Research on Dynamic Social Network Anonymity Technology for Protecting Community Structure

Na Li[1], Xiao-Lin Zhang[1], Yong-Ping Wang[1], Jian Li[1], and Li-Xin Liu[2]
(Corresponding author: Xiao-Lin Zhang)

School of Information Engineering, Inner Mongolia University of Science and Technology[1]
Baotou 014010, China
School of Information, Renmin University of China[2]
Beijing 100872, China
Email: zhangxl@imust.edu.cn
(Received Feb. 8, 2020; Revised and Accepted Aug. 23, 2020; First Online May 30, 2021)

Abstract

The dynamic change of vertex degree in a dynamic social network will lead to vertex identity disclosure. In view of the deficiencies in current privacy protection methods, such as the destruction of community structure and low data processing capability of single workstation. The dynamic social network degree sequence anonymity (DSNDSA) method to protect community structure is proposed. The method obtains the grouping and anonymous results based on compressed binary tree constructed by a new method which is called multidimensional vector. Dummy vertices are added in parallel to construct anonymous graphs. Distributed to merge dummy vertices method based on community is designed to reduce the number of vertices added under the premise of satisfying the anonymity model. A divide and agglomerate algorithm is expanded for community detection. The experimental results show that the proposed algorithm based on GraphX can overcome the defects of the traditional algorithm in community protection while meeting the requirement of anonymity.

Keywords: Compressed Binary Tree; Community Structure; Divide and Agglomerate; Dynamic Social Network Anonymity; GraphX

1 Introduction

With the advent of web 2.0, social networking platforms are becoming more and more popular,such as Facebook, Twitter, LinkedIn and Google+ [1]. These social networking platforms has generated a large amount of data.Massive social network data attracts vast researchers with its great research value.Because these data carries the user's privacy informations inevitably, it is imperative for the data owner to protect the privacy of the information before releasing the data to the third-party.

In order to resist attacks of various background knowledge on users′privacy, a variety of social network anonymization technologies have emerged in recent years. These technologies are mainly focused on single graph anonymity [12]. Due to the existence of a large number of incremental social networks in real life, attackers are likely to re-identify the target vertex with the vertex degree information on two consecutive timestamps as background knowledge. Therefore, it has practical significance to abstract the dynamic social network graph into a set of simple incremental sequences and protect its privacy [7]. In addition, many real-world networks are organized according to a community structure intimately. Much research effort has been devoted to develop methods and algorithms that can efficiently highlight this hidden structure of a network [6]. However, the existing dynamic social network privacy protection models still has great limitation on community structure protection. This has affected that researchers attempts to analyze the characteristics of social networks according to the network community structure, which has reduced the availability of data greatly.

To solve the above problems, a distributed anonymity algorithm of dynamic social network is designed to protect the community structure. The main contributions of this paper are as follows:

1) Aiming at the undirected graph of dynamic social network, a degree sequence attack model is defined, and a dynamic social network degree sequence anonymity model for protecting community structure is proposed, thus preventing privacy attacks by attackers with vertex degree sequences as background knowledge effectively.

2) A privacy protection algorithm of dynamic social network k-degree sequence is designed to protect the community structure while ensuring that anonymous graphs satisfies k-degree sequence anonymity on dif-

ferent timestamps.

3) Experimental tests and analysis on real data sets verifies that the validity of dynamic social network degree sequence anonymity model and the high availability of dynamic social network anonymous graphs in the aspect of community structure.

The rest of this paper is organized as follows. Section 2 reviews the previous related research in more detail. Section 3 formalizes related definition and privacy model. Section 4 presents the DSNDSA algorithm and we conduct experiments on real data sets in Section 5. Finally, the conclusion of the paper is given in Section 6

2 Related Work

Recent development in the technology has made it easier to collect massive amounts of social network data and leads to serious privacy problems. Regarding the privacy information to be preserved in social networks, three main categories of privacy threats have been identified [5]: identity disclosure [4, 7, 8, 10, 14, 15, 21, 22], attribute disclosure [3] and link disclosure [16, 18, 20]. Due to the diversity of privacy threats, more and more researchers has focused their attention on the protection of social network privacy and proposed a variety of anonymity methods.K-anonymity framework is a classic anonymity framework for social networks.K-anonymity requires that for any element in a set, there are at least k-1 duplicate elements that are the same as it, *i.e.* any element can be identified with a probability no greater than 1/k in a set. This paper mainly studies the identity disclosure of social network vertex based on k-anonymity framework.

Identity disclosure includes sub-categories such as vertex existence, vertex properties and graph metrics [5]. Literature [10] introduces a time-saving k-degree anonymization method TSRAM in social network that without having to rescan the data set for different levels of anonymity. It ensures that the attacker takes vertex degree as background knowledge and the probability of successfully identifying the target vertex does not exceed 1/k. Literature [21] designs a general k-anonymization framework, which can be used with various utility measurements to achieve k-anonymity with small utility loss on given social networks. In this method, utility measurements are designed based on more complex community-based graph models includes flat community-based utility model and Hierarchical community-based utility model. Literature [22] addresses the problem of excessive loss of graphlet structural information in the privacy process of published social network data, and proposed a technique of hierarchical k-anonymity for graphlet structural perception. The mehod divide the degrees of nodes according to the degree to which the social networking graphical node obeys the characteristics of a power-law distribution, and the divided nodes define the different privacy levels according to their practical means. The purpose is meet the privacy requirement while protecting the graphical structural information in the social network and improving the utility of the data. Literature [14] considers protecting the weighted social networks from weight-based attacks and propose a method KWGA based on the weighted social networks. And This method combines k-anonymous with generalization method to ensure the security of the social network data when it is published. Literature [15] proposes an improved k-degree anonymity model that provides privacy with low utility loss. This method performs anonymous operations on the basis of dividing communities, making the vertices of the same group indistinguishable based on the degree value.

Social network data publishing is dynamic. Literature [7] proposes a weighted graph incremental sequence k-anonymous privacy protection model, and design a baseline anonymity algorithm WLKA based on weight list and HVKA algorithm based on hypergraph, which prevents the attacks from node point labels and weight packages. Literature [8] makes one-hop neighbor's network structure and label as attacker' background knowledge and define the label neighborhood attack model in dynamic social network releases. A dynamic-l-diversity anonymized method is proposed to resist attacks. And ensuring each vertex with a sensitive label, which can't be identified in the social network with a probability higher than 1/ l. Literature [18] abstracts social networks on different timestamps into a set of simple incremental sequences, and proposes a social network anonymity method DMRA for simultaneous publication of multiple social network graphs. The method ensures that the attacker can successfully infer that the probability of a user participating in any edge and the probability of edge connection between any two vertices are not more than 1/k without any background knowledge. Literature [9] proposes a dynamic k^w-Number of Mutual Friend anonymity algorithm for protecting edge identities of dynamic networks that is released sequentially. The k^w-NMF algorithm anonymizes each release of network data so that the adversary can not re-identify the victim by knowing the knowledge of each release. Literature [17] proposes a new privacy model, dynamic k^w-structural diversity anonymity, for protecting the vertex and multicommunity identities in sequential releases of a dynamic network.

