Fan Chung Graham · Megan Dewar ·
Bogumił Kamiński · Paweł Prałat ·
François Théberge (Eds.)

LNCS 16630

Modelling and Mining Networks

21st International Workshop, WAW 2026
Toronto, ON, Canada, June 15–19, 2026
Proceedings

Springer

The Anonymization Problem in Social Networks

R. G. de Jong[1](✉), M. P. J. van der Loo[1,2], and F. W. Takes[1]

[1] LIACS, Leiden University, Leiden, The Netherlands
r.g.de.jong@liacs.leidenuniv.nl
[2] CBS, Statistics Netherlands, Den Haag, The Netherlands

Abstract. This paper introduces a unified computational framework for the anonymization problem in social networks, where the objective is to maximize node anonymity through graph alterations. We define three variants of the underlying optimization problem: full, partial and budgeted anonymization. In each variant, the objective is to maximize the number of k-anonymous nodes, i.e., nodes for which at least $k-1$ other nodes are equivalent under a particular anonymity measure. We propose four new heuristic network anonymization algorithms and implement these in ANO-NET, a reusable computational framework. Experiments on three common graph models and 19 real-world network datasets yield three empirical findings. First, regarding the method of alteration, experiments on graph models show that random edge deletion is more effective than edge rewiring and addition. Second, we show that the choice of anonymity measure strongly affects both initial network anonymity and the difficulty of anonymization. This highlights the importance of careful measure selection, matching a realistic attacker scenario. Third, comparing the four proposed algorithms and an edge sampling baseline from the literature, we find that an approach which preferentially deletes edges affecting structurally unique nodes, consistently outperforms heuristics based solely on network structure. Overall, our best performing algorithm retains on average 14 times more edges in full anonymization. Moreover, it yields 4.8 times more anonymous nodes than the baseline in the budgeted variant. On top of that, the best performing algorithm achieves a better trade-off between anonymity and data utility. This work provides a foundation for the future development of effective network anonymization algorithms.

Keywords: Complex networks · k-Anonymity · Privacy · Optimization

1 Introduction

In social network analysis, also known as network science, interactions between people are often the object of study for various applications and downstream tasks, including influence maximization [5] and understanding epidemic

F. C. Graham et al. (Eds.): WAW 2026, LNCS 16630, pp. 18–33, 2026.
https://doi.org/10.1007/978-3-032-27193-8_2

spread [2]. To conduct such research, it is desirable to have large real-world networks of people that are as realistic as possible. However, the identity of individuals in these networks can be revealed based on structural network information, even after removing unique identifiers, i.e., after pseudonymization [12,21]. As a result, sharing or publishing social network data can lead to a breach in the privacy of the people represented. To solve this problem, sometimes also referred to as identity anonymization [18], various works introduced methods to share or publish networks in a privacy aware manner, including clustering [4], differential privacy [10] and k-anonymity [9,17]. Each of these approaches anonymizes the data in a different way. In this work, we focus on k-anonymity methods, which, contrary to approaches mentioned above, allow the creation of an altered anonymized version of the original full network dataset, retaining all individuals.

In k-anonymity, the anonymity of a network as a whole is determined as the fraction of anonymous nodes. A node is k-anonymous if it has, according to a particular *anonymity measure*, the same *signature* as at least $k - 1$ other nodes. For example, a node is k-anonymous with respect to the measure of degree if it has the same degree value as $k - 1$ other nodes. The choice of measure essentially determines the attacker scenario against which one protects [13].

Much previous work focuses on $k = 2$, measuring network anonymity as uniqueness: the fraction of nodes with a unique signature [12,21]. Given a network and an anonymity measure, the aim of *network anonymization* is then to decrease the uniqueness to a desired level by means of graph alterations, while ensuring that the resulting network data remain suitable for its intended purposes. Consequently, network anonymization involves a trade-off between anonymity and *data utility*. It is a known fact that altering fewer edges leads to higher utility [3]. Utility can also be measured in terms of how well structural network properties are preserved, or as the extent to which performance in network analysis tasks such as community detection is retained after anonymization.

The time complexity of anonymizing a network with as few alterations as possible depends on the chosen anonymity measure. While for degree, anonymization can be done in quadratic time [18], the problem is NP-hard when the exact structure of the 1-neighborhood and node labels are considered [27]. Furthermore, the objective function in the anonymization problem is neither monotonic nor submodular, elegant properties that enabled the development of efficient methods for other common network analysis problems such as influence maximization [5]. Therefore, existing work on the anonymization of networks focuses on (meta)heuristic algorithms, yet always for either a specific anonymity measure [17,27], tailored to a specific type of network [19], or with a focus on random alterations [9,21].

In this paper, we unite these lines of research and introduce a general formulation of the anonymization problem in the form of three meaningful variants, being 1) full, 2) partial and 3) budgeted anonymization. In addition, we consider four commonly used anonymity measures from the literature, providing insights into how the choice of measure affects the anonymization process. For the latter task, we propose four new measure-agnostic heuristic anonymization algorithms.

By means of experiments on both graph models and real-world network data, we independently investigate the effect of A) the chosen alteration operation, B) the anonymity measure, C) the anonymization algorithm and D) the effect of the problem variants on the attained utility. Together with this paper, we release ANO-NET[1], a computational framework for anonymizing networks that enables researchers to implement and evaluate anonymization algorithms across different anonymity measures and different variants of the problem. The framework is accompanied by a representative real-world benchmark dataset.

