User Identity De-anonymization Based on Attributes

Cheng Zhang[1], Honglu Jiang[1,2(✉)], Yawei Wang[1], Qin Hu[1], Jiguo Yu[2,3,4], and Xiuzhen Cheng[1]

[1] The George Washington University, Washington, DC, USA
{zhangchengcarl,hljiang0720,yawei,qinhu,cheng}@gwu.edu
[2] Qufu Normal University, Rizhao, China
[3] Qilu University of Technology (Shandong Academy of Sciences), Jinan, China
jiguoyu@sina.com
[4] Shandong Computer Science Center (National Supercomputer Center in Jinan), Jinan, China

Abstract. Online social networks provide platforms for people to interact with each other and share moments of their daily life. The online social network data are valuable for both academic and business studies, and are usually processed by anonymization methods before being published to third parties. However, several existing de-anonymization techniques can re-identify the users in anonymized networks. In light of this, we explore the impact of user attributes in social network de-anonymization in this paper. More specifically, we first quantify the significance of attributes in a social network, based on which we propose an attribute-based similarity measure; then we design an algorithm by exploiting attribute-based similarity to de-anonymize social network data; finally we employ a real-world dataset collected from Sina Weibo to conduct experiments, which demonstrate that our design can significantly improve the de-anonymization accuracy compared with a well-known baseline algorithm.

Keywords: Social network · De-anonymizaion · Attribute similarity

1 Introduction

As an innovation of Web 2.0 technology, Online Social Networking (OSN) has been transforming our daily lives. OSN apps such as Facebook, Instagram, and Sina Weibo, provide platforms for individuals and organizations to share their generated original contents such as posts, pictures, and short videos. People who use OSNs enjoy sharing their life and interacting with online friends. Nevertheless, both the contents generated by users and the social relationships among them are valuable for business and academic studies. In order to protect users' privacy, service providers (i.e., data publishers) usually *anonymize* these data before releasing to third parties. Existing anonymization approaches

E. S. Biagioni et al. (Eds.): WASA 2019, LNCS 11604, pp. 458–469, 2019.
https://doi.org/10.1007/978-3-030-23597-0_37

can be generally categorized into six classes, i.e., naive identity removal, edge randomization [8,20], k-anonymity [4,11,21,22], clustering [7,17], differential privacy [3,5,9,19], and random walk [12], which can partially perturb the network structure and attributes, and simultaneously preserve a high level of data utility.

However, there exist a series of structure-based de-anonymization algorithms that are used to re-identify users from anonymized network data [2,14,16,18]. De-anonymization refers to the process of mapping a node from the anonymized graph to a node in the original social network graph (called reference graph). Backstrom *et al.* presented passive attacks and active attacks in [1], in which attackers who are users of the original social network first find their own entities in the anonymized network and then de-anonymize others connected with them (passive attacks), or before publishing the data first fake a number of dummy "Sybil" users and then link them to the victims to create special structures, which can be easily discovered in the anonymized network and used to identify the victims (active attacks). Since active attacks are not scalable, Narayanan *et al.* [14] developed a de-anonymization algorithm based on network structure without involving any dummy "Sybil" users. In [15], a community-enhanced de-anonymization algorithm was proposed, which divides the whole network into smaller communities that can be first mapped (community mapping), then de-anonymizes the users within each community, and finally maps the remaining users in the entire graph. Besides the structural features as mentioned above, user attributes are also employed to improve the accuracy of de-anonymization. In [10], Li *et al.* analyzed the following two major limitations of the structure-based de-anonymization algorithms: structure-based algorithms cannot distinguish two users with similar friends in the anonymized social networks, and a specific user with a few common friends in two social networks can affect the de-anonymization accuracy. To overcome these limitations, the authors introduced an enhanced structure-based de-anonymization scheme leveraging structural transformation similarity in social networks.