In recent years, the discovery and analysis of community structures in social networks play an important role in studying the characteristics of complex networks. Correspondingly, the combination of social network anonymization and protection of community structure has attracted the attention of many scholars. Literature [11] uses the concept of upper approximation of the original set to propose a social network privacy protection method PPGP. The method can effectively protect the graphic community structure in the anonymous process. And it makes the anonymous social network graph have good performance in graph mining tasks such as clustering, classification and PageRank computation. Literature [19] proposes a novel local perturbation tech-

nique that can reach the same privacy requirement of k-anonymity, while minimizing the impact on community structure. Literature [23] proposes a probabilistic anonymizing method to protect the data privacy, which combines k-anonymous with random perturbation. The proposed method can minimize the impact on community structure.

To sum up, different social network anonymity methods can resist different privacy attacks. Most privacy protection models are aimed at a single social network graph, and single graph anonymity technology is not sufficient to cope with the dynamic changes of social networks. In addition, the existing dynamic social network privacy protection technologies ensures the availability of the original social network data while achieving anonymity. But they ignored the protection of the social network community structure. Therefore, this paper studies the vertex identity re-identification of dynamic social networks, taking the degree sequence of a vertex on different timestamps as the attacker's background knowledge. And a dynamic social network anonymity algorithm DSNDSA is proposed, which can protect the social network community structure in the process of social network anonymity effectively.

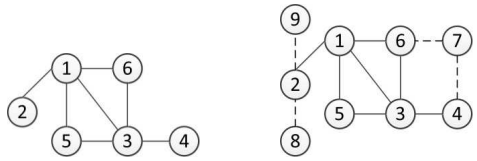
3 Definitions and Dynamic Social Network Degree Sequence Anonymity Model

Definition 1. *(Incremental dynamic social network graph) The sequence of social network graphs on different timestamps is denoted* $g = \langle G_0, G_1, ..., G_t \rangle$, *The social network graph at time t is denoted* $G_t = (V_t, E_t)$, *where* V_t *is a set of vetices representing users at time t,* E_t *is a set of edges representing the interaction among users at time t. With the passage of time, we assume t hat the vertices and edges of social network graph are not decreasing,* i.e. $V_{t-1} \subseteq V_t, E_{t-1} \subseteq E_t$. *The sequence g of social network graph like this is called incremental dynamic social network graph. Figure 1a and Figure 1b are incremental dynamic social network graphs on two consecutive timestamps.*

Definition 2. *(Degree sequence of vertex) Given a incremental dynamic social network graph* $g = \langle G_i, G_{i+1}, \cdots, G_t \rangle$. *if* $\forall v \in V_t$, *the degree of vertex v in social network graph* G_i *is denoted* $d_{(v,G_i)}$. $\Delta_v = (d_{(v,G_i)}, d_{(v,G_{i+1})}, ..., d_{(v,G_t)})$ *is called degree sequence of vertex v in incremental dynamic social network graph g.*

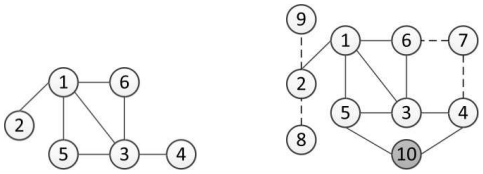
Definition 3. *(First category vertex and second category vertex) Given a incremental dynamic social network graph* $g = \langle G_{t-1}, G_t \rangle$. $V_g = \{v_1, v_2, ..., v_i, ..., v_n\}$ *is a set of all vertices in g.*v_i *belongs to the first category vertex if* $v_i \in V_{t-1}$ *and* $v_i \in V_t$. *Otherwise,*v_i *belongs to the second category vertex if* $v_i \notin V_{t-1}$ *and* $v_i \in V_t$.

Definition 4. *(Multidimensional vector) Given a incremental dynamic social network graph* $g = \langle G_{t-1}, G_t \rangle$ *($v \in V_t$). One of the Multidimensional vector corresponding to g is denoted* (Δ_v, C), *where* Δ_v *is degree sequence of vertex v in incremental dynamic social network graph g, C is the count of vertices whose degree sequence equal to* Δ_v.



(a) Original graph G_{t-1} at time t-1

(b) Original graph G_t at time t



(c) Anonymous graph G^*_{t-1} at time t-1

(d) Anonymous graph G^*_t at time t

Figure 1: Incremental dynamic social network original graph and anonymous graph

Definition 5. *(Dynamic social network vertex degree sequence attack) Given a incremental dynamic social network graph* $g = \langle G_{t-1}, G_t \rangle$ *on two consecutive timestamps. An attacker can successfully identify the target vertex v with the degree sequence* Δ_v *of vertex v as background knowledge, which is called dynamic social network vertex degree sequence attack.*

As shown in Figure 1a and Figure 1b, it is assumed that the attacker knows that the degree of the target vertex Bob in G_{t-1} and G_t is 1 and 2 respectively. Since the release graphs at both timestamps satisfy 2- degree anonymity, the probability of the attacker identifying Bob correctly is 1/2 at each timestamp. If the attacker combines the anonymous graphs on two timestamps, *i.e.* Δ_v=(1,2) as background knowledge, he can identify that the vertex 4 is Bob with 100% probability successfully. In the following, according to the dynamic social network vertex degree sequence attack model, the dynamic social network vertex degree sequence k-anonymity is defined.

Definition 6. *(Vertex k-degree sequence anonymity) Given a incremental dynamic social network original graph* $g = \langle G_{t-1}, G_t \rangle$ *on two consecutive timestamps and the privacy parameter k. Incremental dynamic social network anonymous graph is denoted* $g^* = \langle G^*_{t-1}, G^*_t \rangle$. *For the degree sequence* Δ_v *of any one first category vertex v(second category vertex), there are at least k-1 other first category vertices(second category vertices) with the same degree sequence in* g^*. *The degree of privacy protection increases with the increase of k. It is said that the* g^* *satisfies vertex k-degree sequence anonymity.*

As shown in Figure 1a and Figure 1b,in order to make the dynamic social network meets the anonymity requirement of vertex k-degree sequence, a dummy vertex 10 and two dummy edges e(4, 10) and e(5, 10) are added to G_t. The 2-degree sequence anonymity graphs are shown in Figure 1c and Figure 1d. For any vertex v, there is at least one vertex with the same degree sequence as vertex v, that is, the attacker cannot uniquely identify the target vertex with a probability greater than 1/2. Anonymous graphs satisfies vertex 2-degree sequence anonymity.