The remainder of this paper is structured as follows. First, Sect. 2 reviews related work. Section 3 covers preliminaries on networks and anonymization. In Sect. 4, we formally define the anonymization problem and its variants. Section 5 presents our framework for the anonymization process and the proposed heuristic algorithms. We discuss experiments and results in Sect. 6. Finally, Sect. 7 concludes the paper and outlines directions for future work.

2 Related Work

Below, we discuss the most common measures of anonymity and their commonly used algorithms for anonymization (for more details, see for example [17,27]).

Degree. The simplest anonymity measure is the node degree, for which exact algorithms minimizing the number of alterations [17] have been devised.

Neighborhood. Node anonymity measures based on the ego network or d-neighborhood capture more structural information [1,11,21,27]. In node labeled networks, anonymization under this measure is NP-hard [27], making exact approaches infeasible. Consequently, a greedy algorithm that modifies node neighborhoods to resemble those of similar nodes [27], a genetic algorithm [1] and an edge sampling method [21] were proposed.

Graph Invariants. The work of [13] introduced measures based on graph invariants: the number of nodes and edges in the d-neighborhood of a considered node, and the degree distribution within the d-neighborhood.

VRQ. Vertex Refinement Queries (VRQ) [9] consider the multiset of node degrees within distance d of a given node as the measure of anonymity. Anonymization algorithms include random rewiring and algorithms that find similar neighborhoods [25].

In this paper, we contribute to the existing literature by formally introducing and exploring three meaningful variants of the anonymization problem. Instead of proposing algorithms tailored to a specific measure (i.e., a specific attacker scenario), we present a general framework applicable to all variants, and propose several measure-agnostic heuristic anonymization algorithms. Crucially, we empirically examine the impact of different anonymity measures on the anonymization process, and analyze the trade-off between anonymity and utility.

[1] https://github.com/RacheldeJong/ANONET.

3 Preliminaries

In this section we formally introduce concepts used throughout this paper, including notions of networks, anonymity and the graph alteration operations.

3.1 Networks

We define a network as an undirected graph $G = (V, E)$ containing a set of nodes V and a set of edges $\{v, w\} \in E$ between nodes $v, w \in V$. The degree of a node is the number of connections it has: $degree(v) = |\{w : \{v, w\} \in E\}|$. The clustering coefficient of a node v equals the number of triangles it is part of, divided by the maximum number of triangles it could form. The distance between two nodes, denoted $dist(v, w)$, is the minimal number of edges that needs to be traversed to move from node v to node w. It follows that $dist(v, v) = 0$ and if no path exists, i.e., if the nodes belong to different components, we let $dist(v, w) = \infty$. The component with the largest number of nodes is called the largest connected component (LCC) or giant component.

The d-neighborhood of a node, $N_d(v) = (V_{N_d(v)}, E_{N_d(v)})$, is the subgraph consisting of all nodes that are at most at distance d of node v, and all edges between them. Two d-neighborhoods are structurally indistinguishable if they are *isomorphic*. We can determine whether two d-neighborhoods are isomorphic by comparing their *canonical labeling* [20], a label assigned by a function $\mathcal{C}(G)$ such that two graphs have the same label value only if they are isomorphic.

Networks often contain communities, groups of nodes more densely connected internally than with the rest of the network. Community detection algorithms [26] assign nodes to communities by favoring intra-community connections and limiting inter-community connections. Some nodes in a network may have more central positions than others. Centrality measures can be used to rank nodes based on their position in the network [2]. The commonly used measure of betweenness centrality assigns a centrality score based on the fraction of shortest paths that pass through the node.

3.2 Anonymity in Networks

Anonymity Definitions. We say that two nodes $v, w \in V$ are equivalent according to anonymity measure M if they have the same signature: $M(v) = M(w)$. We then say that these nodes belong to the same *equivalence class*, denoted $eq_M(v)$ for a node v. A node is k-anonymous if $|eq_M(v)| \geq k$. We let P_M denote the partition of node set V into equivalence classes using measure M. When $k = 2$, the anonymity of a graph can be summarized using *uniqueness*, defined as the fraction of unique nodes in the network with respect to measure M, as shown in Eq. 3. Given this paper's focus on $k = 2$, in this context network anonymity is simply $1 - U(G)$. For later use, Eq. 2 also defines the set of *unique edges* as the edges incident to at least one unique node.

$$V_u = \{v \in V : |eq_M(v)| = 1\} \tag{1}$$

$$E_u = \{\{v, w\} \in E : v \in V_u \vee w \in V_u\} \tag{2}$$

$$U(G) = |V_u|/|V| \tag{3}$$

Alteration Operations. We consider three graph alteration operations:

– **Deletion**: $E' \leftarrow E \setminus \{\{v, w\}\},\ s.t.\ \{v, w\} \in E$.
– **Addition**: $E' \leftarrow E \cup \{\{v, w\}\},\ s.t.\ \{v, w\} \notin E$.
– **Rewiring**: $E' \leftarrow E \setminus \{\{v, w\}, \{v', w'\}\} \cup \{\{v, w'\}, \{v', w\}\}$ s.t. $\{v, w\}, \{v', w'\} \in E$ and $\{v, w'\}, \{v', w\} \notin E$.

As preliminary experiments in Sect. 6.2 show that edge deletion is the most effective operation, we focus on edge deletion in the remainder of the text.