Inspired by the structural transformation employed in [10], we notice that the attributes of users could also be used to improve identity de-anonymization in OSNs. Therefore, we examine the attributes of 20 popular OSNs, including 10 OSNs in the USA and 10 OSNs in China, and made the following key observations: (1) the types of attributes obtained during user registration are very similar for all the 20 OSNs; (2) each attribute implies a different amount of discriminatory information for de-anonymization. For example, given a dataset of undergraduate students who are Facebook users from University of Maryland, the attribute *age* may not provide much discriminatory information as *home address* does; and a fine-grained address, such as a detailed postal address, if available, is more valuable for user identification than an address only specifying the state of residency; likewise, the value of attribute *sex*, i.e., female or male, is not as important as sex ratio. Based on the above observations, we propose an attribute-based de-anonymization algorithm in this paper to re-identify users by quantifying the significance values of their attributes and measuring attribute

similarities between user pairs from the reference and the anonymized social network graphs. Our contributions of the paper can be summarized as follows:

- Based on the finding stating that attributes in a social graph are not equally important in de-anonymization, we propose a quantification approach to quantify the *significance value* of each attribute.
- Using the significance values of the attributes, we present an attribute-based de-anonymization algorithm to re-identify user identities in a social graph.
- With the help of a real-world social network dataset collected from the Sina Weibo, we evaluate the performance of our algorithm; and the experimental results show that our proposed attribute-based de-anonymization approach can achieve a better performance compared to a well-known baseline algorithm.

The rest of the paper is organized as follows. In Sect. 2, we present the basic social network model and the corresponding definitions. In Sect. 3, we introduce the existing structure-based de-anonymization methods and the motivation of our study. In Sect. 4, we quantify the significance values of the attributes in a social graph and propose an attribute-based de-anonymization algorithm. Experimental results are reported in Sect. 5. Conclusions and future work are summarized in Sect. 6.

Table 1. Notations

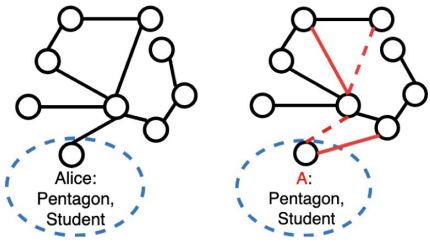
Symbol	Semantics
G	Undirected graph
V	User set
E	Edge set
A	Attribute set
u_i	The ith user in V
A_{u_i}	The attribute set of the ith user
a_k	The kth attribute in A
$v_{a_k}^j$	The jth value of attribute a_k
k_n	The number of attribute values of a_k
$v_{i_{a_k}}$	The value of attribute a_k at user u_i
S_{a_k}	The set of users in V that possess the attribute $a_k \in A$
$S_{a_k}^j$	The set of users possessing the value of $v_{a_k}^j$

2 Social Network Model

Given a social network, we build a corresponding undirected graph $G(V, E, A)$, with vertex (user) set $V = \{u_1, u_2, ..., u_i, ...\}$, edge (friendship relations) set

$E = \{e_{i,j} = (u_i, u_j) | u_i, u_j \in V, i \neq j\}$, and attribute set $A = \{a_1, a_2, ..., a_k, ...\}$. Each user u_i has a set of attributes A_{u_i}, including identity, gender, province, city, etc. We denote a_k as the kth attribute in A, $v_{a_k}^j$ as the jth value of attribute a_k[1], and k_n as the number of attribute values of a_k. Meanwhile, let v_{ia_k} be the value of attribute a_k for user u_i, $S_{a_k}^j$ be the set of users possessing the value of $v_{a_k}^j$, and S_{a_k} be the set of users in V that possess the attribute $a_k \in A$. We summarize the notations and their semantic meanings in Table 1.

Figure 1(a) presents a reference social graph $G_r(V_r, E_r, A_r)$ with real identities and the anonymized social graph $G_a(V_a, E_a, A_a)$ shown in Fig. 1(b) is obtained by perturbing G_r. In V_a, the identities (e.g., name) that can be used to uniquely identify a vertex are replaced with random characters. Other attributes in A_r, such as address and occupation, are preserved for research and business purposes. The edge set E_a is partially modified by adding or deleting edges from E_r where the red dashed lines in Fig. 1(b) are removed edges and the red solid lines are added fake ones. Once adversaries obtain the knowledge of G_a and G_r, they can launch de-anonymization attacks by mapping the users in G_a to those in G_r.



