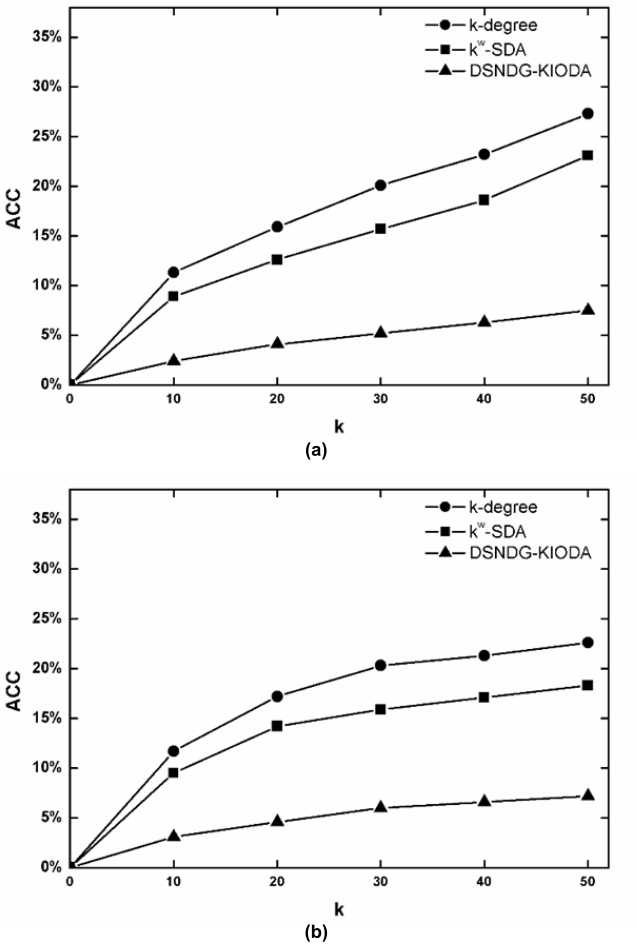

FIGURE 6. Change rate of ACC. (a) Email-Eu-core. (b) Stack Overflow.

greater the rate of change of ACC and EC, the greater the impact of the algorithm on the nature of the graph structure. It can be seen that k^w-SDA and k-degree algorithm have higher change rate than the DSNDG-KIODA algorithm. It shows that k^w-SDA and the k-degree algorithm have a large impact on the graph structure after anonymity. With the increase of k, the change rate of ACC is less than 10% and EC is less than 15% in DSNDG-KIODA algorithm. Therefore, the structural properties of the graph are better guaranteed.

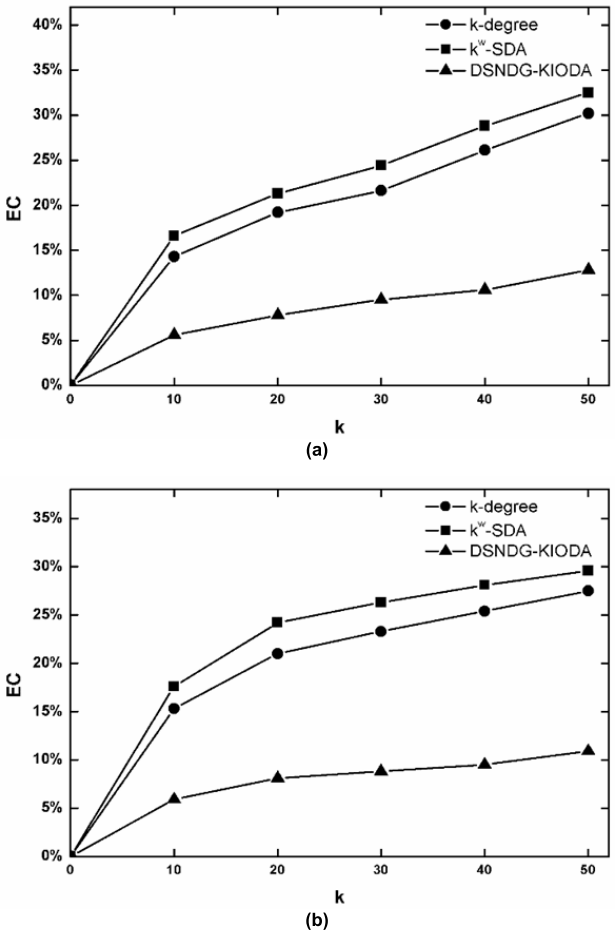

FIGURE 7. Change rate of EC. (a) Email-Eu-core. (b) Stack Overflow.