Definition 7. *(Dynamic social network degree sequence anonymity model) Given a incremental dynamic social network original graph* $g = \langle G_{t-1}, G_t \rangle$ *on two consecutive timestamps and a positive integer k which can adjust the degree of anonymity. If the Incremental dynamic social network anonymous graph* $g^* = \langle G^*_{t-1}, G^*_t \rangle$ *meets the following four requirements, It is said that the* g^* *conforms to the dynamic social network degree sequence anonymity model for protecting community structure.*

1) For the dynamic social network anonymous graph G^*_t at any timestamp, the probability that the attacker identifies the target vertex successfully based on the vertex degree does not exceed 1/k;

2) For the incremental dynamic social network anonymous graph $g^* = \langle G^*_{t-1}, G^*_t \rangle$, the probability that the attacker identifies the target vertex successfully based on the degree sequence of vertex does not exceed 1/k;

3) In the process of anonymity, original vertices and original edges does not change;

4) Social network community structure is protected in the process of social network anonymity.

4 Dynamic Social Network Degree Sequence Anonymity (DSNDSA) Algorithm for Protecting Community Structure

The solution to anonymize dynamic social network graph g is detailed in this section. The algorithm DSNDSA includes three steps:

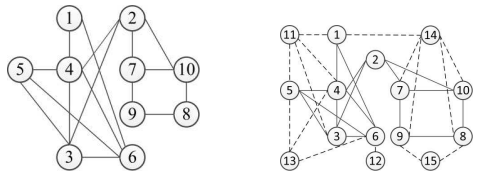
1) Community detection;

2) Vertex grouping and anonymity;

3) Graph reconstruction.

4.1 Community Detection

Given a incremental dynamic social network graph g = $\langle G_{t-1}, G_t \rangle$, the type and number of social network communities is unchanged with the passage of time. The social network vertices in G_{t-1} is divided into different communities by DA algorithm [13]. Each vertex added at timestamp t is regarded as a sub-graph that does not meet the community criterion, and the community to which the added vertex belongs is selected through the biggest AT index for the community to attract sub-graph [13]. Similarly, dummy vertices use the same community detection method.



(a) Original graph G_{t-1} at time t-1 (b) Original graph G_t at time t

Figure 2: Dynamic social network graph

Figure 2 is a dynamic social network original graph g = $\langle G_{t-1}, G_t \rangle$. The 10 vertices in G_{t-1} are divided into two communities using the DA algorithm, C_1={1,3,4,5,6} and C_2={2,7,8,9,10}. Taking the vertex 11 which is newly added in G_t as an example, the AT indexes of two communities to the vertex 11 are calculated respectively, *i.e.* $AT_{11,C_1} = 3/2, AT_{11,C_2} = 0$. Therefore, vertex 11 belongs to community C_1. Updating community collections C_1={1,3,4,5,6,11,12,13} and C_2={2,7,8,9,10, 14,15}.

4.2 Vertex Grouping and Anonymity

Vertex grouping and anonymity is to group vertices of dynamic social network with the objective of minimizing anonymity cost and to determine the anonymity degree sequence of vertices in the grouping. In order to achieve this goal, different grouping and anonymity methods are proposed for different categories of dynamic social network vertices.

4.2.1 Grouping and Anonymity for First Category Vertex

Given a incremental dynamic social network graph g = $\langle G_{t-1}, G_t \rangle$, all vertices in G_{t-1} belongs to the first category vertex. According to literature [10], the process of grouping and anonymity for the first category vertex includes the following three steps:

1) Generating Multidimensional Vector: According to the dynamic social network graph, the vertex degree sequences of the first category vertices are obtained, and one or more vertices with the same vertex degree sequence are expressed as multidimensional vectors.

2) Constructing Compressed Binary Tree: The multidimensional vectors are sorted to generate leaf nodes of a binary tree, and merging leaf nodes and calculating multidimensional vectors of parent nodes based on the "travel time" criterion. This process is iterated many times until the binary tree is constructed. The

compressed binary tree can reflect degree sequence of the first category vertices in g.

3) Cutting Line Drawing on the Tree: Cut a line on the tree according to the privacy parameter k to obtain the grouping result and anonymity degree sequence of the first category vertex. The higher the degree of anonymity, the closer the tangent is to the root.

In the above steps, the leaf nodes of binary tree are represented by multidimensional vectors composed of vertex degree sequences and counts.Since the social network is incremental, the dimensions of the multidimensional vectors corresponding to the first category vertices are consistent. In order to minimize the anonymity cost of the first category vertices anonymity, this paper sorts multidimensional vectors from left to right according to the following rules:

1) Given the degree sequence of the first category vertices $\Delta_v = (d_{(v,G_{t-1})}, d_{(v,G_t)})$, Sorting multidimensional vectors in descending order according to $\sum_{i=t-1}^{t} d_{(v,G_i)}$;

2) If $\sum_{i=t-1}^{t} d_{(v,G_i)}$ are same between them, Sorting in descending order according to the value of the first element in Δ_v, and so on.

Finally, the parent node generated by merging the two leaf nodes Lm and Ln is represented by multidimensional vector $(\max\{d_{(L_m,G_{t-1})}, d_{(L_n,G_{t-1})}\}, \max\{d_{(L_m,G_t)}, d_{(L_n,G_t)}\}, C_m + C_n)$. In the similarity calculation process, the potential fields are calculated for all the data based on Euclidean distance between multidimensional vectors, *i.e.* Euclidean distance in three-dimensional space corresponding to the first category vertex.

As shown in Figure 2, the set of first category vertices is {1,2,3,4,5,6,7,8,9,10}. Firstly, ten first category vertices are represented by seven multidimensional vectors, as shown in Table 1.

Table 1: Multidimensional vector corresponding to first category vertices in dynamic social network

Vector ID	Degree sequence	Count	Total degree	Multidimensional vector
1	(5,7)	1	12	(5,7,1)
2	(4,4)	1	8	(4,4,1)
3	(3,5)	1	8	(3,5,1)
4	(4,5)	1	9	(4,5,1)
5	(4,6)	1	10	(4,6,1)
6	(2,4)	3	6	(2,4,3)
7	(3,4)	2	7	(3,4,2)

Secondly, the purpose of calculating the similarity between multidimensional vectors is merging leaf nodes. For example, $S_{45} = \frac{\phi_4 - \phi_5}{r_{45}^2} = 0.7467602106$ and $S_{42} = \frac{\phi_4 - \phi_2}{r_{24}^2} = 0.773902479$. Because $S_{42} > S_{45}$,parent node (4,5,2) is generated by merging leaf node (4,5,1) and leaf node (4,4,1). Finally, the structure and cut line position of the compressed binary tree corresponding to the first category vertices is shown in Figure 3 when the anonymity parameter k is 3. Therefore, grouping results of the first category vertices is $Group_1$={2,3,4,6},$Group_2$={5,7,10} and $Group_3$={1,8,9}. The anonymity degree sequences corresponding to them are $\Delta_v^*(Group_1) = (5,7)$,$\Delta_v^*(Group_2) = (3,5)$ and $\Delta_v^*(Group_3) = (2,4)$.