(a) $G_r(V_r, E_r, A_r)$ (b) $G_a(V_a, E_a, A_a)$

Fig. 1. Social network model and anonymization process. (Color figure online)

3 Background and Motivation

Structure-based de-anonymization attacks generally involve two phases. The first phase is called *seed identification*, where a small number of "seed" vertices in both the anonymized network and the reference network are identified and mapped to each other. The second is the *propagation* phase, in which the algorithm randomly selects an unmapped user u_a in V_a and computes the similarity value for each unmapped node u_r in V_r at each iteration. The similarity values are usually calculated according to the topology information such as vertex degree of both networks. Moreover, eccentricity [10,13,14] is generally employed to measure

[1] We assume all attribute values are discrete or categorical for simplicity.

how much a candidate vertex u_r "stands out" from the others, which is defined as

$$ecc(S) = \frac{Max_1(S) - Max_2(S)}{\sigma(S)}, \tag{1}$$

where S is the set of similarity values between u_a and candidate vertices in V_r, $Max_1(\cdot)$ and $Max_2(\cdot)$ are the two highest similarity values in S, and σ denotes the standard deviation of the values in S. User u_a is mapped to u_r if $ecc(S)$ is above some threshold and the similarity value of u_a and u_r equals $Max_1(S)$.

Structure-based de-anonymization attacks rely heavily on the similarity of typologies between the reference graph and the anonymized graph. Existing research considered mainly the topology structure of the graphs, ignoring the rich information carried by the node attributes. In this paper, we intend to exploit node attribute values to facilitate de-anonymization and enhance the de-anonymization success rate for perturbed graphs, which are obtained by adding edges into or removing edges from the corresponding reference graphs. Based on our study, we notice that in a social network dataset, different attributes and different values of the same attribute contribute an unequal amount of discriminatory information to de-anonymization; and the distribution of the attribute values may illustrate a lot more than the value itself. Therefore, we first present a method to quantify the *significance value* of an attribute, which can reflect the importance of attributes in a social network; then we design a mechanism based on the attribute significance value to calculate the similarities among unmapped users u_a and u_r during the process of de-anonymization; finally, we implement the above-mentioned attribute-based similarity measurement in the de-anonymization method to reduce the negative impact of graph perturbation.

4 The Design of Our Algorithm

4.1 Significance Values of Attributes

Intuitively, a widely-distributed attribute with diverse values in a social graph has great importance. This observation motivates the definition of attribute significance values. We first define $\sigma'_{a_k} = J_{a_k}/J^u_{a_k}$, which approximates the distribution information of attribute a_k in the graph of a social network. As shown in (2), J_{a_k} is defined as the ratio of edges between two nodes sharing the same attribute value of a_k to the total number of edges plus that of the nodes possessing the attribute a_k to the total number of nodes in the social graph. In this case, a large J_{a_k} implies that a_k is widely-distributed in the social graph, and accordingly, a large number of users possess this attribute. $J^u_{a_k}$ shown in (3) approximates the fraction of edges and nodes for attribute a_k when the attribute values are placed randomly. A small $J^u_{a_k}$ means that attribute a_k has more diverse attribute values.

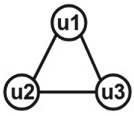
$$J_{a_k} = \frac{|\{e = (u_i, u_j) \in E : (u_i \in S_{a_k}) \wedge (u_j \in S_{a_k}) \wedge (v_{ia_k} = v_{ja_k})\}|}{|E|} + \frac{|S_{a_k}|}{|V|}. \tag{2}$$

$$J^u_{a_k} = \frac{\sum_{j=1}^{k_n} |S^j_{a_k}|^2}{(\sum_{j=1}^{k_n} |S^j_{a_k}|)^2}. \tag{3}$$

With the calculated distribution information σ'_{a_k}, we define the *significance value* σ_{a_k} of an attribute a_k as follows,

$$\sigma_{a_k} = \begin{cases} \sigma'_{a_k}, & \text{if } \sigma'_{a_k} \geq 1, \\ 1, & \text{if } \sigma'_{a_k} < 1. \end{cases} \tag{4}$$

Therefore, σ_{a_k} quantifies how important a_k is in G compared with the scenario where the values of a_k are randomly placed. Note that σ_{a_k} is in the range of $[1, |V|]$, and a larger value indicates that a_k plays a more significant role. For example, let a_k refer to the address attribute, and then if coarse-grained address values of the users are given, such as states or countries, then σ_{a_k} should be close to 1, while when fine-grained addresses are provided, e.g., the detailed postal address, σ_{a_k} should be close to $|V|$.



User	SSN	State
u1	1001	Virginia
u2	1002	Virginia
u3	1003	Maryland

Fig. 2. Example of calculating significance value of attributes.