Figure. 8 shows the change rate of the graph structure (k=10) as a function of timestamp t on CAIDA AS dataset. It can be seen that the number of nodes and edges increases with the increase of timestamp t, and the change rate of graph structure properties of DSNDG-KIODA algorithm are all very small. The DSNDG-KIODA algorithm can guarantee the graph structure properties of dynamic social network directed graph better than k^w-SDA and k-degree algorithm.

D. DATA AVAILABILITY ANALYSIS

The second small eigenvalue (μ_2) of the Laplacian matrix(L) is an important eigenvalue of L. It indicates that how the community is separated, where μ_i ($0 = \mu_1 \leq \mu_2 \leq ... \leq \mu_m \leq m$)

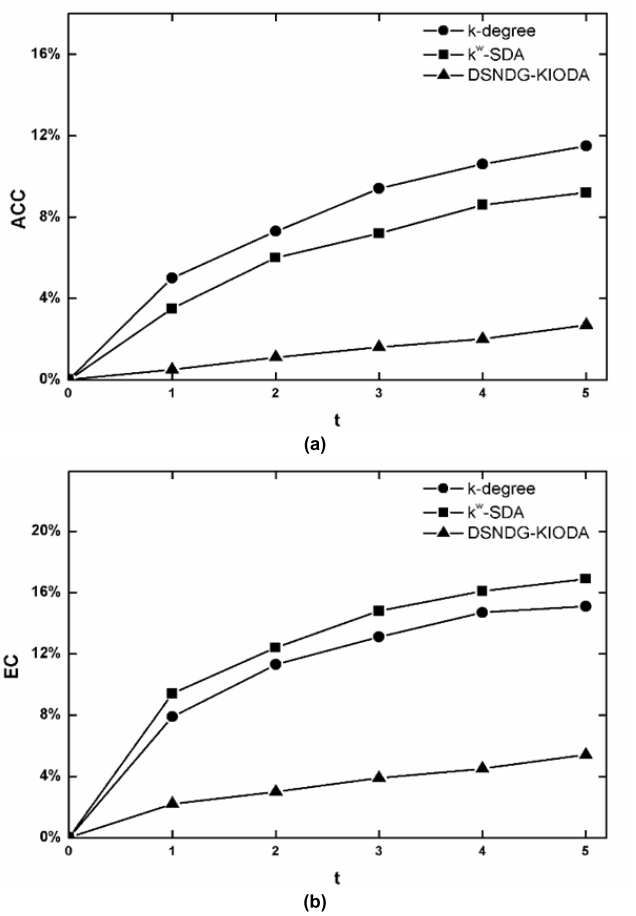

FIGURE 8. Change rate of the graph structure(k=10) on CAIDA AS dataset. (a) ACC. (b) EC.

is the eigenvalue of L. Figure. 9 shows the comparison of the similarity of the μ_2 of the different algorithms on CollegeMsg and Wiki-talk dataset as the k increases. It can be seen that the μ_2 of DSNDG-KIODA algorithm is more similar to the original graph. This is because DSNDG-KIODA algorithm considers the community structure of the original graph when merging and deleting virtual nodes. However, k-degree algorithm ignores the protection of the community structure which causes great loss to the community structure.

Figure. 10 shows the change rate of the community which the node belongs to after anonymity. It can be seen from Figure. 10, the k-degree algorithm does not consider the community of the node in the process of anonymity, so the change rate of the community is large. The DSNDG-KIODA and k^w-SDA algorithm consider the community structure when anonymizing. The community change rate of DSNDG-KIODA algorithm is less than 10% after anonymity, which better maintains the data availability of anonymous graphs in community detection.