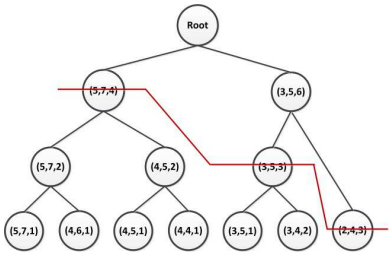


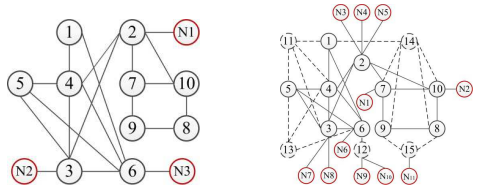
Figure 3: Compressed binary tree corresponding to the first category vertex

4.2.2 Grouping and Anonymity for Second Category Vertex

The essence of the second category vertex anonymity is single graph anonymity. The process of grouping and anonymity is divided into the following two steps:

1) Remove the second category vertices whose degree is in the set DSet(t) or who satisfies the k-anonymity condition themselves, and the remaining vertices are called the sequence of vertices who will be anonymized. And the elements in set DSet(t) are composed of the anonymious degrees of the first category vertices in G_t.

2) If the count of second category vertices who will be anonymized is not less than k, the anonymity process of it is similar to the first category vertex, where leaf nodes are represented by two-dimensional vectors. On the contrary, the anonymous degree with the smallest difference compared to its degree and greater than its own degree is selected as its target degree for every second category vertex who will be anonymized.



(a) G'_{t-1} (b) G'_t

Figure 4: Dynamic social network initial anonymous graph $\langle G'_{t-1}, G'_t \rangle$

As shown in Figure 2, the set of second category vertices is {11,12,13,14,15} and DSet(t)={4,5,7}. Therefore, the set of second category vertices who will be anonymized is {12,13,15}. The corresponding anonymious degrees are 3, 3 and 3 when k=3 separately.

The algorithm DSNDSA adds dummy vertices to the original dynamic social network in parallel based on GraphX to achieve the requirement of vertex k-degree sequence anonymity. The initial anonymious graph of dynamic social network is shown in Figure 4.

4.3 Graph Reconstruction

In order to reduce the number of dummy vertices in the social network publishing graphs and improve the usability of community structure, DSNDSA algorithm designs dummy vertex removal-addition conditions and rules based on the community to which the vertex belongs. Priority is given to dummy vertex removal-addition operations in the same community, and then dummy vertex removal-addition operation between different communities is performed.

The graph reconstruction process is implemented based on the distributed graph processing system spark graphX, which follows the characteristic of "node-centered" and improves the efficiency of privacy protection technology in processing large-scale graph data by means of message transmission between vertices. In order to select a suitable dummy neighbor vertex, n-hop dummy neighbor table(NDT) information is needed. The vertex data structure is represented by quintuple(NID,deg(N_u),com, type, tag), and each quintuple is a dummy neighbor table entry(DNTE).

1) NID:the vertex ID;

2) Deg(N_u):the degree of vertex u;

3) Com: community to which vertex u belongs;

4) Type:type of vertex u. Type=0 means that vertex u belongs to dummy vertex, type=1 and type=2 means that vertex u belongs to the first category vertex and the second category vertex respectively;

5) Tag: whether degree deg(N_u) of vertex u exists in anonymous degree set DSet(i), if so, tag=1, otherwise, tag=0. Therefore, all vertices tag=0 before adding dummy vertices.

Definition 8. *(Removal-addition condition in same community, RACSC) If dummy vertex N_u and N_v can remove-add in the same community, then N_u and N_v must meet the following three conditions at the same time:*

1) $deg(N_u)+deg(N_v) \leqslant SC\_deg_{max}$, where SC_deg_{max} represents the maximum anonymious degree of vertices in the community to which N_u belongs;

2) For the vertex N_w with type$\neq$0, the edge e(N_u,N_w) and the edge e(N_v,N_w) do not exist at the same time;

3) $N_u.com$=$N_u.com$.

Definition 9. *(Removal-addition rule in same community, RARSC) For any vertex N_u with type=0, the DNT of N_u is obtained through message transfer mechanism. For any vertex N_v in DNT is placed in the candidate set $N_u.CandiSet\_sc$ of N_u if it meets RACSC. The removal-addition rules in same community are as follow:*

1) If the element in the set $N_u.CandiSet\_sc$ is unique, it is the best candidate vertex of N_u;

2) If the number of elements in the set $N_u.CandiSet\_sc$ is greater than 1, the dummy vertex N_v with small value of $deg(N_u)+deg(N_v)$ is considered for remove-adding preferentially.

Any dummy vertex N_u selects the best dummy vertex algorithm SCS as follows:

Algorithm 1 Same Community Select (SCS)

Input: N_u.CandiSet_sc
Output: N_v

```
1: if (Nu.CandiSet_sc.size > 1) then
2:   for (each N_v in N_u.CandiSet_sc) do
3:     deg(N_r)=deg(N_u)+deg(N_v);
4:   end for
5:   M=the number of dummy nodes with degree equal
     to min(deg(N_r))
6:   if (M = 1) then
7:     return N_v
8:   else
9:     randomly select N_v
10:    return N_v
11:  end if
12: else
13:   return N_v
14: end if
```

Algorithm 2 shows dummy vertex removal-addition algorithm SCRA in the same community. Lines 3-22 remove-adds dummy vertices in parallel after all supersteps are completed. Among them, lines 2-4 looks for dummy vertex neighbor information based on Pregel model. Lines 5-9 will form a candidate set of dummy vertices N_u with the virtual neighbor information that meets the RACSC. Lines 10-18 remove-adds dummy vertices N_u and N_v.

In each superstep before parallel removal-addition of dummy vertices, DSNDSA algorithm ensures that each dummy vertex is remove-added at most once by traversing the dummy vertices and updating the candidate set of dummy vertices continuously. Taking the initial anonymous graph G'_t at time t in Figure 3b as an example, the dummy vertex receives its own 3-hop dummy vertex neighbor information, thus obtaining the dummy vertex candidate set as shown in Table 2 when superstep=3. Virtual vertex N_1 selects N_2 as the best candidate vertex according to algorithm 1, and the candidate set information

updated for the first time is shown in column 3 of Table 2. By analogy, the result of selecting the best candidate vertex for all dummy vertices is shown in column 5 of Table 2. The state of virtual vertices N_1, N_2, N_6 and N_7 is set to Inactive. This means that they will not participate in the other supersteps after superstep=3. And the dummy vertex removal-addition operation is performed after all supersteps are completed. The algorithm iterates twice, and the result of remove-adding dummy vertices in the same community is shown in Figure 5.