Measures for *k*-Anonymity. Various anonymity measures for k-anonymity have been introduced in the literature, differing both in the structural properties they capture and how far they reach. In this paper, we use the measures listed in Table 1 selecting one representative measure from each category of approaches from previous work covered in Sect. 2. All measures are parameterized by distance d, indicating up to which distance structural information is considered.

Table 1. Anonymity measures (left column), signature for a given node v and distance d (middle column), and node set affected when altering edge $\{v, w\}$ (right column).

Measure	$M(v, d)$	$A_M(\{v, w\})$
DEGREE [17,18]	$degree(v)$	$\{v, w\}$
COUNT [11]	$(\lvert V_{N_d(v)} \rvert, \lvert E_{N_d(v)} \rvert)$	$V_{N_d(v)} \cap V_{N_d(w)}$
d-k-ANONYMITY [21,27]	$\mathcal{C}(N_d(v))$	$V_{N_d(v)} \cap V_{N_d(w)}$
VRQ [9,25]	$\{degree(u) : u \in V,\ dist(u, v) = d\}$	$V_{N_d(v)} \cup V_{N_d(w)}$

To determine its anonymity, each node is assigned a *signature* based on the chosen measure, denoted by $M(v, d)$ in Table 1. When $d \geq 2$, signatures for all $d \geq 1$ up to d should be equal for node equivalence. After an edge is altered, the signature of a subset of the graph's node set changes. Which nodes are affected depends on the *reach* of the anonymity measure, i.e., how far structural information is considered to compute the node signature [13]. The rightmost column in Table 1 lists the set of affected nodes $A_M(\{v, w\})$ after deleting edge $\{v, w\}$, for each measure. This is used later to avoid redundant uniqueness recomputations.

4 The Network Anonymization Problem

We now define the anonymization problem and its three variants in Definition 1.

Definition 1 (Network anonymization problem). *Given a network G, anonymity measure M with range d, a value of k and a set of allowed alteration operations:*

1. ***Full anonymization**: Make all nodes k-anonymous with as few alterations to the network as possible.*
2. ***Partial anonymization**: Ensure that a fraction α (with $0.0 < \alpha < 1.0$) of the nodes is k-anonymous using as few alterations to the network as possible.*
3. ***Budgeted anonymization**: Perform at most B (with $0 < B < |E|$) alterations to the network while maximizing the number of k-anonymous nodes.*

Most existing works focus on full anonymization, aiming to anonymize all nodes in the network. However, some nodes, for example with a high degree as commonly present in real-world data, may require substantially more alterations, motivating the partial variant which generalizes the subset variant proposed in [6]. Since data utility is an important aspect of the problem [17], the budgeted variant limits the number of allowed alterations.

In this paper, we focus on $k = 2$ and $d = 1$, as prior work shows that increasing k (up to 5) has limited effect [12], while $d > 1$ is less realistic and substantially reduces anonymity [13]. Further exploration is left for future work.

5 Approach

In this section we present the general framework for anonymization and introduce the five algorithms (four new, one baseline) used in the remainder of this paper.

5.1 Anonymization Algorithm

The framework outlined in Algorithm 1 can be used for all variants of the anonymization problem introduced in Sect. 4. It takes as input a budget and target anonymity, i.e., the desired number of nodes that is at least k-anonymous. For the full and partial variant, the budget equals $|E|$ and the target anonymity T should be set to $|V|$ and $\alpha * |V|$ respectively, where α equals the fraction of nodes that should be anonymous for the partial variant of the problem. For the budgeted variant, one should give the budget B and a target anonymity of $|V|$. The other inputs required are the original graph G, the measure M and value k, cf. Definition 1. The final two parameters are the chosen anonymization algorithm (see Sect. 5.3), and recompute gap R, which determines after how many edge alterations P_M must be recomputed, further discussed in Sect. 5.2.

The algorithm works as follows. In each iteration of the while loop in lines 3 to 12, B' edges are altered, as selected on line 5 by the chosen anonymization algorithm. The network is altered (line 6) and the affected part of the partition (as defined in the third column of Table 1) is updated in line 7. The remainder of the lines are bookkeeping operations to ensure the right number of edges B', equal to R or the remaining budget if $B < R$, is deleted each iteration and the best result, G_{best}, is updated. The *anonymity()* function returns how many nodes are k-anonymous in the altered graph. When the while loop terminates, either budget B is depleted or target anonymity T is reached. Finally, line 13 returns the graph with the highest anonymity found.

5.2 Recompute Gap

Altering edges changes the network structure, which in turn influences the node signatures, and thus anonymity. After each alteration to the graph, updating the corresponding equivalence classes and data structures is computationally expensive. Therefore, we incorporate a trade-off between accuracy and computational cost by updating the graph and recomputing only after R alterations. With recompute gap $R = 1$ the graph and partition are updated after each alteration, i.e., no recompute gap. For $R = B/100$ the values are updated 100 times.

5.3 Anonymization Algorithms

The heuristic anonymization algorithms described below each assign a probability to edges to be deleted based on specific criteria. We focus on edge deletion as initial experiments in Sect. 6.2 showed that this yielded the best results. This is to be expected, as removing edges reduces the size of node neighborhoods. Smaller neighborhoods admit fewer possible network structures, making these less likely to be unique. The effect of edge deletion is largest on the two incident nodes, which lose one neighbor and all triangles involving that edge. Other affected nodes lose a single edge from their neighborhood. Our structure based heuristic algorithms defined below, leverage these effects. The five proposed heuristic algorithms fall into three categories: random edge sampling; our baseline, two structure based heuristics, and two uniqueness based heuristics.