Normalized Mutual Information (NMI) [23] is an important index to measure the similarity between two clustering results and community discovery. It can objectively evaluate the accuracy of a community detection compared with standard partitioning. The range of NMI is [0,1], and the higher

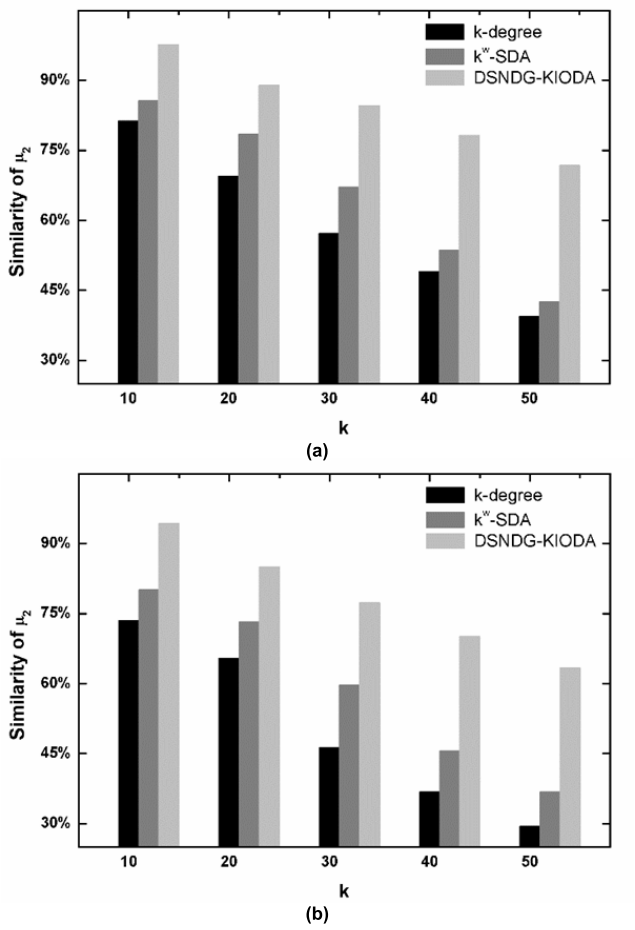

FIGURE 9. Similarity of μ_2. (a) CollegeMsg. (b) Wiki-talk.

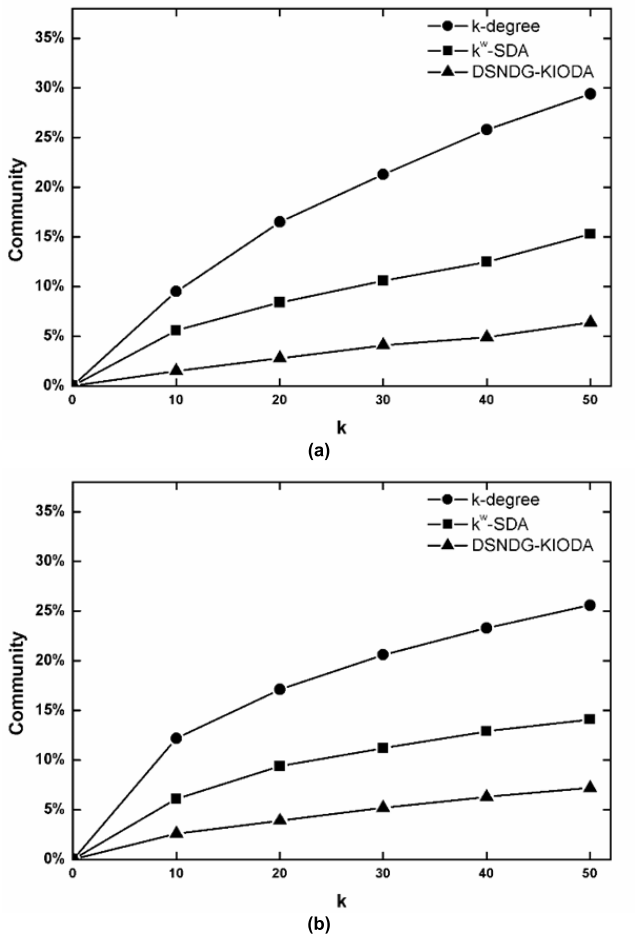

FIGURE 10. Change rate of community. (a) CollegeMsg. (b) Wiki-talk.