Next, we present an example to illustrate how to calculate the *significance value* of an attribute in a social graph. As shown in Fig. 2, we assume that there is an undirected social graph with three vertices and three edges. Each user has two attributes: *SSN* (Social Security Number) and *State*, where *SSN* is unique for each user while *State* can be the place where a lot of users live in. Intuitively, *SSN* is more important than *State*. Based on the *significance value* defined above, we can calculate the corresponding value of each attribute as follows,

- For *SSN*: $S_{a_{SSN}} = \{u_1, u_2, u_3\}$; $S^{1001}_{a_{SSN}} = 1$; $S^{1002}_{a_{SSN}} = 1$; $S^{1003}_{a_{SSN}} = 1$. Since three users have different *SSN*, we have $J_{a_{SSN}} = 1$, $J^u_{a_{SSN}} = 1/3$, and $\sigma'_{a_{SSN}} = J_{a_{SSN}}/J^u_{a_{SSN}} = 3 > 1$. Thus, the *significance value* of *SSN* is $\sigma_{a_{SSN}} = 3$.
- For *State*: $S_{a_{State}} = \{u_1, u_2, u_3\}$; $S^{Virginia}_{a_{State}} = 2$; $S^{Maryland}_{a_{State}} = 1$. Since u1 and u2 both live in "Virginia", we can calculate $J_{a_{State}} = 4/3$, $J^u_{a_{State}} = 5/9$, and $\sigma'_{a_{State}} = J_{a_{State}}/J^u_{a_{State}} = 2.4 > 1$. Finally, the *significance value* of *State* is $\sigma_{a_{State}} = 2.4$.

The above calculation results indicate that the attribute *SSN* has a larger significance value than *State* in this example, and SSN is more crucial for identifying users. Next, we leverage the significance values of the attributes in a social graph to design an attribute-based similarity measurement, which will be further exploited to improve the performance of social graph de-anonymization.

4.2 Attribute-Based Similarity Measurement

In this section, we employ the significance values of attributes proposed in the above subsection to measure the similarity value between two vertices. The basic idea of is straightforward: for any pair of users, the more number of same attribute values they have, the more similar they are; moreover, the higher the significance values of the attributes between two users, the higher the similarity.

More specifically, given $G_a(V_a, E_a, A_a)$ and $G_r(V_r, E_r, A_r)$ as the anonymized graph and the reference one with true user identities, respectively, we can compute the *significance value* σ_{a_k} for each attribute a_k based on the information provided by G_r. For a vertex u_a in V_a and a vertex u_r in V_r, let $\{a'_1, a'_2, ..., a'_n\}$ be the set of their common attributes; then their attribute-based similarity value $d_{(u_a,u_r)}$ can be calculated as follows: we initially set $d_{(u_a,u_r)} = 1$; then for each common attribute a'_k, if two vertices have the same attribute value (i.e., $v_{ia'_k} = v_{ja'_k}$), we define $d_{(u_a,u_r)} = d_{(u_a,u_r)} \times \sigma_{a'_k}$; otherwise, we set $d_{(u_a,u_r)} = d_{(u_a,u_r)} \times \frac{1}{\sigma_{a'_k}}$. Note that the term $\frac{1}{\sigma_{a'_k}}$ indicates that two nodes that have the same attribute but the corresponding attribute values are different have small similarity values.

4.3 Design of De-anonymization Algorithm

Based on the concepts presented in the previous subsections, we propose a de-anonymization algorithm for a social graph where each vertex has several attributes. Our algorithm takes advantage of the existing structure-based algorithms with two phases: *seed identification* and *propagation*.

Seed Identification. In this phase, we identify a few "seed" pairs of users (u_a, u_r) between G_a and G_r. To achieve this goal, several seeding methods such as k-clique, matching top nodes, and random selection discussed in [6], can be employed. The next propagation phase starts from these "seed" pairs, and newly mapped user pairs will become new "seeds".

Propagation. In this phase, starting from the identified "seeds", the de-anonymization algorithm iteratively maps each unmapped vertex u_a in G_a to an unmapped vertex u_r in G_r. We use $S = \{(u, u') | u \in V_a, u' \in V_r\}$ to denote the already mapped "seed" pairs, and calculate significance values $\{\sigma_{a_k}\}$ for all attributes in A_r of the reference graph $G_r(V_r, E_r, A_r)$. At each iteration, we pick an arbitrary unmapped user u_a who has a successfully mapped neighbor from anonymized social graph $G_a(V_a, E_a, A_a)$, and calculate its similarity values with all unmapped nodes in V_r who possess at least one successfully mapped neighbor. This process is detailed in Algorihtm 1 whose inputs include G_a, G_r, significance values $\{\sigma_{a_k}\}$, u_a, and "seed" pairs S. We first initialize V_{map} to be the collection of seed vertices in V_r and D to be the empty set. Then we pick up a u_r who is not in V_r but has a seed neighbor in V_r (Lines 3 and 4), and calculate the attribute-based similarity value between u_a and u_r (Lines 6–13)

