Noname manuscript No.
(will be inserted by the editor)

k-Degree Anonymity on Directed Networks

Jordi Casas-Roma · Julián Salas · Fragkiskos D. Malliaros · Michalis Vazirgiannis

Received: date / Accepted: date

Abstract In this paper, we consider the problem of anonymization on directed networks. Although there are several anonymization methods for networks, most of them have explicitly been designed to work with undirected networks and they can not be straightforwardly applied when they are directed. Moreover, ignoring the direction of the edges causes important information loss on the anonymized networks in the best case. In the worst case, the direction of the edges may be used for reidentification, if it is not considered in the anonymization process. Here, we propose two different models for k-degree anonymity on directed networks, and we also present algorithms to fulfill these k-degree anonymity models. Given a network G, we construct a k-degree anonymous network by the minimum number of edge additions. Our algorithms use multivariate micro-aggregation to anonymize the degree sequence, and then they modify the graph structure to meet the k-degree anonymous sequence. We apply our algorithms to several real datasets and demonstrate their efficiency and practical utility.

Keywords Anonymity · Social networks · Directed networks · Data utility · Privacy

Jordi Casas-Roma
Universitat Oberta de Catalunya
Barcelona, Spain
E-mail: jcasasr@uoc.edu

Julián Salas
Universitat Oberta de Catalunya
Barcelona, Spain
E-mail: jsalaspi@uoc.edu

Fragkiskos D. Malliaros
CentraleSupélec, University of Paris-Saclay and Inria Saclay
Gif-sur-Yvette, France
E-mail: fragkiskos.malliaros@centralesupelec.fr

Michalis Vazirgiannis
École Polytechnique
Palaiseau, France
E-mail: mvazirg@lix.polytechnique.fr

1 Introduction

In recent years, a huge amount of social and human interaction networks have been made publicly available. Embedded within this data, there is user's private information that must be preserved before releasing the data to third parties and researchers. The study by Ferri et al. [16] reveals that up to 90% of user groups are concerned by data owners sharing data about them. Backstrom et. al. [1] point out that the simple technique of anonymizing graphs by removing the identities of the vertices before publishing the actual graph does not always guarantee privacy. In particular, they have shown that an adversary can infer the identity of the vertices by solving a set of restricted graph isomorphism problems. It is evident that network anonymization processes become an important issue under this scenario.

Several methods have been developed to protect users' privacy on networks, but none of them has been designed specifically for directed networks. Some methods remove the direction of edges in order to convert directed networks to undirected ones and then they utilise undirected algorithms to protect users' privacy. This has two drawbacks. First, if the published network is undirected, the direction of the edges is lost, hence in the published version there may be connected nodes that were not connected by a directed path in the original directed graph. Second, if the network is anonymized without considering the direction of the relations, then this information may be used for reidentification, that is the case when considering k-degree anonymization without considering the direction of the edges. However, removing the direction of the edges produces a severe loss of information regarding the structure of the network, in the sense that the in-degree and out-degree of each node are combined into a single characteristic that is anonymized using models designed for undirected networks. There are cases where we are interested to treat the in-degree and out-degree sequences of a graph in a different manner – and not as the combined undirected degree – during the anonymization process. For example, in Twitter's who-follows-whom social graph, one may be interested to consider different levels of anonymity for the in-degree (followers) and the out-degree (followees) of a user, as the out-degree may contain more sensitive information (e.g., in the case of a celebrity), and is also relevant to consider the direction of the relation (who follows whom), since the flow of information goes only in one direction (e.g., a celebrity does not knows what his followers post).

1.1 Our contributions

In this paper, we define two k-anonymity models specifically designed for directed networks. Additionally, we present algorithms to implement these models and empirically demonstrate their practical application on real directed networks. Since these graphs have no attributes or labels on the edges, information is contained only in the structure of the graph itself and, due to this, preserving network's structure and edges' direction are critical to reduce information loss. The contributions of this work can be summarized as follows:

- We define two different models for k-anonymity on directed networks, offering different privacy protection levels.
- We introduce algorithms to achieve the desired privacy levels based on the previously proposed models.

- We show that our algorithms are able to deal with large networks of thousands and millions of vertices and edges, demonstrating their practical utility in real-world problems.
- We conduct an empirical evaluation of these models on several real networks, comparing information loss based on different graph properties and also on clustering-specific processes.
- We demonstrate that our models preserve data privacy, while simultaneously conduct the anonymization process towards reducing information loss and increasing data utility.

1.2 Notation

Let $G = (V, A)$ be a directed and unlabeled graph (also called *digraph*), where V is the set of vertices (or nodes) and A the set of arcs (or edges) in G. We define $n = |V|$ to denote the *number of vertices* and $m = |A|$ to denote the *number of arcs*. We use $(v_i, v_j) \in A$ to denote a directed arc from vertex v_i to v_j but not vice versa. Finally, we denote by $G = (V, A)$ and $\widetilde{G} = (\widetilde{V}, \widetilde{A})$ the original and the perturbed graph produced by the anonymization process, respectively.

1.3 Roadmap

This paper is organized as follows. In Section 2, we review the related work and the state of the art on privacy-preserving methods for networks. Section 3 introduces the preliminary concepts and our k-anonymity models for directed graphs. Then, in Section 4, we propose algorithms to fulfill the privacy levels pointed out by our models[1]. Our experimental framework is provided in Section 5, and then we discuss the results in terms of information loss and data utility in Section 6. Experiments about scalability issues are presented in Section 7. Lastly, we discuss the conclusions of this work and future research directions in Section 8.

2 Related Work

The k-anonymity model was introduced in [36,37] for privacy preservation on structured or relational data. Formally, the k-anonymity model is defined as follows: let $RT(A_1, \ldots, A_n)$ be a table and QI_{RT} be the quasi-identifier associated with it. RT is said to satisfy k-anonymity if and only if each sequence of values in $RT[QI_{RT}]$ appears with at least k occurrences in $RT[QI_{RT}]$. The k-anonymity model indicates that an attacker can not distinguish between different k records although he manages to find a group of quasi-identifiers. Therefore, the attacker cannot re-identify an individual with a probability greater than $\frac{1}{k}$.