Algorithm 2 Same Community Remove_Add (SCRA)

Input: G'
Output: $G^{\#}$

for (SuperStep=1 to 6) **do**
  sendMessToNeighbors
  for (each dummy vertex N_u in G'_i) **do**
    update N_u.DNT
    for (each N_v in N_u.DNT) **do**
      if (N_v satify RACSC) **then**
        N_u.CandiSet_sc $\leftarrow N_v$
      end if
    end for
    if (N_u.CandiSet_sc $\geqslant$ 1) **then**
      N_v=Same Community Select(N_u.CandiSet_sc)
      G′.EdgeRDD.Remove $\langle$ m,N_u $\rangle$
      G′.EdgeRDD.Remove $\langle$ n,N_v $\rangle$
      G′.EdgeRDD.Add $\langle$ m,N_r $\rangle$
      G′.EdgeRDD.Add $\langle$ n,N_r $\rangle$
      NDRDD.Add (N_u,N_v)
      VoteToHalt (N_u,N_v)
    end if
  end for
end for
return $G^{\#}$

Table 2: Dummy vertex candidate set update table when surperstep=3

Dummy vertex number	candidate set	first update	second update	best candidate vertex
N_1	N_2,N_3,N_4,N_5	N_2	N_2	N_2
N_2	N_1,N_3,N_4,N_5	N_1	N_1	N_1
N_3	N_1,N_2	∅	∅	
N_4	N_1,N_2	∅	∅	
N_5	N_1,N_2	∅	∅	
N_6	N_7,N_8,N_9,N_{10}	N_7,N_8,N_9,N_{10}	N_7	N_7
N_7	N_6	N_6	N_6	N_6
N_8	N_6	N_6	∅	
N_9	N_6	N_6	∅	
N_{10}	N_6	N_6	∅	
N_{11}	∅	∅	∅	

Definition 10. *(Removal-addition condition in different communities, RACDC) If dummy vertex N_u and N_v can remove-add in the different communities, then N_u and N_v must meet the following three conditions at the same time:*

1) $deg(N_u)$+$deg(N_v)$ $\leqslant$ deg_{max} and $deg(N_u)$+$deg(N_v)$ $\in$ DSet(i), where deg_{max} represents the maximum anonymious degree in current social network;

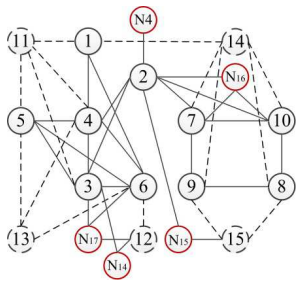


Figure 5: Result of remove-adding dummy vertices in the same community

2) For the vertex N_w of type $\neq$ 0, the edge e(N_u,N_w) and the edge e(N_v,N_w) do not exist at the same time;

3) N_u.tag$\wedge N_v$. Tag = 0.

Algorithm 3 Different Community Select(DCS)

Input: N_u.CandiSet_dc
Output: N_v

if (N_u.CandiSet_dc.size > 1) **then**
  for (each N_v in N_u.CandiSet_dc) **do**
    if (N_u.tag=0 && N_v.tag=0) **then**
      $List1_u \leftarrow N_v$
    else
      $List2_u \leftarrow N_v$
    end if
  end for
  if ($List1_u$.size!=0) **then**
    List= $List1_u$
  else
    List= $List2_u$
  end if
  for (each N_v in List) **do**
    deg(N_r)=deg(N_u)+deg(N_v)
  end for
  M=the number of dummy nodes with degree equal to min(deg(N_r))
  if (M = 1) **then**
    return N_v
  else
    randomly select N_v
    return N_v
  end if
else
  return N_v
end if

Definition 11. *(Removal-addition rule in different communities, RARDC) For any vertex N_u with tag=0, the DNT of N_u is obtained. For any vertex N_v in DNT is placed in the candidate set N_u.CandiSet_dc of N_u if it meets RACDC. The removal-addition rules in different communities are as follow:*

1) If the element in the set N_u.CandiSet_dc is unique, it is the best candidate vertex of N_u;

2) If the number of elements in the set N_u.CandiSet_dc is greater than 1, dummy vertex N_v with tag =0 is selected for remove-adding preferentially, and then select dummy vertex N_v who has minimum value of deg(N_u)+deg(N_v).

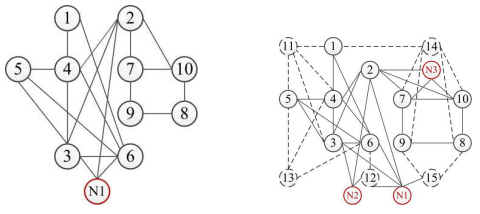
Any dummy vertex N_u selects the best dummy vertex algorithm in different communities DCS as show in Algorithm 3.

If there is dummy vertex N_u in the social network with tag=0 after algorithm 2 is executed, the dummy vertex removal-addition in different communities algorithm as shown in algorithm 4 is executed once. Lines 2-9 obtains candidate vertex sets of dummy vertices with tag=0. And lines 10-18 remove-adds dummy vertices to obtain social network publishing graph G_i^*.

Algorithm 4 Different Community Remove_Add(DCRA)

Input: $G^{\#}$
Output: G^*

1: **for** (SuperStep = 1 to 6) **do**
2: sendMessToNeighbors
3: **if** (dummy node N_u.tag=0 in $G_i^{\#}$) **then**
4: update N_u.DNT
5: **for** (each N_v in N_u.DNT) **do**
6: **if** (N_v satify RACDC) **then**
7: N_u.CandiSet_dc $\leftarrow N_v$
8: **end if**
9: **end for**
10: **if** (N_u.CandiSet_dc $\geqslant$ 1) **then**
11: N_v=Different Community Select(N_u.CandiSet_dc)
12: $G^{\#}$.EdgeRDD.Remove $\langle$ m,N_u $\rangle$
13: $G^{\#}$.EdgeRDD.Remove $\langle$ n,N_v $\rangle$
14: $G^{\#}$.EdgeRDD.Add $\langle$ m,N_r $\rangle$
15: $G^{\#}$.EdgeRDD.Add $\langle$ n,N_r $\rangle$
16: NDRDD.Add (N_u,N_v)
17: VoteToHalt (N_u,N_v)
18: **end if**
19: **end if**
20: **end for**
21: return G^*



(a) G_{t-1}^* (b) G_t^*

Figure 6: Dynamic social network release graph

In Figure 5, dummy vertices N_4, N_{14}, and N_{15} does not satisfy anonymity requirement. And the social network release graph G_t^* is obtained by executing the DCRA algorithm as shown in Figure 6b. All vertices meets the anonymity requirement of k=3.

DSNDSA algorithm is shown in Algorithm 5. The corresponding dynamic social network release graph of Figure 2 is shown in Figure 6.