based on the similarity parameter $d_{(u_a,u_r)}$ defined in Sect. 4.2. Finally we select the u_r (i.e., the u'_r in Algorithm 1) based on the eccentricity measure defined in (1) as the mapping of u_a. This newly mapped vertex pair (u_a, u'_r) is inserted into S as a new seed pair for the next round of iteration (the next execution of Algorithm 1).

Algorithm 1. Attribute-based Similarity Calculation

Input : Two social graphs: $G_a(V_a, E_a, A_a)$ and $G_r(V_r, E_r, A_r)$
Unmapped users: $u_a \in V_a$
Significance values of attributes: σ_{a_k}
Seed pairs: $S = \{(u, u')|u \in V_a \text{ and } u' \in V_r\}$
Output: the newly mapped pair (u_a, u'_r) with $u'_r \in V_r$

Set $V_{map} = \{u'|u' \in V_r \text{ and } \exists(u, u') \in S\}$
set $D = \emptyset$
for *each* $u' \in V_{map}$ *and* $(u', u_r) \in E_r$ **do**
  if $u_r \notin V_{map}$ **then**
    Set attribute-based similarity value: $d_{(u_a,u_r)} = 1$
    for *each common attribute* $a'_k \in A_{u_a} \cap A_{u_r}$ **do**
      // $v_{a'_k}(\cdot)$: attribute value of a'_k
      if $v_{a'_k}(u_a) = v_{a'_k}(u_r)$ **then**
        $d_{(u_a,u_r)} = d_{(u_a,u_r)} \times \sigma_{a'_k}$
      else
        $d_{(u_a,u_r)} = d_{(u_a,u_r)} \times \frac{1}{\sigma_{a'_k}}$
      end
    end
    $D = D \cup d_{(u_a,u_r)}$
  end
end
Calculate $ecc(D)$ based on (1) and select the node u'_r with the highest $d_{(u_a,u'_r)}$ if $ecc(D)$ is above a threshold
Set $S = S \cup (u_a, u'_r)$
return (u_a, u'_r)

5 Experiments

In this section, we use a dataset collected from Sina Weibo, a popular social media in China, to evaluate the performance of the proposed algorithm. The Weibo social network is based on "following" relationships among users. Our dataset includes 5663 nodes and 10000 edges. We convert this dataset into an undirected graph/network with an average degree of 3.5317. Each vertex (user) in the network has three attributes: province, city, and gender. We duplicate the network to get an anonymized version and apply Random Add/Del or Random Switch, which are edge randomization methods proposed in [20], to add noise. For comparison, we also implement the most influential de-anonymization method

proposed in [14] to obtain experimental results that can serve as the baseline. In the seed identification phase, we randomly select nodes from 3-cliques in the reference graph and employ the corresponding nodes in the anonymized network as the matching seeds. In other words, there is no error in seed mapping. This consideration makes us focus on the performance of the propagation phase of the de-anonymization algorithm.

The performance metric is the de-anonymization accuracy, which is the ratio of the number of nodes correctly de-anonymized over the total number of nodes in the whole network. The reported results in the following subsections are the average of 50 trials.

5.1 Impact of the Number of Seeds

At first, we evaluate the impact of the number of seeds on de-anonymization accuracy. In this experiment, no noise is added, the eccentricity threshold is set to 0.1, and the number of seeds varies from 10 to 90. As shown in Fig. 3, when the number of seeds is increased from 10 to 90, the percentage of vertices correctly de-anonymized is increased, and our algorithm can correctly de-anonymize more vertices than the baseline algorithm. More particularly, when the number of seeds is 10, the results of baseline algorithm and our algorithm are respectively 0.0219 and 0.5890; when the number of seeds is increased to 90, the results are increased to 0.1636 and 0.7864, respectively.