Several concepts can be used as quasi-identifiers for k-anonymity on graph structured data. A widely applied option is to use the vertex degree as a quasi-identifier. Accordingly, we assume that the attacker knows the degree of some target vertices. If the attacker identifies a single vertex with the same degree in

[1] The source code for the paper is available at: https://jcasasr.wordpress.com/software/dga

the anonymous graph, then he has re-identified this vertex. That is, $deg(v_i) \neq deg(v_j)\ \forall j \neq i$. This model is called k-degree anonymity [25] and these methods are based on modifying the graph structure (by edge modifications) to ensure that all vertices satisfy k-anonymity for their degree. In other words, the main objective is that all vertices have at least $k-1$ other vertices sharing the same degree. Furthermore, Liu and Terzi [25] developed a method based on dynamic programming and edge switch in order to construct a new k-degree anonymous graph, where $V = \widetilde{V}$ and $E \cap \widetilde{E} \approx E$. Their work inspired many other authors who proposed improved solutions based on different kinds of heuristics, such as [29, 19, 6, 10].

Instead of using the vertex degree, Zhou and Pei [40] consider the 1-neighbourhood subgraph of the objective vertices ($\Gamma(v)$) as a quasi-identifier. For a vertex $v_0 \in V$, v_0 is k-anonymous in G if there are at least $k-1$ other vertices $v_1, \ldots, v_{k-1} \in V$ such that $\Gamma(v_0), \Gamma(v_1), \ldots, \Gamma(v_{k-1})$ are isomorphic. They demonstrated that the neighborhood anonymity problem for vertex-labeled graphs is NP-hard. Other authors modeled more complex adversary knowledge and used them as quasi-identifiers. For instance, Hay et al. [21] proposed a method called *k-candidate anonymity*, where a vertex v_0 is k-candidate anonymous with respect to question Q if there are at least $k-1$ other vertices in the graph with the same answer. Formally, $|cand_Q(v_0)| \geq k$ where $cand_Q(v_0) = \{v_j \in V : Q(v_0) = Q(v_j)\}$. A graph is k-candidate anonymous with respect to question Q if all of its vertices are k-candidate with respect to Q. Zhou et al. [42] and Zhou and Pei [41] considered all structural information about a target vertex as quasi-identifier and proposed a new model called *k-automorphism* to anonymize a network and ensure privacy against this attack. They define a k-automorphic graph as follows: (a) if there exist $k-1$ automorphic functions $F_a (a = 1, \ldots, k-1)$ in G, and (b) for each vertex v_i in G, $F_{a_1}(v_i) \neq F_{a_2} (1 \leq a_1 \neq a_2 \leq k-1)$, then G is called a k-automorphic graph.

Rossi et al. [32] studied the problem of k-degree anonymization on time-varying (and multilayer) graphs. Let $\mathcal{G} = \{G_1, \ldots, G_T\}$ be a time-varying graph with a fixed set of vertices V, where $|V| = n$. In other words, $\mathcal{G}$ is defined as a sequence of undirected graphs $G_t = (V, E_T),\ t = 1, \ldots, T$, where E_t denotes the set of edges at time t. Also, let $D = \{d_{it}\}$ be the $n \times T$ degree matrix, where d_{it} is the degree of the i-th node of G_t. We say that matrix D is a set of k-anonymous vectors, if for every row $d_{i:}$ there are at least $k-1$ vectors $d_{j:}$ such that $d_{it} = d_{jt}$, for each $t = 1 \ldots, T$. Then, a time-varying graph $\mathcal{G}$ is defined to be k-degree anonymous, if the degree D defines a set of k-anonymous vectors. Similar to the work of Liu and Terzi [25], the authors of [32] propose a three-step approach where firstly they enforce anonymity, then enforce realizability, and finally construct the graph. However, their realizability constraints are only for undirected graphs.

All the aforementioned methods work only with simple and undirected graphs, and it is not straightforward to extend those methods to directed networks. The naïve approach to convert digraphs to undirected graphs, anonymize them and finally transform back to directed graphs, causes severe perturbations to the graph's structure. We will provide an empirical example of such approach in Section 6. Other works focus on the problem of edge-weight anonymization, e.g., [13] aims at anonymizing the weights of a graph with the aim of preserving the utility for algorithms such as the Minimum Spanning Tree – thus, emphasizes at preserving the inequalities among the edge weights; [24] protects the weights of the edges

by adding Gaussian noise to them. To sum up, those methods preserve the edge weights and not the amount of edge relations, hence, they cannot be adapted to degree anonymity for directed networks. Alternatively, other types of privacy-preserving methods can easily be extended to work with directed networks, such as randomization techniques [17] or class-based generalization techniques [2,12].

However, [2,12] consider preventing an attacker from learning interactions between entities, which is equivalent to protecting against edge disclosure in bipartite graphs, that for example may represent users and interactions, or costumers and products. The authors of [17] aim at explicitly preserve the degrees of the nodes while randomizing the graphs. Thus, even when adapted for directed graphs, those approaches may still be vulnerable to attacks based on the degrees.

Therefore, in this paper, we are interested in proposing a k-anonymity model specifically designed for directed networks and also to develop algorithms to protect user's privacy with guarantees of the k-degree anonymity model.

Related to the complexity of k-degree anonymization algorithms, Hartung et. al. [20] proved that the problem of degree anonymity (by only adding edges) is NP-hard on 3-colorable graphs and on graphs with H-index three. Also, they proved that there is a polynomial-time algorithm that transforms any instance of the degree anonymity problem into an equivalent instance with at most $O(\Delta^7)$ nodes. A similar result is obtained in [3] for directed graphs, that is, a polynomial size problem kernel for the combined parameter (s, Δ_D), where Δ_D denotes the maximum in- or out-degree of the input digraph D and s is the number of edges to be added. We emphasize that both papers obtain solutions for the original question of Liu and Terzi of obtaining a k-degree anonymous graph, that contains the original graph as subgraph, and are equivalent to generating graphs with specified degree sequences and excluded graphs, as in [34]. While we argue that the original edges are to be preserved as much as possible, we are aware that there are many cases where this is not possible. So, we propose an algorithm that tries to obtain k-degree anonymous directed graphs by only adding edges until it is necessary to modify the original graph.