Algorithm 5 DSNDSA

Input: $g = \langle G_{t-1}, G_t \rangle$,k
Output: g^*=$\langle G_{t-1}^*, G_t^* \rangle$

1: Obtaining social network anonymious degree sequences on different timestamps
2: **for** (i=t-1 to t) **do**
3: Adding dummy vertices to original graph G_i according to anonymious degree sequence to generate initial anonymous graph G_t'
4: Initial: G_i',DNRDD=$\varnothing$, N_u.CandiSet_sc=$\varnothing$, N_u.CandiSet_dc=$\varnothing$
5: EdgeRDD=G_i'.EdgeRDD
6: update Det(i)
7: **while** ($\exists N_u, N_v \in G_i'$ && N_u, N_v satisfy RACSC) **do**
8: $G_i^{\#}$ = SCRA(G_i')
9: **end while**
10: **if** ($\exists$ $N_u \in G_i^{\#}$ && N_u.tag!=1) **then**
11: G_i^*=DCRA ($G_i^{\#}$)
12: **else**
13: G_i^*= $G_i^{\#}$
14: **end if**
15: $g^* \leftarrow G_i^*$
16: **end for**
17: return g^*

5 Experimental Results

This section analyzes and evaluates DSNDSA algorithm performance. The DSNDSA algorithm is compared with the dynamic k^w-number of mutual friend anonymity algorithm proposed by Jyothi [9] and the dynamic k^w-structure diversity anonymity algorithm proposed by Tai [17]. The experiment is tested using real social network datasets: Caida, Super User and wiki-talk. Among them, the Caida dataset is a relationships dataset that contains 122 CAIDA AS graphs from January 2004 to November 2007. The graph data of the network on 7 timestamps were obtained in the experiment. The Super User dataset is a temporal network of interactions on the stack exchange web site Super User. Edges (u, v, t) represents that user u answered user v's question at time t. The wiki-talk dataset is a temporal network representing Wikipedia users editing each other's Talk page.Edges (u, v, t) means that user u edited user v's talk page at time t. Dataset statistics is shown in Table 3.

In this paper, the directed graphs are processed to undirected graphs before the experiment. Experimental environment: CPU 1.80GHz, RAM 16GB, Hadoop 2.7.2,

Spark 2.4.3, programming language Scala 2.13.0 and 15 computing nodes.

Table 3: Dataset statistics

dataset	vertices	temporal edges	time span
Caida	26475	106762	122graphs
Super User	167981	430033	2773 days
wiki-talk	1140149	7833140	2320 days

5.1 Information Loss

In order to measure the influence of DSNDSA algorithm on graph structure in anonymous process, the average betweenness (BW) and average path length (APL) are used to evaluate the algorithm in the experiment. The BW is defined as the average of betweenness centrality of all vertices. The betweenness centrality of a vertexis calculated as Equation (1),where σ_{st} is the number of shortest paths from vertex s to vertex t and $\sigma_{st}(v)$ is the total number of those paths that pass through vertex v length of vertices u and v[16].

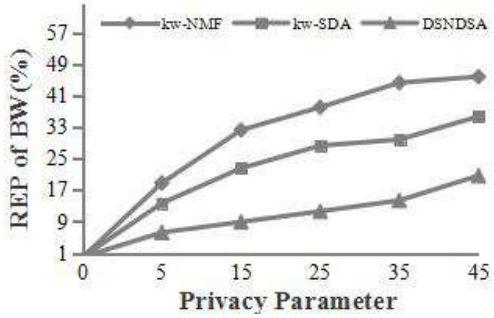
$$g(v) = \sum_{s \neq v \neq t} \frac{\sigma_{st}(v)}{\sigma_{st}} \tag{1}$$

In order to express the information loss after anonymity intuitively, using the relative error percentage(REP) to measure the change of graph structure. As shown in Equation (2), G and G* represents the graph structure values in the original and anonymous social network graph respectively. The lower the value, the smaller the information loss.

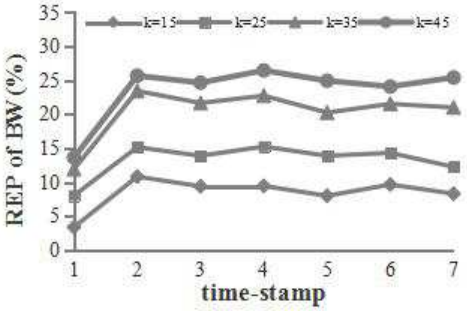
$$REP = \frac{|G - G^*|}{|G|} \times 100\% \tag{2}$$

Figure 7a shows the relative error percentage of BW after anonymity. With the increase of privacy parameter k, the relative error percentage curves of BW corresponding to different methods shows an upward trend, this means that the loss of information increases with the degree of privacy protection. The relative error percentages of k^w-NMF algorithm and k^w-SDA algorithm are both larger than DSNDSA algorithm after anonymity. Overall, DSNDSA algorithm has the best effect in ensuring the utility of graph structure.

Figure 7b shows the BW on each group data of Caida dataset, *i.e.* the relative error percentage of BW with time t under different k settings. Six groups of anonymous graphs are obtained when t=7. The relative error percentage increases with the increase of k on six groups of anonymous graphs. When k is fixed, t = 1 corresponds to the smallest relative error percentage for single graph anonymity. The relative error percentage of BW does not change much when t is other value, this is because the scale of each group of dynamic social network increases slightly with the passage of time.



(a) Super User



(b) Caida

Figure 7: Average betweenness

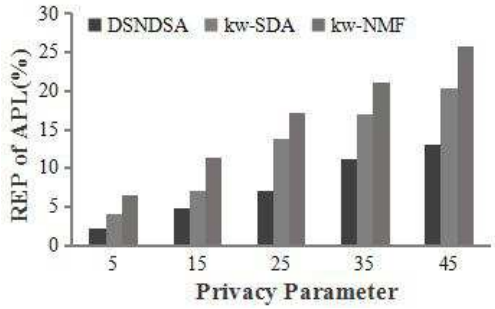
Figure 8a shows the average path lengths (APL) under different k settings on the Caida dataset. The low degree of privacy protection corresponds to the low relative error percentage of APL. The relative error percentage of APL corresponding to DSNSDSA algorithm is smaller, which shows that our method can better guarantee the structural properties of dynamic social network graph than k^w-NMF algorithm and k^w-SDA algorithm.

Figure 8b shows the relative error percentage of APL with time t on the Caida dataset. When k reaches 35, the relative error percentage of APL does not exceed 11.97.