Random Edge Sampling. The baseline algorithm, ES [21], randomly selects which edges to delete by assigning the same probability to each edge.

$$P_{\text{ES}}(\{v, w\}) = 1/|E| \tag{4}$$

Algorithm 1. ANONYMIZATION

Input: graph G**, measure** M**, value of** k**, anonymization algorithm** $Anonalg$**, budget** B**, target anonymity** T**, recompute gap** R

$G' \leftarrow G$, $G_{best} \leftarrow G$, $P_M \leftarrow compute\_P_M(G, M)$, $P_{Mbest} \leftarrow P_M$

while $B > 0$ **and** $anonymity(G', P_M, k) < T$ **do**

  $B' \leftarrow min(R, B)$

  $E_A \leftarrow Anonalg(G', P_M, M, k, B')$ ▷ Select edges to alter (Section 5.3)

  $G' \leftarrow update\_graph(G', E_A)$

  $P_M \leftarrow update\_P_M(G', P_M, M, E_A)$

  $B \leftarrow B - B'$

  if $anonymity(G', P_M, k) > anonymity(G_{best}, P_{Mbest}, k)$ **then**

    $G_{best} \leftarrow G'$, $P_{Mbest} \leftarrow P_M$

  end if

end while

Return: G_{best}

Structure Based. The DEGREE heuristic aims to affect more nodes by assigning higher probabilities to edges connecting two high degree nodes. These edges are likely part of many triangles.

$$P_{\text{DEGREE}}(\{v, w\}) = \frac{min(degree(v), degree(w))}{\sum_{\{v',w'\} \in E} min(degree(v'), degree(w'))} \quad (5)$$

The AFF heuristic computes the exact number of nodes affected by deleting an edge as defined in the rightmost column of Table 1.

$$P_{\text{AFF}}(\{v, w\}) = \frac{|A_M(\{v, w\})|}{\sum_{\{v',w'\} \in E} |A_M(\{v', w'\})|} \quad (6)$$

Uniqueness Based. Since primarily unique nodes need to be altered in order to increase anonymity, this category of algorithms aims to have a larger impact on unique nodes. The first uniqueness-based algorithm, UNIQUE, explicitly targets unique edges, i.e., edges incident to at least one unique node (see Eq. 2). From this set, the desired number of edges B' is chosen at random. If the set of unique edges is insufficiently large, the remaining edges are selected at random from the remaining edge set.

$$P_{\text{UNIQUE}}(\{v, w\}) = \begin{cases} 1/|E_u| & \text{if } \{v, w\} \in E_u,\ |E_u| > B' \\ 1 & \text{if } \{v, w\} \in E_u,\ |E_u| \leq B' \\ 1/|E \setminus E_u| & \text{if } \{v, w\} \notin E_u,\ |E_u| < B' \\ 0 & \text{otherwise} \end{cases} \quad (7)$$

The second uniqueness based algorithm, Unique Affected (UA), prioritizes edges whose alteration affects a larger number of unique nodes. To ensure no edge is assigned a selection probability of zero, a factor of $\frac{1}{|E|}$ is added. Similar to the AFF heuristic, this heuristic is more expensive to compute when using the COUNT anonymity measure considered in this paper, as the number of affected nodes need to be computed for each edge.

$$P_{\text{UA}}(\{v, w\}) = \frac{|A_M(\{v, w\}) \cap V_u| + 1/|E|}{\sum_{\{v',w'\} \in E} (|A_M(\{v', w'\}) \cap V_u| + 1/|E|)} \quad (8)$$

5.4 Utility

In this work, we choose to measure data utility based on the change in topological network properties and performance on common network analysis tasks:

- **Clustering coefficient.** The average clustering coefficient over all nodes. This value will likely decrease as triangles are destroyed by deleting edges.
- **Average distance.** The average shortest path length $dist(v, w)$ over all pairs of nodes in the same component. As edges are deleted, paths are destroyed, hence the average shortest path length increases. However, when the network starts to split into components, this value can decrease.

- **Largest connected component (LCC).** The fraction of nodes in the giant component. This is expected to decrease as edges are deleted.
- **Centrality.** Overlap in top 100 most central nodes according to betweenness centrality before and after anonymization. If the top 100 nodes did not change the value equals 1.0, whereas it is 0.0 if there are no common nodes.
- **Community structure.** Normalized Mutual Information (NMI) [15] between the communities found by the Leiden algorithm [26], before and after anonymization. To account for non-determinism of the community detection algorithm, we report on $NMI_{utility} = \max(0.0, NMI_{stab} - NMI_{anon})$. Here NMI_{stab} equals the average NMI comparing the community assignments C of the original network, and NMI_{anon} the average NMI between community assignments in the network before (C) and after (C') anonymization; a value of 1.0 implies community structure is preserved, 0.0 that it is completely lost.

6 Results

In this section we first summarize the experimental setup and network datasets. Then, two initial experiments compare the effectiveness of different alteration operations on graph models, and the effect of different anonymity measures on real-world network data. Finally, we compare the proposed anonymization algorithms based on their effectiveness and achieved utility on the three variants.