3 k-Anonymity Models on Directed Networks

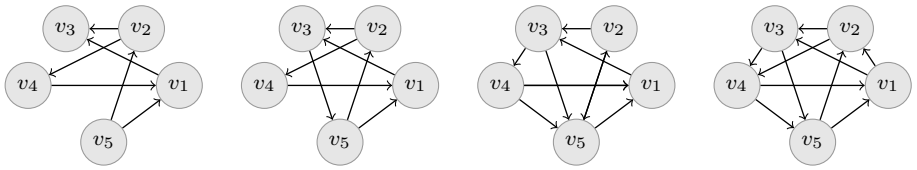
In this section, we define our models based on k-degree anonymity to preserve user's privacy on directed networks.

3.1 k-degree anonymity

The concept of k-degree anonymity was proposed by Liu and Terzi in [25] for undirected networks and it can be directly mapped to the degree sequence.

Definition 1 A vector of integers V is k-anonymous, if every distinct value $v_i \in V$ appears at least k times.

Definition 2 An undirected network $G = (V, E)$ is k-degree anonymous, if the degree sequence of G is k-anonymous.



(a) Original network (b) Independent 2-anonymous (c) Independent $(1,2)$-anonymous (d) Paired 5-degree anonymous

Fig. 1: Toy example showing an anonymization process from the original graph to an Independent 2-degree, Independent $(1,2)$-degree and Paired 5-degree anonymous versions of the same network.

Let V and W correspond to the degree sequences of the input and anonymized graph respectively. The distance between two vectors of integers $V = [v_1, \ldots, v_n]$ and $W = [w_1, \ldots, w_n]$ is defined by Equation 1:

$$\Delta(V, W) = \sum_{i=1}^{n} |v_i - w_i|, \tag{1}$$

where $v_i \in V$, $w_i \in W$ and $|V| = |W| = n$. The lower the value of Δ, the lower the information loss of the anonymized network.

3.2 k-degree anonymity for directed networks

Direct successors of vertex $v_i \in V$, denoted by $\Gamma^+(v_i)$, are defined as the vertices at distance 1 from v_i, i.e. all $v_j : (v_i, v_j) \in A$. The number of successors is defined as the vertex's out-degree, $d_{out}(v_i) = |\Gamma^+(v_i)|$. Similarly, direct predecessors of vertex v_i are all vertices from which v_i can be reached at one hop. That is, $\Gamma^{-1}(v_i) = \{v_j : (v_j, v_i) \in A\}$ and vertex's in-degree is defined as $d_{in}(v_i) = |\Gamma^{-1}(v_i)|$. Therefore, a directed graph is associated with two degree sequences: the in-degree sequence, $d_{in} = \{d_{in}(v_1), \ldots, d_{in}(v_n)\}$, and the out-degree sequence, $d_{out} = \{d_{out}(v_1), \ldots, d_{out}(v_n)\}$. Since each arc connects two vertices, it is obvious that:

$$\sum_{i=1}^{n} d_{in}(v_i) = \sum_{i=1}^{n} d_{out}(v_i). \tag{2}$$

It is important to note that, in the anonymization process, the same number of arcs have to be added to both in-degree and out-degree, since each added arc implies adding value one to in-degree and also to out-degree. Thus, anonymous in-degree and out-degree have to satisfy Equation 2.

Next, we propose two models to achieve different privacy levels according to the k-anonymity model.

3.2.1 ***Independent (k_i,k_o)-degree anonymity***

This model assumes that an adversary knows the in-degree or the out-degree of some target vertices, but does not knows the in- and out-degree of the target vertices.

Definition 3 A directed network $G = (V, A)$ is Independent (k_i,k_o)-degree anonymous if the in-degree sequence of G is k_i-anonymous and the out-degree sequence is k_o-anonymous.

In the case that $k_i = k_o = k$, we simply call it Independent k-degree anonymity.

Definition 4 A directed network $G = (V, A)$ is Independent k-degree anonymous if both the in-degree and the out-degree sequences of G are k-anonymous.

Example 1 A toy example of Independent k-degree anonymity can be seen in Figure 1. The original network, shown in Figure 1a, contains 5 vertices and 6 arcs and its degree sequences are $d_{in} = \{2, 1, 2, 1, 0\}$ and $d_{out} = \{1, 2, 0, 1, 2\}$. Thus, adding just one arc from v_3 to v_5 is enough to convert this network into a Independent 2-degree anonymous graph. Figure 1b shows the anonymous network, which has $d_{in} = \{2, 1, 2, 1, 1\}$ and $d_{out} = \{1, 2, 1, 1, 2\}$.

Example 2 The graph represented in Figure 1b is also $(2, 2)$-anonymous according to our definition. However, using this model we are able to create asymmetric privacy levels if we consider that, for example, the out-degree of some target vertices can be the main knowledge of an adversary and we want to protect our network accordingly. Figure 1c shows an Independent $(1, 2)$-anonymous version of the graph, where $d_{in} = \{2, 1, 2, 1, 3\}$ and $d_{out} = \{1, 1, 2, 2, 1\}$. Hence, it is possible to re-identify a user using in-degree information but it is not possible using information related to out-degree of some target vertices.

3.2.2 ***Paired k-degree anonymity***

This model assumes that an adversary knows both the in-degree and the out-degree of some target vertices. Obviously, this model gives us a higher privacy protection than the above models, since it also protects users from an adversary who knows only the in- or out-degree of some target vertices. We define the paired degree of a vertex as a pair of integer numbers, where the first one is the in-degree of the vertex and the second one is the out-degree, that is, $(d_{in}(v_i), d_{out}(v_i))$.

Definition 5 A directed network $G = (V, A)$ is Paired k-degree anonymous if the paired degree sequence of G is k anonymous, i.e., for each pair (a, b) representing the in-degree and the out-degree of a vertex, there exist at least $k - 1$ other pairs with the same values.

Notice that, a Paired k-degree anonymous graph is always an Independent (k,k)-degree anonymous one, but not vice versa. Thus, Paired k-degree anonymity is stronger than Independent (k,k)-degree anonymity.

Example 3 Figure 1d presents the Paired 5-degree anonymous version of our toy example. Four arcs must be added to fulfill the properties of this model, and its degree sequences are $d_{in} = \{2, 2, 2, 2, 2\}$ and $d_{out} = \{2, 2, 2, 2, 2\}$. It is interesting to see that this network is also Independent $(5, 5)$-degree anonymous. Moreover, the network depicted in Figure 1c is Independent $(2, 2)$-anonymous, but it is not Paired 2-anonymous. Actually, it is Paired 1-anonymous.