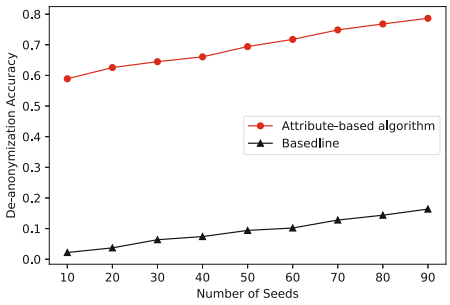


Fig. 3. Impact of the number of seeds on the de-annoynimization accuracy.

5.2 Impact of Noise

Next, we add noises into the anonymized network with two edge randomization methods. The first one is Random Add/Del, which randomly removes certain number of edges from a network and then randomly adds the same amount of edges back into the network. The second one is called Random Switch, which removes two randomly chosen edges $e_{m,n}$ and $e_{a,b}$ from a network and creates two new edges $e_{m,a}$ and $e_{n,b}$ back into the network. The *noise ratio* is defined

to be the ratio of the number of newly added and deleted edges over the total number of edges in the network. In this evaluation, the noise ratio is increased from 0 to 0.3 at an interval of 0.05, the number of seeds is fixed to 90, and the eccentricity threshold is set to 0.1.

Figure 4 presents the accuracy of de-anonymization in the network processed by Random Add/Del. Since adding or removing edges changes the degrees of the vertices in the anonymized network, both de-anonymization algorithms are negatively affected, compared to the case when no noise is added. Nevertheless, our algorithm has better performance than the baseline. More specifically, when the noise ratio increases from 0 to 0.3, the result of our algorithm decreases from 0.7864 to 0.5285; while that of the baseline algorithm decreases from 0.1636 to 0.0340.

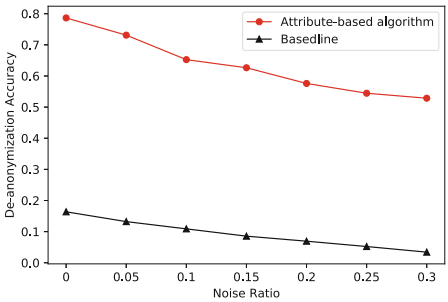


Fig. 4. Impact of noise on the de-annoynimization accuracy in the Random Add/Del annonymized graph.

Figure 5 demonstrates the result of de-anonymization in the network processed by Random Switch. From the result we can see that Random Switch has a little impact on the baseline algorithm but noticeably impacts on our algorithm. This is because the similarity calculation of the baseline algorithm is based on the local structure similarity between two nodes, while that of our algorithm is based on the attribute similarity between two vertices. Random Switch does not change the degree of vertices, but it disturbs the structure of the network, and more severely affects the attribute similarity between two vertices. As shown in Fig. 5, when the noise ratio is increased from 0 to 0.3, the result of our algorithm drops from 0.7864 to 0.3880, while that of the baseline algorithm drops from 0.1636 to 0.1399. Despite this, our algorithm is still more efficient than the baseline one.

6 Conclusion and Future Work

In this paper, we investigate the impact of attributes on social network de-anonymization and propose a method to quantify the significance values of the

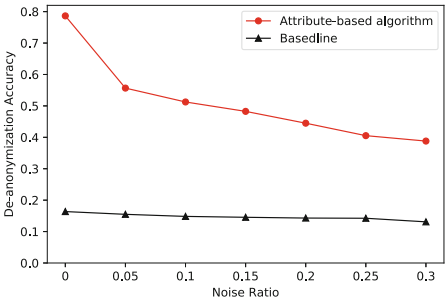


Fig. 5. Impact of noise on the de-annoynimization accuracy in the Random Switch annonymized graph.

attributes in social networks. Based on the significance values, we design an algorithm to de-anonymize social networks based on the attribute similarity of users. Finally, we evaluate our algorithm on a Sina Weibo dataset processed by two edge randomization methods. The experimental results indicate that the proposed algorithm can achieve a much higher de-anonymization accuracy compared to the selected baseline algorithm. For future research, we can refine the definition of the significance value by considering different value scopes, such as a continuous space. Moreover, we can further modify the similarity definition of two nodes by considering both attribute-based similarity and local structure similarity.

Acknowledgment. This work was partially supported by the US National Science Foundation under grants CNS-1704397, CNS-1704287, and CNS-1704274, and the National Science Foundation of China under grants 61832012, 61771289, and 61672321.