6.1 Experimental Setup and Data

We implement the algorithms introduced in Sect. 5 in our reusable C++ ANONET framework.[2] Experiments on graph models use Python and NetworkX [8], utility analysis is performed with igraph [7]. Experiments are conducted on both graph models and real-world networks. We use three graph models: Erdős-Rnyi (ER), where edges are added at random; Barabási-Albert (BA), accounting for the preferential attachment mechanism that generates a powerlaw degree distribution, and Watts-Strogatz (WS) with rewire probability 0.05, which generates clustered networks [2]. Each model uses graphs with 500 nodes and average degrees 2, 4, 16 and 64. Properties of the real-world network datasets used can be found in Table 2.

In each experiment, edges are sequentially deleted until the desired anonymity level is reached, or the runtime exceeds 30 min. In the latter case, anonymization continues until a budget of $B = 5\%$ of the edges is deleted, corresponding to the budgeted variant. In partial anonymization, we set target anonymity α to 95% of the nodes. The recompute gap is set to delete 1% of the edges in each iteration. To account for nondeterminism in the algorithms and graph models, we report averages and show error bars indicating standard deviations over five runs and five generated graph instances.

For utility analysis, we generate anonymized graphs for the 19 networks that finished within the time limit by deleting the selected edges and compute the utility metrics summarized in Sect. 5.4. For community detection, we run the Leiden algorithm [26] 10 times for each original and anonymized network.

[2] https://github.com/RacheldeJong/ANONET.

Table 2. Real-world networks used, listing the number of nodes, edges, average degree, average clustering coefficient, fraction of nodes in the largest connected component (LCC) and average distance. Results include the initial uniqueness using COUNT; values below 0.05 (partially anonymous networks) are shown in italics. The improvement ratio for the three variants is defined as the uniqueness achieved by ES divided by that achieved by UA. Results not included in Sect. 6.4 are indicated by a dash.

	$\|V\|$	$\|E\|$	Avg. deg.	Clust. coeff	Frac. LCC	Avg. dist.	$U(G)$	Improvement UA vs. ES 1. Full	2. Partial	3. Budgeted
Radoslaw emails [14]	167	3,250	38.92	0.69	1.00	1.97	0.766	8.0	2.3	1.9
Primary school [24]	242	8,317	68.74	0.53	1.00	1.73	0.975	4.5	1.8	1.0
Moreno innov. [14]	241	923	7.66	0.31	0.49	2.47	0.245	6.2	1.8	2.1
Gene fusion [14]	291	279	1.92	0.00	0.08	3.90	*0.024*	4.8	1.0	5.6
Copnet calls [23]	536	621	2.32	0.25	0.65	7.37	*0.024*	5.5	1.0	8.6
Copnet sms [23]	568	697	2.45	0.22	0.80	7.32	*0.026*	2.4	1.0	3.1
Copnet FB [23]	800	6,418	16.05	0.32	1.00	2.98	0.488	7.6	2.1	1.4
FB Reed98 [22]	962	18,812	39.11	0.33	1.00	2.46	0.778	5.3	2.3	1.0
Arenas email [14]	1,133	5,451	9.62	0.25	1.00	3.61	0.230	8.4	1.7	2.4
Euroroads [14]	1,174	1,417	2.41	0.02	0.03	18.37	*0.003*	1.8	1.0	3.7
Air traffic control [14]	1,226	2,408	3.93	0.07	1.00	5.93	*0.042*	5.4	1.0	5.6
Network science [14]	1,461	2,742	3.75	0.88	0.00	5.82	*0.039*	10.0	1.0	44.4
FB Simmons81 [22]	1,518	32,988	43.46	0.33	0.99	2.57	0.785	2.4	2.2	1.4
DNC emails [14]	1,866	4,384	4.70	0.59	0.98	3.37	0.092	15.5	1.5	1.4
Moreno health [14]	2,539	10,455	8.24	0.15	1.00	4.56	0.054	6.7	1.0	3.6
US power grid [14]	3,783	14,124	7.47	0.28	1.00	3.57	*0.008*	5.2	1.0	3.3
Bitcoin alpha [22]	4,941	6,594	2.67	0.11	1.00	18.99	0.113	66.5	1.5	1.9
GRQC collab. [16]	5,241	14,484	5.53	0.69	0.79	6.05	0.054	42.2	1.4	3.1
Pajek Erds [14]	6,927	11,850	3.42	0.40	1.00	3.78	*0.044*	55.7	1.0	4.0
FB GWU54 [22]	12,193	469,528	77.02	0.22	1.00	2.83	0.682	-	-	1.1
Enron email [16]	36,692	183,831	10.02	0.72	0.92	4.03	0.071	-	-	2.0
FB wall 2009 [14]	45,813	183,412	8.01	0.15	0.96	5.60	*0.033*	-	-	4.6
Brightkite [22]	58,228	214,078	7.35	0.27	0.97	4.92	*0.048*	-	-	2.6
Twitter [14]	465,017	833,540	3.59	0.06	1.00	4.59	*0.004*	-	-	1.4

6.2 Alteration Operations and Anonymity

Figure 1 shows results on how repeatedly applying each operation discussed in Sect. 3.2 affects the anonymity of graph models. First note that the number of unique nodes when no edges are altered varies with the average degree. Graphs with higher average degree have a higher initial uniqueness, which corresponds to findings in earlier work [12, 21]. Clearly, random deletion is more effective than random addition or rewiring, confirming what we theorized in Sect. 5.3. Interestingly, for the graph models, random addition and rewiring can even result in a higher uniqueness, as shown for the BA model, and the ER and WS models when the average degree is larger than 16. However, for the WS model, with average degree 16, rewiring performs slightly better than deletion when altering a fraction of 0.2 to 0.5 of the edges. This is likely due to the many nearly identical near-cliques in the generated networks.