4 Anonymization of Directed Graphs

In this section, we present the DGA (Directed Graph Anonymization) algorithm, designed to preserve user's privacy on directed and unlabeled networks according to the proposed anonymization models. We use the concept of k-degree anonymity to anonymize users' relationship, performing modifications only on the edge set, so as to generate a new anonymous graph $G^k = (V, A \cup A^k)$, where G^k is k-degree anonymous and $|A^k|$ is minimized.

Our approach to anonymize a directed graph relies on Definition 2. Thus, we anonymize both the in-degree and the out-degree sequences of $G = (V, A)$ by edge-addition in order to meet the k-degree anonymity for a directed graph. Our approach is based on two steps (similar to the one in [25]):

1. *Anonymization of degree sequences.* We construct a k-degree anonymous sequence $d^k_{in} = \{d^k_{in}(v_1), \ldots, d^k_{in}(v_n)\}$ from the in-degree sequence $d_{in} = \{d_{in}(v_1), \ldots, d_{in}(v_n)\}$ of the original graph using Definition 1. The same process is applied to obtain an anonymized version of the out-degree sequence, d^k_{out}.
2. *Adding fake arcs.* The second step adds fake arcs between vertices to meet the anonymized in-degree (d^k_{in}) and out-degree (d^k_{out}), achieving a k-degree anonymous directed graph $G^k = (V, A \cup A^k)$, where $|A^k|$ is minimized.

4.1 Step I: Anonymization of degree sequences

This step provides the anonymity level through the in- and out-degree sequences. Therefore, we develop two different strategies according to the privacy models we have introduced previously. First, we present the algorithm for Independent (k_i,k_o)-degree anonymity, and later we propose a second approach for achieving Paired k-degree anonymity. Last but not least, we detail a post-processing method that needs to be applied when Equation 2 is not satisfied after the anonymization of degree sequences.

4.1.1 ***Independent (k_i,k_o)-degree anonymity***

We refer to (k_i,k_o)-DGA when Independent (k_i,k_o)-degree is considered. The same process is applied both to in-degree (d_{in}) and out-degree (d_{out}) sequences, therefore we will detail the process on a general degree sequence (d). The objective of this step is to anonymize the degree sequence of the original network, d. Optimal univariate micro-aggregation by Hansen and Mukherjee [18] is used to achieve the best group distribution for both in-degree and out-degree sequences and then we compute the values for each group that minimize the distance from the original degree sequences by Equation 1. We choose such algorithm with complexity $O(k^2 n)$

for its flexibility; by changing only one parameter, it can compute the optimal k-anonymous degree sequences for different metrics such as euclidean, linear, or any function of the nodes in the k-groups – contrary to Clarkson et al.'s algorithm [11] that has complexity $O(n)$ but is specifically tailored for taking the maximum on the k-groups. Moreover, we implement it with the improvements proposed by [35], which greatly reduce the execution time. Note that, the degree sequence anonymization is the less expensive part, as can be seen on Table 5.

Our approach starts by applying a permutation f to the degree sequence to reorder the elements. We refer to the ordered degree sequence as a monotonic, non-decreasing sequence of the vertex' degrees, that is $d(v_i) \leq d(v_j)\ \forall i < j$. Let k be an integer such that $1 \leq k < n$ which is the k-degree anonymity value, i.e. k_i in case of in-degree and k_o otherwise. Typically, k is much smaller than n. In order to apply the optimal univariate micro-aggregation and according to [18], we construct a new directed network $H_{k,n}$ and get the optimal partition which is exactly the set of groups that corresponds to the arcs of the shortest path from vertex 0 to vertex n on this graph. We denote by $g = \{g_1, \ldots, g_p\}$ the optimal partition, where $\frac{n}{2k-1} \leq p \leq \frac{n}{k}$, and each of them has between k and $2k-1$ items. Obviously, each $d_i \in d$ belongs to a specific group $g_j \in g$. Since our approach relies only on edge addition to modify the graph structure, we have to increase or keep the same degree values, but not to decrease any of them which would be equivalent to an edge removal. Therefore, the optimal partition corresponds to increasing the value of each vertex's degree up to the maximum value of its group, i.e., $d_i = \max(d_q)\ \forall d_i, d_q \in g_j$. The cost of the shortest path on $H_{k,n}$ denotes the number of added arcs that is needed in order to meet the k-anonymity value.

4.1.2 ***Paired k-degree anonymity***

We refer to k-DGA when Paired k-degree is considered. In this model, we need to consider simultaneously both the in- and out-degree of each vertex. Thus, each pair $(d_{in}(v_i), d_{out}(v_i))$ represents the in-degree and the out-degree of a vertex v_i. According to Definition 5, we must find the optimal partition in this 2-dimensional space. The decision problem of finding a paired k-degree anonymous sequence by adding exactly s edges (referred to as the *Numbers Only Digraph Degree Anonymity* problem), was proven to be NP-hard (Ref. [4], Theorem 23). Hence, we use multivariate microaggregation to find quasi-optimal partitions in a reasonable time; specifically, we have applied the MDAV algorithm [15,14]. Similarly to the aforementioned method, the optimal partition corresponds to increasing the pair values of each vertex's degree up to the maximum pair values of its group.

4.1.3 ***Degree sequences post-processing***

It is important to note that the same number of arcs need to be added to the in-degree and out-degree sequences, since each new arc implies adding value one to both the in-degree and out-degree sequence. Consequently, anonymous in-degree and out-degree sequences have to satisfy Equation 2.

We denote as η_{in} the number of added arcs on the in-degree and by η_{out} the number of added arcs on the out-degree sequence, for a given k-degree anonymization of a directed graph G. If $\eta_{in} \neq \eta_{out}$, our anonymous degree sequences do not satisfy Equation 2, which is required for directed graphs. Hence, the minimum

number of arcs we must add to the original graph is at least $\max\{\eta_{in}, \eta_{out}\}$, if we consider that η_{in}, η_{out} are the number of edges needed in an optimal microagreggation for the in/out-degree sequences, respectively. Hence if we get a k-degree anonymous sequence with $max(\eta_{in}, \eta_{out})$ edges, then, we know that it is optimal.