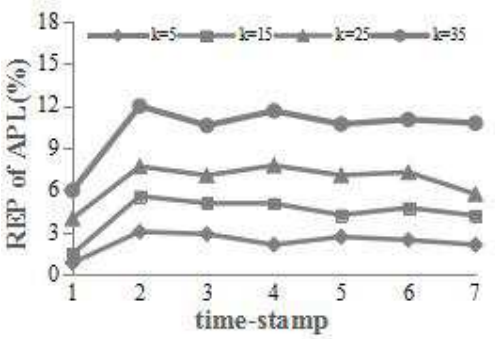
5.2 Availability of Community Structure

Suppose that the community sets in the original and anonymous social network are $C = \{C_1, C_2, \cdots, C_n\}$ and $C^* = \{C_1^*, C_2^*, \cdots, C_m^*\}$ respectively. Jaccard similarity coefficient can measure the similarity of social network community structure before and after anonymity. Jaccard similarity coefficient is expressed as a percentage. The greater its value, the higher the degree of protection of anonymity algorithm to the community structure of the original social network.

Figure 9a shows the variation of Jaccard similarity coefficient with k on wiki-talk dataset. And the vertical axis reflects the degree to which the anonymous graph preserves the original community structure. With the increase of k, the Jaccard similarity coefficients corresponding to the three methods shows a downward trend. For the case of larger k, DSNDSA algorithm protects the community structure more than 69.6%. Compared with the other two algorithms, it has better effect in community struc-

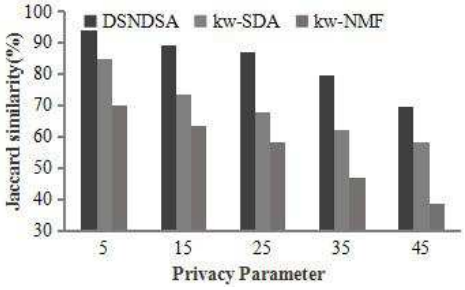


(a) Caida

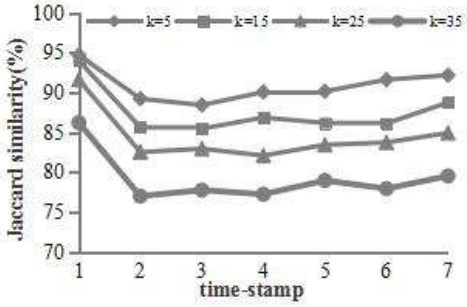


(b) Caida

Figure 8: Average path lengths



(a) wiki-talk



(b) Caida

Figure 9: Jaccard similarity coefficient

ture protection. Figure 9b shows the variation of Jaccard similarity coefficient with time t on Caida dataset. It can be observed that in the process of anonymity for two consecutive moments, the degree of protection of the anonymous graph to the original community structure will not decrease significantly with the increase of k.

Normalized Mutual Information(NMI) can measure the similarity between the community detection results of the algorithm and the real results. The paper takes the result of social network community detection before anonymity as the real result and compares it with the community structure after anonymity. Assuming that x and y represents two specific division results of the network respectively. The greater the NMI, the more information x and y can provide to each other and the closer they are. The calculation is shown in Equation (3).

$$U(X,Y) = \frac{2I(X,Y)}{H(X)+H(Y)} \tag{3}$$

Where,

$$\begin{aligned} I(X,Y) =& H(X) - H(X|Y) \\ =& \sum_{y\in Y}\sum_{x\in X} p(x,y)\log(\frac{p(x,y)}{p(x)p(y)}) \end{aligned} \tag{4}$$

p(x,y) represents the joint distribution probability of x and y, and adjusts the mutual information to 0-1 with an expected value of 1.

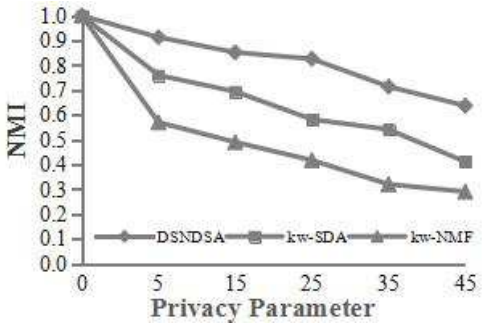
Figure 10a shows the variation of NMI with k on Super User dataset. With the increase of k, the NMI value tends to decrease, but the overall value is close to 1. This shows that DSNDSA algorithm has higher data availability than other algorithms in terms of community structure. Figure 10b shows the variation of NMI with time t on Caida dataset. We observe that each group of anonymous graphs can well protect the original social network community structure.

The precision index [2] can measure the change of the community to which the vertex belongs in the anonymity process. The precision index can be defined as Equation (5). If the community to which the vertex belongs remains unchanged after anonymity, the value of $\rho_{l_{tv}(v)=l_{pv}(v)}$ is 1; On the contrary, the value of $\rho_{l_{tv}(v)=l_{pv}(v)}$ is 0. The precision index is a value in range [0, 1]. The higher the value, the higher the usability of the community structure.

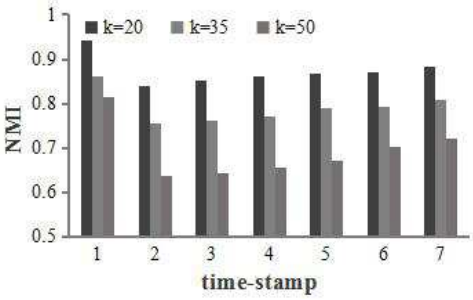
$$Precision\ index = \frac{1}{n}\sum\nolimits_{v\in G}\rho_{l_{tv}(v)=l_{pv}(v)} \tag{5}$$

Figure 11a shows the variation of precision index with k on Super User dataset. The precision index value decreases with the increase of k. Since DSNDSA algorithm considers the community structure in the process of anonymity, so the precision index is close to 1 after anonymity. The DSNDSA algorithm is better than the other two algorithms in terms of community structure availability.

Figure 11b shows the variation of precision index with time t on Caida dataset. When k is fixed and t is not less than 2, the precision index changes little, because the size of each group of dynamic networks increases slightly with time.

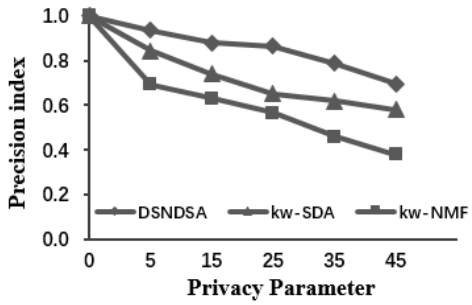


(a) Super User

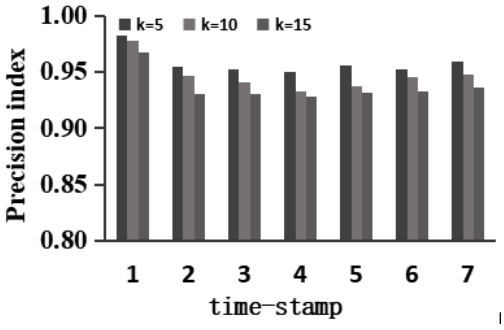


(b) Caida

Figure 10: Normalized mutual information



(a) Super User



(b) Caida

Figure 11: Precision index

6 Conclusion

The proliferation of online network data leads the dynamic network analysis and related privacy issues to become more important. This paper studies the privacy protection of dynamic social networks. The paper defines a vertex degree sequence attack model for dynamic social networks and proposes a distributed k-anonymity algorithm for dynamic social networks. The algorithm constructs a compressed binary tree to obtain vertex anonymity degree sequence, and adds dummy vertices to obtain dynamic social network anonymious graph. In addition, the algorithm merges dummy vertices in parallel based on the community to which the vertices belongs in order to improve the usability of the published graph in community structure. Experiments on real datasets shows that the algorithm in this paper can prevent the identity disclosure of dynamic social network vertices effectively while preserving the social network community structure and other graph structure properties, such as average betweenness, average path length, etc.