References

1. Backstrom, L., Dwork, C., Kleinberg, J.: Wherefore art thou R3579x?: anonymized social networks, hidden patterns, and structural steganography. In: Proceedings of the 16th International Conference on World Wide Web, pp. 181–190. ACM (2007)
2. Cai, Z., He, Z., Guan, X., Li, Y.: Collective data-sanitization for preventing sensitive information inference attacks in social networks. IEEE Trans. Dependable Secure Comput. **15**(4), 577–590 (2018)
3. Chen, S., Zhou, S.: Recursive mechanism: towards node differential privacy and unrestricted joins. In: Proceedings of the 2013 ACM SIGMOD International Conference on Management of Data, pp. 653–664. ACM (2013)
4. Cheng, J., Fu, A.W.C., Liu, J.: K-isomorphism: privacy preserving network publication against structural attacks. In: Proceedings of the 2010 ACM SIGMOD International Conference on Management of Data, pp. 459–470. ACM (2010)
5. Day, W.Y., Li, N., Lyu, M.: Publishing graph degree distribution with node differential privacy. In: Proceedings of the 2016 International Conference on Management of Data, pp. 123–138. ACM (2016)

6. Gulyás, G.G., Imre, S.: Measuring importance of seeding for structural de-anonymization attacks in social networks. In: 2014 IEEE International Conference on Pervasive Computing and Communication Workshops (PERCOM WORKSHOPS), pp. 610–615. IEEE (2014)
7. Hay, M., Miklau, G., Jensen, D., Towsley, D., Weis, P.: Resisting structural re-identification in anonymized social networks. Proc. VLDB Endow. **1**(1), 102–114 (2008)
8. He, Z., Cai, Z., Yu, J.: Latent-data privacy preserving with customized data utility for social network data. IEEE Trans. Veh. Technol. **67**(1), 665–673 (2018)
9. Li, C., Hay, M., Miklau, G., Wang, Y.: A data-and workload-aware algorithm for range queries under differential privacy. Proc. VLDB Endow. **7**(5), 341–352 (2014)
10. Li, H., Zhang, C., He, Y., Cheng, X., Liu, Y., Sun, L.: An enhanced structure-based de-anonymization of online social networks. In: Yang, Q., Yu, W., Challal, Y. (eds.) WASA 2016. LNCS, vol. 9798, pp. 331–342. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-42836-9_30
11. Liu, K., Terzi, E.: Towards identity anonymization on graphs. In: Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data, pp. 93–106. ACM (2008)
12. Liu, Y., Ji, S., Mittal, P.: SmartWalk: enhancing social network security via adaptive random walks. In: Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, pp. 492–503. ACM (2016)
13. Narayanan, A., Shmatikov, V.: Robust de-anonymization of large sparse datasets. In: 2008 IEEE Symposium on Security and Privacy, pp. 111–125. IEEE (2008)
14. Narayanan, A., Shmatikov, V.: De-anonymizing social networks. arXiv preprint arXiv:0903.3276 (2009)
15. Nilizadeh, S., Kapadia, A., Ahn, Y.Y.: Community-enhanced de-anonymization of online social networks. In: Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security, pp. 537–548. ACM (2014)
16. Qian, J., Li, X.Y., Zhang, C., Chen, L., Jung, T., Han, J.: Social network de-anonymization and privacy inference with knowledge graph model. IEEE Trans. Dependable Secure Comput. (2017)
17. Thompson, B., Yao, D.: The union-split algorithm and cluster-based anonymization of social networks. In: Proceedings of the 4th International Symposium on Information, Computer, and Communications Security, pp. 218–227. ACM (2009)
18. Tian, W., Mao, J., Jiang, J., He, Z., Zhou, Z., Liu, J.: Deeply understanding structure-based social network de-anonymization. Procedia Comput. Sci. **129**, 52–58 (2018)
19. Wang, Q., Zhang, Y., Lu, X., Wang, Z., Qin, Z., Ren, K.: Real-time and spatio-temporal crowd-sourced social network data publishing with differential privacy. IEEE Trans. Dependable Secure Comput. **15**(4), 591–606 (2018)
20. Ying, X., Wu, X.: Randomizing social networks: a spectrum preserving approach. In: Proceedings of the 2008 SIAM International Conference on Data Mining, pp. 739–750. SIAM (2008)
21. Zhou, B., Pei, J.: Preserving privacy in social networks against neighborhood attacks (2008)
22. Zou, L., Chen, L., Özsu, M.T.: K-automorphism: a general framework for privacy preserving network publication. Proc. VLDB Endow. **2**(1), 946–957 (2009)