Let S_{in} and S_{out} be the optimal in- and out-degree sequence partition obtained after applying the micro-aggregation algorithms, where $S_{in} = \cup_{i=1}^{p} s_{in}^i$ and $S_{out} = \cup_{i=1}^{q} s_{out}^i$. Note that the number of partitions does not have to be equal ($p \neq q$). Also, it is important to note that the minimal edge addition to fulfill Equation 2 is represented by finding the minimal values to solve:

$$\sum_{i=1}^{p} c_{in}^i + \alpha_i \times |s_{in}^i| = \sum_{i=1}^{q} c_{out}^i + \beta_i \times |s_{out}^i|, \tag{3}$$

where c_{in}^i and c_{out}^i represents the number of added edges at partition i computed by Equation 4, $\alpha_i, \beta_i \geq 0$ and $\alpha_i, \beta_i \in \mathbb{N}$:

$$c_{in}^i = \sum |v_j - \Delta_i| : v_j \in s_{in}^i, \tag{4}$$

where $\Delta_i = max\{deg(v_j) : v_j \in s_{in}^i\}$. In order to simplify the equation and the calculations, we consider only the different sizes of s_{in}^i and s_{out}^i, which are denoted by a_i and b_i respectively. We will denote $\sum_{i=1}^{p} c_{in}^i - \sum_{i=1}^{q} c_{out}^i$ as R. Then, we can obtain the following equation from Eq. 3:

$$\sum_{i=1}^{p'} \alpha_i a_i + R = \sum_{i=1}^{q'} \beta_i b_i, \tag{5}$$

where $p' < p$ and $q' < q$, since we are taking out the repeated values of $|s_{in}^i|$ and $|s_{out}^i|$. For the same reason, the values of α_i, β_i in Equation 5 are different from the values in Equation 3.

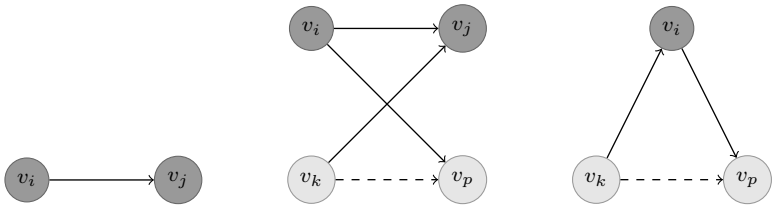
Recall that in optimal microaggregation, $k \leq |s_{in}^i|, |s_{out}^i| \leq 2k - 1$ for all $i \leq max(p, q)$. Hence, $k \leq a_i, b_i \leq 2k - 1$ for all $i \leq max\{p', q'\}$. If we assume that $\beta_{i_0} \neq 0$ for a given i_0, then we obtain the equation:

$$\sum_{i=1}^{p'} \alpha_i a_i + R - \sum_{i \neq i_0} \beta_i b_i = \beta_{i_0} b_{i_0} \tag{6}$$

Therefore, a solution can be obtained by solving the following equation:

$$\sum_{i=1}^{p'} \alpha_i a_i + R - \sum_{i \neq i_0} \beta_i b_i \equiv 0 \pmod{b_{i_0}} \tag{7}$$

Now, since we are working with congruences (mod b_{i_0}) we can consider the coefficients α_i, β_i to be less than b_{i_0}, which gives a large reduction of our search space for the solutions. In the worst case, we can obtain a solution by brute force, considering all the combinations of $\alpha_i, \beta_i \leq b_{i_0}$ which would be a search in $O(k^k)$, since $b_{i_0} \leq 2k$. Moreover, in practice we can find solutions to Eq. 7 much faster. In all the sequences we have studied, $\alpha_i = 0$ for all i. While, for some $i_1 \neq i_0$ and β_{i_1} the congruence $R - \beta_{i_1} b_{i_1} \equiv 0 \pmod{b_{i_0}}$ was verified, so it is enough in most cases to consider only one variable $i_1 \neq i_0$.



(a) Edge addition (b) Edge switch (c) Edge extension

Fig. 2: Illustration of edge addition, switch and extension processes. Solid lines represent new edges to be added and dashed lines existing edges to be deleted. Vertex color indicates whether a vertex changes its degree (dark grey) or not (light grey) after the edge modification has been carried out.

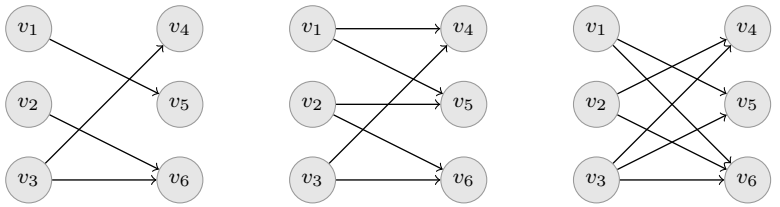
4.2 Step II: Graph modification

As mentioned earlier, our algorithm is based on adding fake arcs. Other methods anonymize the graph's structure by adding and removing arcs, instead of additions only. In our approach, we consider keeping arcs of the original network, since true relations between users can be important for clustering or other graph mining tasks. The authors of [8] empirically proved that edge addition is the best method to keep graph's properties when perturbing scale-free networks, which constitute the most common type of real-world networks.

Once we have computed the k-degree anonymous in-degree and out-degree sequences, our approach computes the vector of differences between the original and anonymous sequences. That is, $\delta_{in} = d^k_{in} - d_{in}$ and $\delta_{out} = d^k_{out} - d_{out}$. Each vector clearly shows which vertices have to increase their in-degree (δ_{in}) and out-degree (δ_{out}). For each of them, we use three edge modification processes to increase the in- and out-degree of vertices in δ_{in} and δ_{out} respectively, which are the following:

1. **Edge addition** randomly chooses a combination of vertices which satisfies $(v_i, v_j) \notin A$, where $v_i \in \delta_{out} : \delta_{out}(v_i) > 0$ and $v_j \in \delta_{in} : \delta_{in}(v_j) > 0$. The out-degree of vertex v_i and the in-degree of v_j both increase, as shown in Figure 2a.
2. **Edge switch** occurs between four vertices $v_i, v_j, v_k, v_p \in V$ where $(v_i, v_j), (v_k, v_p) \in A$ and $(v_i, v_p), (v_k, v_j) \notin A$. It is defined by deleting arc (v_k, v_p) and adding new arcs (v_i, v_p) and (v_k, v_j), as Figure 2b illustrates. Note that, the out-degree of vertex v_i and the in-degree of vertex v_j will increase by 1, while other vertices' degree will remain the same.
3. **Edge extension** exists between three vertices $v_i, v_k, v_p \in V$, where $(v_k, v_p) \in A$ and $(v_k, v_i), (v_i, v_p) \notin A$. Arc (v_k, v_p) is deleted and new arcs (v_k, v_i) and (v_i, v_p) are created, as Figure 2c illustrates. Note that the in- and out-degree of vertex v_i increases, while auxiliary vertices' degree remain the same.