While these preliminary results can not rule out that edge addition and rewiring can be more effective and perhaps preferred in specific cases, for exam-

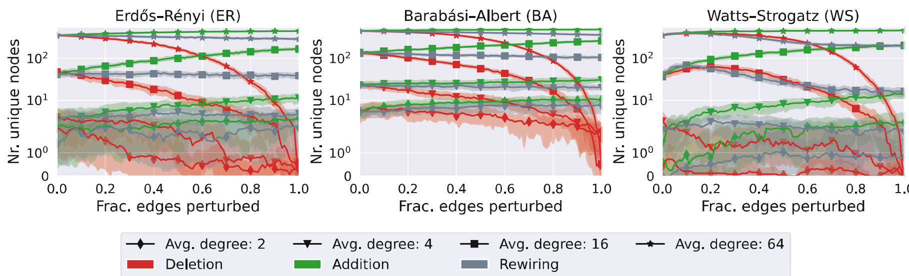

Fig. 1. Random edge deletion (red), addition (green), and rewiring (grey) applied to, using the COUNT measure, anonymize ER (left), BA (middle) and WS (right) graphs with $|V| = 500$ and average degree $\in \{2, 4, 16, 64\}$. (Color figure online)

ple when targeting specific edges, we choose to focus on edge deletion in the remainder of the paper, as it unequivocally performed best on average.

6.3 Anonymization and Measures

Figure 2 shows how uniqueness changes when deleting edges using different measures for anonymity and each algorithm in Sect. 5.3. We depict a selection of five networks representative of the overall observed behavior for the networks listed in Table 2. This is, from top to bottom, left to right in Fig. 2, 1) a plateau with a steep decrease, 2) linear decrease, 3) plateau with linear decrease, 4) sigmoid-like decrease and 5) initial increase before decrease.

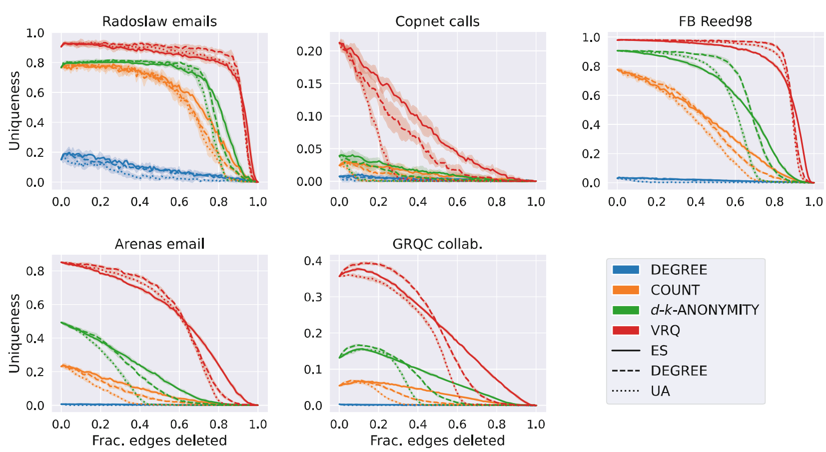

Fig. 2. Uniqueness for four different anonymity measures (color) and three anonymization algorithms (linestyle) on five real-world networks. For each network we show the fraction of deleted edges (horizontal axis) and the attained uniqueness (vertical axis).

Overall, we see that the measure for anonymity used influences both the initial uniqueness and how fast the uniqueness decreases. The simplest measure, DEGREE, has the lowest initial uniqueness and is the easiest to anonymize for. The measure with the furthest reach, VRQ, has the highest initial uniqueness and is the most difficult to anonymize for, corresponding with findings in [13].

Counterintuitively, in the "GRQC collab." network (bottom right), the ES algorithm shows an increase in uniqueness before decreasing. This is likely due to the collaboration network's many cliques. When randomly deleting edges, these cliques are destroyed, which may make certain nodes unique. If edges are targeted more specifically, for example, using the UA heuristic, this initial increase does not occur.

In the remainder of this paper, we focus on the COUNT measure, as this models a realistic attacker scenario, has a substantial initial uniqueness and is easier to anonymize for than, for example, the d-k-ANONYMITY measure.

6.4 Full, Partial and Budgeted Network Anonymization

In this section, we investigate how effective the proposed algorithms from Sect. 5.3 are at solving the three variants of the network anonymization problem.