the value is, the more accurate the division is. The NMI is calculated as follows:

$$U(X, Y) = \frac{2I(X, Y)}{H(X) + H(Y)} \tag{7}$$

where, I (X, Y) = H (X) −H (X|Y)

$$= \sum_{y \in Y} \sum_{x \in X} p(x, y) \log \left(\frac{p(x, y)}{p(x)p(y)} \right) \tag{8}$$

p(x,y) represents the joint distribution probability of X, Y.

Figure. 11 shows the value of NMI of different algorithms as the value of k increases. It can be seen from Figure. 11, the NMI value of the DSNDG-KIODA algorithm is closer to 1. It shows that the DSNDG-KIODA algorithm has the least impact on the community structure, which better guarantees the availability of community structure analysis when data is released.

VI. CONCLUSION AND FUTURE WORK

Aiming at the large-scale dynamic social network directed graph, a new dynamic in&out-degree attack model is defined according to different times, and DSNDG-KIODA algorithm for protecting community structure is proposed. The algorithm groups and anonymous K-in&out-degree sequence according to dynamic grouping anonymous rule, and the nodes are merged and deleted in parallel to reduce information loss. In the process of merging and deleting virtual nodes, the change value of the directed graph modularity is guaranteed to be the smallest. Experiments based on real social network data show that the DSNDG-KIODA algorithm implements privacy protection for large-scale dynamic social network data. The DSNDG-KIODA algorithm is implemented in a distributed parallel environment. Compared with the traditional K-degree anonymous algorithm, the processing efficiency of the directed graph data of large-scale dynamic social networks is improved, and the availability of community structure analysis of data release is better ensured. DSNDG-KIODA algorithm has good effects in terms of average clustering coefficient (ACC), eigenvector centrality (EC), the second small eigenvalue (μ_2) of the Laplacian matrix, community structure protection, and normalized mutual information (NMI).

The DSNDG-KIODA algorithm protects the privacy information of nodes based on the protection of the availability of community structure analysis. However, the current algo-

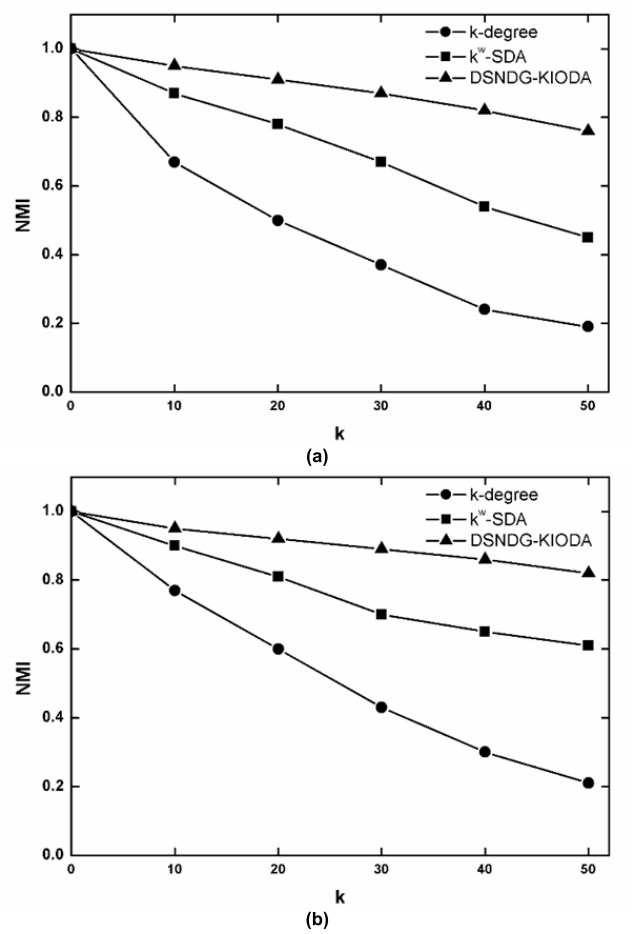

FIGURE 11. NMI value. (a) CollegeMsg. (b) Wiki-talk.