The process is described in Algorithm 1. For each vertex $v_i \in \delta_{out}$, the algorithm finds $v_j \in \delta_{in}$ and adds an arc between them. Due to the edge sparsity of real networks, this process is possible in several cases. However, in some cases it is not possible to create a fake edge as described previously. Then, we propose to use edge



(a) Original graph (b) Adding (v_1, v_4) and (v_2, v_5) (c) Adding (v_1, v_6), (v_2, v_4) and (v_3, v_5)

Fig. 3: The order for adding edges may be relevant.

switch or edge extension to alter graph's structure to fulfill the anonymous degree sequences. It may be the case that the order of adding edges may be relevant, as in Figure 3. Suppose that the anonymous sequence should be $\delta_{out} = (2, 2, 3)$ and $\delta_{in} = (2, 2, 3)$, and the algorithm added the edges (v_1, v_4) and (v_2, v_5) first, as in (b). In this case, it will not be possible to apply any of our three edge modification processes to add one to the degrees of nodes v_3 and v_6, however by adding the correct edges the sequence could have been obtained with our edge modification processes, as in (c). Notice that, we have never encountered such situation in our experiments possibly because of the sparsity of social networks, and due to the fact that our algorithms choose the added edges at random – arriving at such situation that no possible edge can be added, will only require to re-run the algorithm to avoid it.

5 Experimental Framework

In this section, we will describe the experimental framework we have used to analyse and compare the information loss induced by our anonymization methods. For each dataset, we compute the Paired and Independent k-degree anonymous networks considering different values of k in the range $[1, 10]$. Notice that, $k = 1$ corresponds to the original network. Independent (k_i, k_o)-degree anonymous networks are evaluated in the range of $k_i \in [1, 10]$ and $k_o \in [1, 10]$; this implies a total of 100 anonymous networks for each dataset.

5.1 Description of network datasets

We have used five standard and well-known real networks to test our methods: (1) Polblogs [22], a network of hyperlinks between weblogs on US politics; (2) UC-Irvine [23], which contains messages sent between the users of an online community of students from the University of California, Irvine; (3) Wiki-vote [28] (Wikipedia vote network) contains all the Wikipedia voting data from the inception of Wikipedia till January 2008, where vertices in the network represent wikipedia users and a directed edge from node v_i to node v_j represents that user i voted on user j; (4) DBLP-cite [27] is the citation network of DBLP, a database of scientific publications such as papers and books, where each vertex is a publication

```
Function graph_modification_process
    Input: δ_in, δ_out, V and A
    Output: Anonymized arc set (Ã).
    for v_i : δ_out(v_i) > 0 do
        if ! edge_addition (v_i) then
            for v_j : δ_in(v_j) > 0 do
                if (v_i, v_j) ∈ A then
                    edge_switch (v_i, v_j)
                else
                    edge_extension (v_i)
                end
            end
        end
    end
    return Ã
end
Function edge_addition(v_i)
    for v_j : δ_in(v_j) > 0 do
        if (v_i, v_j) ∉ A then
            create (v_i, v_j)
            return true
        end
    end
    return false
end
Function edge_switch(v_i, v_j)
    find v_k, v_p : (v_k, v_p) ∈ A and (v_k, v_j), (v_i, v_p) ∉ A
    delete (v_k, v_p)
    create (v_i, v_p) and (v_k, v_j)
    return true
end
Function edge_extension(v_i)
    find v_k, v_p : (v_k, v_p) ∈ A and (v_k, v_i), (v_i, v_p) ∉ A
    delete (v_k, v_p)
    create (v_k, v_i) and (v_i, v_p)
    return true
end
```

Algorithm 1: Edge modification process.

and each edge represents a citation of a publication by another publication; and (5) Epinions [31] is a who-trust-whom online social network of a general consumer review site Epinions.com, where members of the site can decide whether to "trust" each other. We have selected these datasets because they have diverse statistics and properties, as shown in Table 1. We have removed loops and multiple edges from all analyzed networks.

5.2 Information loss evaluation

In this part, we describe the criteria that are used to quantify the information loss that is introduced by our anonymization models. Following the approach presented in [7], we use diverse structural measures which are strongly or moderately correlated with clustering-specific processes. We claim that, by choosing those measures, our results will be applicable not only to graph's properties but also to clustering and community detection processes. The first graph structural measure

Table 1: Datasets used in this study. For each network we present the number of vertices (n), number of edges (m), average degree ($\overline{deg}$), average distance ($\overline{dist}$) and diameter (d).

Dataset	n	m	$\overline{deg}$	$\overline{dist}$	d
POLBLOGS	1,490	19,022	25.53	3.39	9
UC-IRVINE	1,899	20,296	21.37	3.19	8
WIKI-VOTE	7,115	103,689	29.14	3.34	10
DBLP-CITE	12,591	49,728	7.89	5.42	20
EPINIONS	75,879	508,837	13.41	4.75	16

is the *average distance* ($\overline{dist}$), which is defined as the average of the distances between each pair of vertices in the graph. *Diameter* (d) is defined as the largest minimum distance between two vertices in the graph, and *edge intersection* is the percentage of original arcs which are also present in the perturbed version of the graph, i.e. $EI(G,\widetilde{G}) = \frac{|A \cap \widetilde{A}|}{max(|A|,|\widetilde{A}|)}$. The above measures evaluate the entire graph as a unique score. We compute the error on these graph metrics as follows:

$$\epsilon_m(G,\widetilde{G}) = |m(G) - m(\widetilde{G})| \tag{8}$$

where m is one of the graph metrics defined above, G is the original graph and $\widetilde{G}$ is the k-anonymous graph.