Acknowledgments

This work is partially supported by Natural Science Foundation of China (No.61562065) and Natural Science Foundation Project of Inner Mongolia (No.2019MS06001). The authors also gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

References

[1] A. Bhardwaj, V. Avasthi, and S. Goundar, "Impact of social networking on indian youth - a survey," *International Journal of Electronics and Information Engineering*, vol. 7, no. 1, pp. 41–51, 2017.

[2] B. J. Cai, H. Y. Wang, H. R. Zheng, and H. Wang, "Evaluation repeated random walks in community detection of social networks," *Proceedings of the Ninth International Conference on Machine Learning and Cybernetics*, vol. 4, pp. 1849–1854, 2010.

[3] C. P. Cao and X. Zheng, "Research of anonymity model for privacy-preserving in social network," *Journal of Chinese Computer Systems*, vol. 37, no. 8, pp. 1821–1825, 2016.

[4] J. Casas-Roma, J. Herrera-Joancomartí, and V. Torra, "k-degree anonymity and edge selection: Improving data utility in large networks," *Knowledge and Information Systems*, vol. 50, no. 2, pp. 447–474, 2017.

[5] J. Casas-Roma, J. Herrera-Joancomartí, and V. Torra, "A survey of graph-modification techniques for privacy-preserving on networks," *Artificial Intelligence Review*, vol. 47, no. 3, pp. 341–366, 2017.

[6] M. Coscia, F. Giannotti, and D. Pedreschi, "A classification for community discovery methods in com-

plex networks," *Statistical Analysis and Data Mining: The ASA Data Science Journal*, vol. 4, no. 5, pp. 512–546, 2011.

[7] C. H. Guo, B. Wang, H. J. Zhu, and X. C. Yang, "Incremental dynamic social network anonymity technology," *Journal of Computer Research and Development*, vol. 53, no. 6, pp. 1352–1364, 2016.

[8] X. Y. Hu, L. Wang, J. Q. Tang, C. Lei, P. Liu, and X. X. Li, "Anonymizing approach to resist label-neighborhood attacks in dynamic releases of social networks," in *IEEE 19th International Conference on e-Health Networking, Applications and Services (Healthcom'17)*, pp. 1–6, 2017.

[9] V. Jyothi and V. V. Kumari, "Privacy preserving in dynamic social networks," in *Proceedings of the International Conference on Informatics and Analytics*, pp. 1–8, 2016.

[10] M. Kiabod, M. N. Dehkordi, and B. Barekatain, "Tsram: A time-saving k-degree anonymization method in social network," *Expert Systems with Applications*, vol. 125, pp. 378–396, 2019.

[11] S. Kumar and P. Kumar, "Upper approximation based privacy preserving in online social networks," *Expert Systems with Applications*, vol. 88, pp. 276–289, 2017.

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

[13] Z. Y. Liu and Y. H. Ma, "A divide and agglomerate algorithm for community detection in social networks," *Information Sciences*, vol. 482, pp. 321–333, 2019.

[14] T. H. Ma, Y. Hao, X. F. Suo, Y. Xue, and J. Cao, "A weighted collaboration network generalization method for privacy protection in c-dblp," *Intelligent Data Analysis*, vol. 22, no. 1, pp. 3–19, 2018.

[15] K. R. Macwan and S. J. Patel, "k-degree anonymity model for social network data publishing," *Advances in Electrical and Computer Engineering*, vol. 17, no. 4, pp. 117–125, 2017.

[16] K. R. Macwan and S. J. Patel, "k-nmf anonymization in social network data publishing," *The Computer Journal*, vol. 61, no. 4, pp. 601–613, 2018.

[17] C. H. Tai, P. J. Tseng, S. Y. Philip, and M. S Chen, "Identity protection in sequential releases of dynamic networks," *IEEE transactions on Knowledge and Data Engineering*, vol. 26, no. 3, pp. 635–651, 2013.

[18] C. L. Wang, E. T. Wang, and A. L. Chen, "Anonymization for multiple released social network graphs," in *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pp. 99–110, 2013.

[19] H. J. Wang, P. Liu, S. Lin, and X. X. Li, "A local-perturbation anonymizing approach to preserving community structure in released social networks," in *International Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness*, pp. 36–45, 2016.

[20] S. L. Wang, Y. C. Tsai, T. P. Hong, and H. Y. Kao, "k(-)-anonymization of multiple shortest paths," *Soft Computing: A Fusion of Foundations, Methodologies and Applications*, vol. 21, pp. 4215–4226, 2017.

[21] Y. Z. Wang, L. Xie, B. H. Zheng, and K. C. Lee, "High utility k-anonymization for social network publishing," *Knowledge and Information Systems*, vol. 41, no. 3, pp. 697–725, 2014.

[22] D. R. Yu, H. X. Zhao, L. Wang, P. Liu, and X. X. Li, "A hierarchical k-anonymous technique of graphlet structural perception in social network publishing," in *International Conference on Mobile, Secure, and Programmable Networking*, pp. 224–239, 2018.

[23] Y. Zhao and Z. J. Li, "Privacy management in social network data publishing with community structure," in *The International Conference on Healthcare Science and Engineering*, pp. 141–151, 2018.

Biography

Na Li, is a graduate student in the School of Information Engineering, Inner Mongolia University of Science & Technology.Her research interests include Social Network privacy protection.

Xiao-lin Zhang, received the PhD degree from the Northeastern University of China, Shenyang, in 2006. She is a professor in the School of Information Engineering, Inner Mongolia University of Science & Technology. Her research interests include database theory and information security, Cloud Computing and Social Network privacy protection.

Yong-ping Wang, received the Master's degree from Wuhan University of Technology in 2010. She is a lecturer in the School of Information Engineering, Inner Mongolia University of Science & Technology. Her research interests include Data privacy protection.

Jian Li, is a graduate student in the School of Information Engineering, Inner Mongolia University of Science & Technology. His research interests include Community discovery and Social Network privacy protection.

Li-xin Liu, is a PhD candidate at Renmin University of China and the Member of China Computer Federation. Her main research interests include privacy protection and blockchain.