rithm is aimed at users with the same level of privacy protection, and future work will continue to develop for different levels of privacy protection and continuously optimize the algorithm.

REFERENCES

[1] S. Bhagat, G. Cormode, B. Krishnamurthy, and D. Srivastava, "Prediction promotes privacy in dynamic social networks," in *Proc. 3rd Conf. Online Social Netw.*. 2010, p. 6.

[2] M. Kitsak, L. K. Gallos, S. Havlin, F. Liljeros, L. Muchnik, H. E. Stanley, and H. A. Makse, "Identification of influential spreaders in complex networks," *Nature Phys.*, vol. 6, no. 11, pp. 888–893, Aug. 2010.

[3] X. Y. Liu, B. Wang, and X. C. Yang, "Survey on privacy preserving techniques for publishing social network data," *J. Softw.*, vol. 25, no. 3, pp. 576–590, 2014.

[4] K. Liu and E. Terzi, "Towards identity anonymization on graphs," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, Jun. 2008, pp. 93–106.

[5] J. Salas and V. Torra, "Graphic sequences, distances and k-degree anonymity," *Discrete Appl. Math.*, vol. 188, no. 1, pp. 25–31, Jun. 2015.

[6] Y. Li, Y. Li, Q. Yan, and R. H. Deng, "Privacy leakage analysis in online social networks," *Comput. Secur.*, vol. 49, pp. 239–254, Mar. 2015.

[7] J. Casas-Roma, J. Herrera-Joancomartí, and V. Torra, "k-Degree anonymity and edge selection: Improving data utility in large networks," *Knowl. Inf. Syst.*, vol. 50, no. 2, pp. 447–474, Feb. 2016.

[8] Y. Sun, Y. Yuan, G. Wang, and Y. Cheng, "Splitting anonymization: A novel privacy-preserving approach of social network," *Knowl. Inf. Syst.*, vol. 47, no. 3, pp. 595–623, Jun. 2016.

[9] K. R. Macwan and S. J. Patel, "k-degree anonymity model for social network data publishing," *Adv. Electr. Comput. Eng.*, vol. 17, no. 4, pp. 117–124, Nov. 2017. doi: 10.4316/AECE.2017.04014.

[10] Y. Dongran, H. Zhao, L. Wang, P. Liu, and X. Li, *A Hierarchical K-Anonymous Technique of Graphlet Structural Perception in Social Network Publishing* (Lecture Notes in Computer Science), 2019, pp. 224–239.

[11] S. Bhagat, G. Cormode, B. Krishnamurthy, and D. Srivastava, "Privacy in dynamic social networks," in *Proc. Int. Conf. World Wide Web*, Apr. 2010, pp. 1059–1060.

[12] X. Ding, L. Zhang, Z. Wan, and M. Gu, "De-anonymizing dynamic social networks," in *Proc. IEEE Global Telecommun. Conf.*, Dec. 2011, pp. 1–6.

[13] C. J. L. Wang, E. T. Wang, and A. L. P. Chen, "Anonymization for multiple released social network graphs," in *Proc. Pacific-Asia Conf. Knowl. Discovery Data Mining*, 2013, pp. 99–110.

[14] L. Rossi, M. Musolesi, and A. Torsello, "On the k-anonymization of time-varying and multi-layer social graphs," in *Proc. 9th Int. Conf. Web Social Media (ICWSM)*, 2015.

[15] M. Kiabod, M. N. Dehkordi, and B. Barekatain, "TSRAM: A time-saving k-degree anonymization method in social network," *Expert Syst. Appl.*, vol. 125, pp. 378–396, Jul. 2019.