The following metrics evaluate specific structural properties for each vertex of the graph: the first one is *betweenness centrality* (C_B), which measures the fraction of the shortest paths that go through each vertex. The second one is *closeness centrality* (C_C) and it measures how many steps are required to access every other vertex from a given vertex. We refer to C_C^- when the in-degree is considered and C_C^+ in case of considering the out-degree. Finally, we use the *degree centrality* (C_D), which evaluates the centrality of each vertex based on its degree, i.e., the fraction of vertices connected to it. Similarly, C_D^- refers to in-degree and C_D^+ to the out-degree of each vertex. We compute the error on vertex metrics by:

$$\epsilon_m(G,\widetilde{G}) = \sqrt{\frac{1}{n}\big((g_1 - \widetilde{g}_1)^2 + \ldots + (g_n - \widetilde{g}_n)^2\big)}, \tag{9}$$

where g_i and $\widetilde{g}_i$ are the values of the metric m for vertex v_i of G and v_i of $\widetilde{G}$ respectively.

5.3 Clustering-specific evaluation

Variations in the generic graph properties is a good way to assess the information loss but they have their limitations because they are just a proxy for the changes in data utility we actually want to measure. We define the specific information loss measures as a task-specific measure for quantifying the data utility and the information loss associated to a data publishing process. We focus on clustering-specific processes, due to their importance in networks arising from diverse applications, including social, biological and healthcare networks. Similar to generic graph measures, we compare the results obtained both by the original and the perturbed

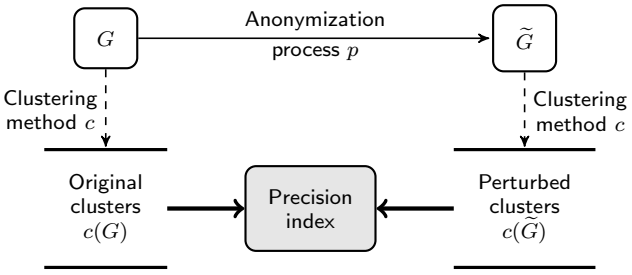


Fig. 4: Framework for evaluating the clustering-specific information loss measure.

data in order to quantify the level of noise introduced in the perturbed data. This measure is specific and application-dependent, but it is necessary to test the perturbed data in real graph-mining processes.

We consider the following approach to measure the clustering assessment for a particular perturbation and clustering method: (1) apply our k-degree anonymity algorithms to the original graph G and obtain $\widetilde{G}$; (2) apply a particular clustering method c to G and obtain clusters $c(G)$ and then apply the same method to $\widetilde{G}$ to obtain $c(\widetilde{G})$; (3) compare the clusters $c(G)$ to $c(\widetilde{G})$ as shown in Figure 4. With respect to information loss, it is clear that the more similar $c(\widetilde{G})$ is to $c(G)$, we have the less information loss. Thus, clustering specific information loss measures should evaluate the divergence between both sets of clusters $c(G)$ and $c(\widetilde{G})$.

Ideally, the results should be the same, that is, the same number of sets (i.e., clusters) with the same elements in each set. In this case, we can say that the anonymization process has not affected the clustering process. When the sets do not match, we should be able to calculate a measure of divergence. For this purpose, we use the *precision index* [5]. Assuming that we know the true communities of a graph, the precision index can directly be used to evaluate the similarity between two cluster assignments. Given a graph of n nodes and q true communities, we assign to nodes the same labels $l_{tc}(\cdot)$ as the community they belong to. In our case, the true communities are the ones assigned on the original dataset (i.e., $c(G)$), since we want to obtain communities as close as the ones we would get on non-anonymized data – we are not interested in the ground truth communities. Assuming that the perturbed graph has been divided into clusters (i.e., $c(\widetilde{G})$), then for every cluster, we examine all the nodes within it and assign to them as predicted label $l_{pc}(\cdot)$ the most frequent true label in that cluster (basically the mode). Then, the precision index can be defined as follows:

$$precision(G, \widetilde{G}) = \frac{1}{n} \sum_{i=1}^{n} \delta_{l_{tc}(v_i), l_{pc}(v_i)}, \tag{10}$$

where δ is the Kronecker delta function, i.e., $\delta_{x,y}$ equals 1 if $x = y$ and 0 otherwise. Note that the precision index is a value in the range $[0, 1]$, which takes the value 0 when there is no overlap between the sets and the value 1 when the overlap between the sets is complete.

We have considered two different graph clustering algorithms to evaluate the anonymization process. Both them are unsupervised algorithms based on different

concepts and developed for different applications and scopes. The selected clustering algorithms are: (1) *Infomap* [33] that optimizes the map equation, which exploits the information-theoretic duality between the problem of data compression and the one of detecting significant structures in the graph; and (2) *Walktrap* [30] that finds densely connected subgraphs via random walks.

6 Information Loss and Data Utility

In this section, we present the results of our anonymization techniques in terms of data utility and information loss. Generic information loss measures, which are based on several graph's properties, will be described in the next section, while information loss regarding clustering-specific graph-mining tasks will be analyzed in subsequent section.

6.1 Generic information loss evaluation

In this subsection, we compare the results of anonymizing several networks using our models and algorithms for k-degree anonymity on directed networks. Specifically, we will use DGA for Paired and Independent k-degree anonymity. We apply both algorithms on the same data with the same parameters and compare the results in terms of information loss and data utility. It is important to note that the privacy level achieved for both algorithms is similar, but not the same. As we have discussed previously, Paired k-degree anonymity is a stronger model than the one of Independent k-degree anonymity. However, the former or the latter method could be of interest depending on the dataset and the application scenario. Unfortunately, we cannot compare our methods to other k-degree anonymity algorithms, due to the fact that our work is the first that considers k-degree anonymity models specifically designed for directed networks.

Firstly, an in-depth analysis of generic information loss measures on DBLP-CITE network will is performed. We propose a detailed study of how generic information loss measures evolves during the anonymization process. Then, we present the same analysis on the other four networks, but skipping details due to the space constraints.