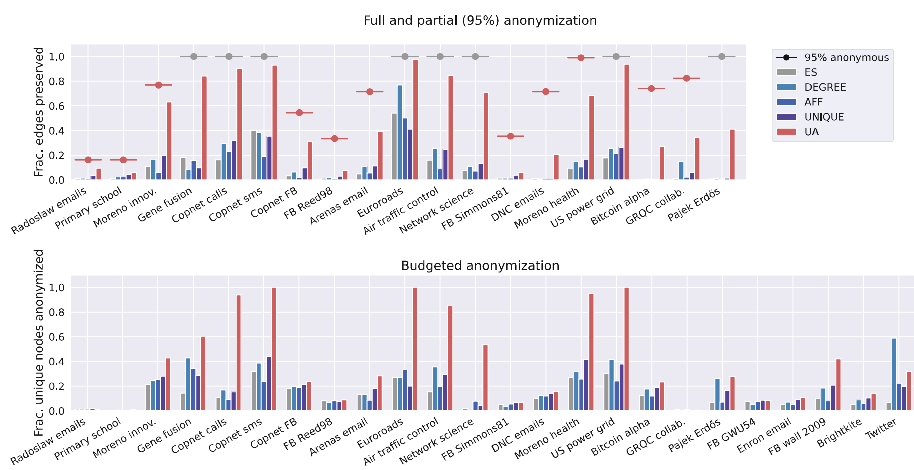

Fig. 3. Top: Results for full and partial anonymization. Bars indicate the fraction of edges preserved in full anonymization, horizontal lines the fraction of edges preserved for partial anonymization. Bottom: Bars indicate the fraction of unique nodes anonymized within the budget of $B = 0.05|E|$. Higher values correspond to better performance.

The top of Fig. 3 shows the fraction of edges preserved $\frac{|E'|}{|E|}$ after deleting edges until the desired level of anonymity is reached. For the partial variant, eight of the networks, including “US Power grid” and “Euroroads” retain 100% of their edges as initially less than 5% of their nodes is unique.

For some networks, such as “Radoslaw emails” and “FB Simmons81” only a small fraction of edges can be preserved. In networks where more edges can be preserved, the uniqueness based algorithm UA outperforms the other algorithms. Dividing the performance of UA by that of ES, we obtain the *improvement ratio* reported in the three rightmost columns of Table 2. On average, for the full and partial variants, 13.9 and 1.8 times more edges can be preserved, suggesting that even simple heuristics provide substantial improvement. The bottom of Fig. 3 shows results for the budgeted variant. To account for differences in initial uniqueness, the vertical axis shows the fraction of unique nodes that is anonymized, $1 - \frac{U(G')}{U(G)}$, when deleting at most 5% of the edges. A higher value indicates greater effectiveness within the given budget.

Overall, the UA algorithm performs well in anonymizing networks, particularly those with lower initial uniqueness, average degree, and higher average distance. Some of the networks, such as “Radoslaw emails” and “GRQC collab.”, require more than 5% of edge deletion to achieve substantial improvements. On average, in the budgeted variant, UA anonymizes 4.8 times more nodes than ES.

6.5 Utility

The utility of the anonymized networks in terms of how well various network properties and performance on network analysis tasks are preserved across the different variants of the anonymization problem is summarized in Fig. 4. A property is considered preserved if the value after anonymization, including one standard deviation, differs by less than 5% from the original value. For partial anonymization, we exclude the eight partially anonymous networks (initial uniqueness below 5%), leaving 11 networks.

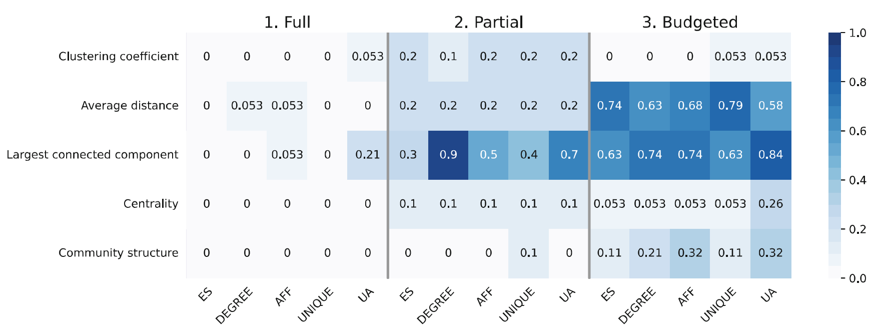

Fig. 4. For each variant of the anonymization problem (columns) and each algorithm (horizontal axis), cells indicate the fraction of networks for which utility (vertical axis) is preserved, i.e., the change ± one standard deviation is less than 5% of the original.

Comparing the three variants, we observe that for full and partial anonymization most properties are generally not preserved. Exceptions are the clustering coefficient and average distance which are preserved for one network, and the largest component size, which the UA algorithm preserves in four networks for the full variant—likely because UA requires fewer edge deletions and targets different edges. In the budgeted variant properties are preserved more often, and differences between algorithms are smaller. The UA algorithm best preserves the majority of properties, except for average shortest path length.

It is also important to note that even random edge deletion (ES) does not always preserve network properties. From the considered properties, average shortest path length and largest component size are preserved most consistently. The latter indicates that the networks often stay connected after anonymization. Community structure and the most central nodes are preserved for a substantial fraction of the networks. However, the average clustering coefficient seems challenging to preserve. These findings suggest that the budgeted variant provides a promising setting for further investigation and comparison of anonymization algorithms when data utility is of importance.