[16] A. Campan, Y. Alufaisan, and T. M. Truta, "Preserving communities in anonymized social networks," *Trans. Data Privacy*, vol. 8, no. 1, pp. 55–87, Apr. 2015.

[17] H. Wang, P. Liu, S. Lin, and X. Li, "A local-perturbation anonymizing approach to preserving community structure in released social networks," in *Proc. Int. Conf. Heterogeneous Netw. Qual., Rel., Secur. Robustness*, 2016, pp. 36–45.

[18] S. Kumar and P. Kumar, "Upper approximation based privacy preserving in online social networks," *Expert Syst. Appl.*, vol. 88, pp. 276–289, Dec. 2017.

[19] F. Rousseau, J. Casas-Roma, and M. Vazirgiannis, "Community-preserving anonymization of graphs," *Knowl. Inf. Syst.*, vol. 54, no. 2, pp. 315–343, Feb. 2017.

[20] K. R. Macwan and S. J. Patel, "k-NMF anonymization in social network data publishing," *Comput. J.*, vol. 61, no. 4, pp. 601–613, Feb. 2018.

[21] M. E. J. Newman, "Fast algorithm for detecting community structure in networks," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 69, no. 6, 2004, Art. no. 066133.

[22] C. H. Tai, P. J. Tseng, P. S. Yu, and M. S. Chen, "Identity protection in sequential releases of dynamic networks," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 3, pp. 635–651, Mar. 2014.

[23] W. Yongcheng, "A survey of complex network community discovery algorithms," *Res. Telecommun. Technol.*, 2015.

XIAOLIN ZHANG was born in Baotou, China, in December 1966. She received the bachelor's degree in computer science and technology from Northeastern University, in 1988, the master's degree in autochemistry from the Beijing University of Science and Technology, in 1995, and the Ph.D. degree in computer science and technology from Northeastern University, in 2006.

Since 1988, she was with the Inner Mongolia University of Science and Technology, where she is currently the Deputy Director of the Professor Committee of the Information Technology College, the Head of the Computer Science Department, and the Director of the Department of Computer Science. She have trained more than 60 master's degree students. She have been trained 15 master's degree students. She has published over 60 academic papers, including more than 20 articles in EI and 2 articles in SCI. She is responsible for many projects such as the National Natural Science Foundation of China, the National Social Science Fund Project, the Chunhui Project of the Ministry of Education, the Natural Science Foundation of Inner Mongolia Project, and the Inner Mongolia Education Department Fund Project. Her current research interests include big data processing technology, social network privacy protection technology, XML databases, XML data stream, wireless sensor networks, uncertain databases, and flame image databases.

Prof. Zhang is a member of the Chinese Computer Society, the Information System Professional Committee, China Computer Society, and the Director of the Inner Mongolia Autonomous Region Computer Society.

JIAO LIU was born in Tangshan, China, in September 1995. She received the B.S. degree in medical information engineering from Taishan Medical College, China, in 2017. She is currently pursuing the master's degree in computer science and technology with the Inner Mongolia University of Science and Technology. Her research interests include big data processing technology and social network privacy protection technology.

JIAN LI was born in Hulunbeier, China, in June 1995. He received the B.S. degree in software engineering from the Inner Mongolia University of Science and Technology, in 2017, where he is currently pursuing the master's degree in computer science and technology. His research areas include big data processing technology and social network privacy protection technology.

LIXIN LIU was born in Baotou, China, in 1984. She received the bachelor's degree in information security from Central South University, in 2007, and the master's degree in computer science and technology from Central South University, in 2010. She is currently pursuing the Ph.D. degree with Renmin University, under the supervision of X. Meng.

Since 2010, she has been with the Inner Mongolia University of Science and Technology, where she is currently a Lecturer with the Department of Computer Science, School of Information Engineering. She has presided over one provincial and ministerial level scientific research project, one school-level project, four scientific research projects, and four academic articles, including one EI journal. Her main research interests include big data storage and management, and big data security.