7 Conclusion

In this paper, we introduced the anonymization problem in social networks and its three variants. We introduced ANO-NET, a generic computational algorithmic framework that allows one to anonymize networks under these variants, supporting a range of anonymity measures, the proposed anonymization algorithms and options to explore relevant parameters. Experiments on graph models showed that, among the considered edge alteration operations, random edge deletion is most effective. Moreover, we found that stricter anonymity measures come with higher initial uniqueness and make the anonymization process more challenging. We proposed and evaluated five measure-agnostic anonymization algorithms. Their effectiveness varied across networks; those with lower initial uniqueness and higher average path length were easier to anonymize. The UA algorithm, which targets edges affecting the largest number of unique nodes, consistently showed to be most effective. Compared to the baseline, it anonymized on average 4.8 times more nodes in the budgeted variant and preserved 13.9 and 1.8 times more edges for full and partial anonymization, respectively. Additionally UA better preserved several utility metrics in the budgeted variant, realizing the best overall trade-off between anonymity and data utility.

This work lays the foundation for further studying the anonymization problem and its variants and for extending anonymity measures and algorithms to account for additional network properties such as node labels, layered edges or temporal aspects. Finally, dynamically monitoring utility during the anonymization process to guide edge alterations represents another promising avenue for future research.

References

1. Alavi, A., Gupta, R., Qian, Z.: When the attacker knows a lot: the gaga graph anonymizer. In: Proceedings of the 21st Springer International Conference on Information Security, pp. 211–230 (2019)
2. Barabási, A.L.: Network Science (2016)
3. Borgatti, S.P., Carley, K.M., Krackhardt, D.: On the robustness of centrality measures under conditions of imperfect data. Soc. Networks **28**(2), 124–136 (2006)
4. Campan, A., Truta, T.M.: Data and structural k-anonymity in social networks. In: Privacy, Security, and Trust in KDD, pp. 33–54. Berlin, Heidelberg (2009)
5. Chen, W., Wang, Y., Yang, S.: Efficient influence maximization in social networks. In: Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 199–208 (2009)
6. Chester, S., Kapron, B.M., Srivastava, G., Venkatesh, S.: Complexity of social network anonymization. Soc. Network Anal. Mining **3**(2), 151–166 (2013)
7. Csardi, G., Nepusz, T.: The igraph software package for complex network research. Int. J. Complex Syst., 1695 (2006)
8. Hagberg, A.A., Schult, D.A., Swart, P.J.: Exploring network structure, dynamics, and function using networkx. In: Proceedings of the 7th Python in Science Conference, pp. 11–15 (2008)
9. Hay, M., Miklau, G., Jensen, D., Towsley, D., Weis, P.: Resisting structural reidentification in anonymized social networks. In: Proceedings of the VLDB Endowment, vol. 1, pp. 102–114 (2008)
10. Jiang, H., Pei, J., Yu, D., Yu, J., Gong, B., Cheng, X.: Applications of differential privacy in social network analysis: a survey. IEEE Trans. Knowl. Data Eng. **35**(1), 108–127 (2021)
11. de Jong, R.G., van der Loo, M.P.J., Takes, F.W.: Algorithms for efficiently computing structural anonymity in complex networks. ACM J. Exp. Algorithmics **28** (2023)
12. de Jong, R.G., van der Loo, M.P.J., Takes, F.W.: The effect of distant connections on node anonymity in complex networks. Sci. Rep. **14**(1), 1156 (2024)
13. de Jong, R.G., van der Loo, M.P.J., Takes, F.W.: A systematic comparison of measures for k-anonymity in networks. arXiv preprint arXiv:2407.02290 (2024)
14. Kunegis, J.: Konect: the koblenz network collection. In: Proceedings of the 22nd International Conference on World Wide Web, pp. 1343–1350 (2013)
15. Lancichinetti, A., Fortunato, S.: Consensus clustering in complex networks. Sci. Rep. **2**(1), 336 (2012)
16. Leskovec, J., Krevl, A.: Snap datasets: stanford large network dataset collection (2014). http://snap.stanford.edu/data
17. Liu, K., Terzi, E.: Towards identity anonymization on graphs. In: Proceedings of the ACM SIGMOD International Conference on Management of Data, pp. 93–106 (2008)
18. Lu, X., Song, Y., Bressan, S.: Fast identity anonymization on graphs. In: Database and Expert Systems Applications, pp. 281–295 (2012)
19. Malik, A., Adams, B., Hassan, A.: Towards graph-anonymization of software analytics data: empirical study on JIT defect prediction. Empir. Softw. Eng. **29**(4), 76 (2024)
20. McKay, B.D., Piperno, A.: Practical graph isomorphism. II. J. Symb. Comput. **60**, 94–112 (2014)

21. Romanini, D., Lehmann, S., Kivelä, M.: Privacy and uniqueness of neighborhoods in social networks. Sci. Rep. **11**(1), 20104 (2021)
22. Rossi, R.A., Ahmed, N.K.: The network data repository with interactive graph analytics and visualization. In: Proceedings of the 29th AAAI Conference on Artificial Intelligence, pp. 4292–4293 (2015)
23. Sapiezynski, P., Stopczynski, A., Lassen, D.D., Jørgensen, S.L.: The copenhagen networks study interaction data. figshare (2019). https://doi.org/10.6084/m9.figshare.7267433.v1
24. Sociopatters: Datasets (2021). http://www.sociopatterns.org/datasets/
25. 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 (2009)
26. Traag, V.A., Waltman, L., Van Eck, N.J.: From Louvain to Leiden: guaranteeing well-connected communities. Sci. Rep. **9**(1), 1–12 (2019)
27. Zhou, B., Pei, J.: The k-anonymity and l-diversity approaches for privacy preservation in social networks against neighborhood attacks. Knowl. Inf. Syst. **28**(1), 47–77 (2011)