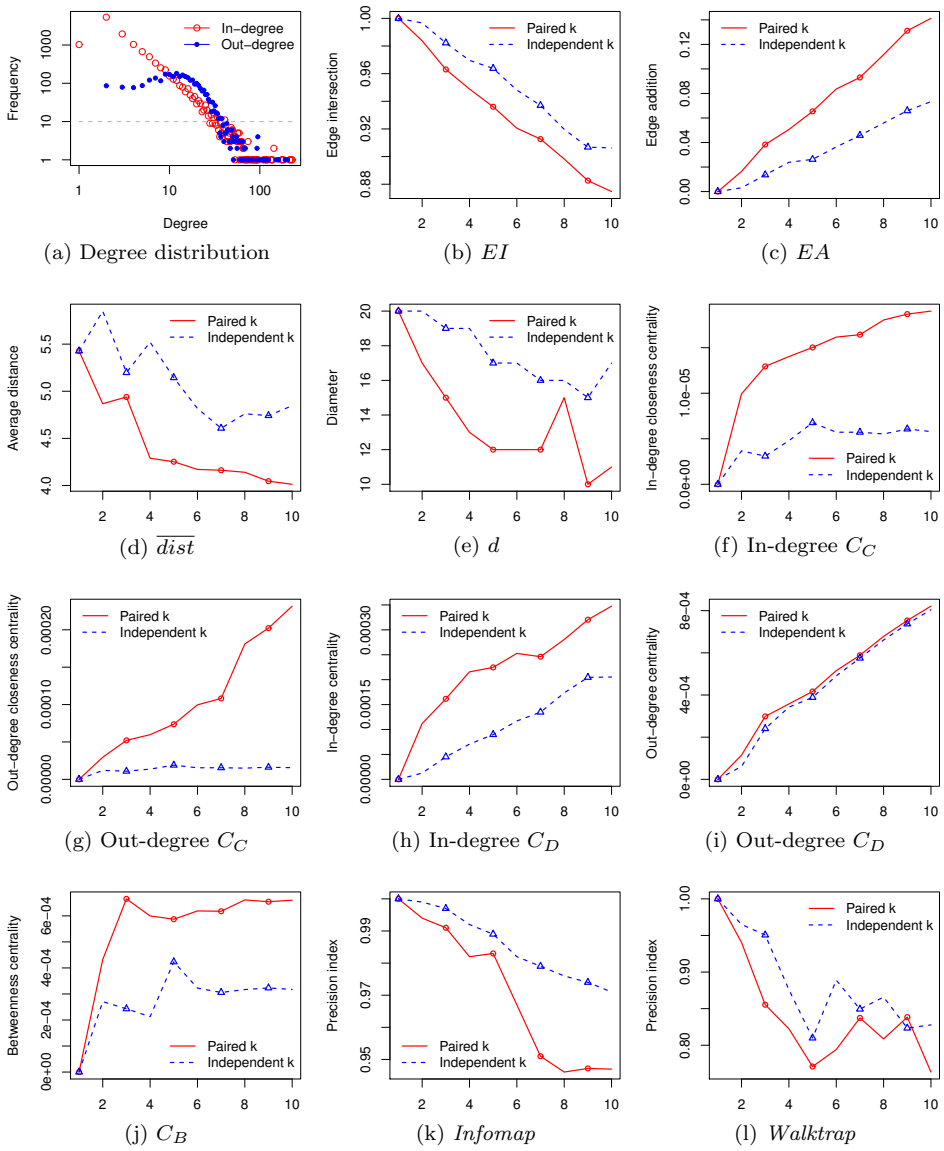
The results are shown in Table 2. Each row indicates the scored value for the corresponding measure and method, and each column corresponds to an experiment with a different k-anonymity value. For each dataset and method, we vary k from 1 to 10 (k=1 corresponds to the original dataset) and compare the results obtained on EI, EA, $\overline{dist}$, d, C_B, C_C^-, C_C^+, C_D^- and C_D^+. The last column corresponds to the average error $\overline{\epsilon_m}$. Each characteristic is reported two times, corresponding to Paired and Independent k-degree anonymity. Clearly, the lower the information loss, the better the method. Perfect performance in a row would be indicated by achieving exactly the same score as in the original network (the k=1 column). Although deviation is undesirable, it is inevitable due to the edge modification process. Complementary information is introduced in Figure 5, where we can see graphical details about the behaviour of different models during anonymization process (those are the same values that we have reported in Table 2). Addition-

Table 2: Results for Paired k (P-k) and Independent k (I-k) degree anonymity over 10 levels of anonymization on DBLP-CITE dataset. For each method, we compare the results obtained on *edge intersection* (EI), *edge addition* (EA), *average distance* ($\overline{dist}$), *diameter* (d), *betweenness centrality* (C_B), *closeness centrality* based on the in-degree (C_C^-) and out-degree (C_C^+), *in-degree* (C_D^-) and *out-degree* centrality (C_D^+) and *precision index* using Infomap and Walktrap clustering algorithms. The last column corresponds to the *average error* ($\overline{\epsilon_m}$) for generic information loss measures (rows 1 to 9) or average precision score ($\overline{p}$) for clustering information loss measures (rows 10 and 11).

Metric	**Model**	$k=1$	2	3	4	5	6	7	8	9	10	$\overline{\epsilon_m}$
EI (%)	P-k	1	0.983	0.963	0.948	0.936	0.920	0.912	0.898	0.882	0.874	0.067
	I-k		0.996	0.982	0.969	0.963	0.948	0.936	0.919	0.906	0.906	0.046
EA (%)	P-k	0	1.641	3.837	5.060	6.540	8.361	9.305	11.165	13.115	14.121	7.314
	I-k		0.308	1.359	2.385	2.626	3.638	4.579	5.590	6.590	7.336	3.441
$\overline{dist}$	P-k	5.427	4.867	4.939	4.289	4.252	4.170	4.161	4.140	4.045	4.012	0.996
	I-k		5.850	5.197	5.521	5.143	4.817	4.607	4.760	4.742	4.846	0.439
d	P-k	20	17	15	13	12	12	12	15	10	11	6.3
	I-k		20	19	19	17	17	16	16	15	17	2.4
$C_B(e^{-4})$	P-k	0	4.324	6.653	5.998	5.871	6.190	6.177	6.613	6.543	6.603	5.497
	I-k		2.706	2.429	2.126	4.235	3.225	3.057	3.168	3.231	3.173	2.735
$C_C^-(e^{-5})$	P-k	0	0.993	1.294	1.402	1.502	1.616	1.643	1.804	1.868	1.902	1.402
	I-k		0.366	0.309	0.478	0.676	0.572	0.571	0.552	0.607	0.579	0.471
$C_C^+(e^{-5})$	P-k	0	2.955	5.211	5.974	7.366	9.963	10.816	18.101	20.226	23.162	10.377
	I-k		1.198	1.088	1.370	1.894	1.546	1.548	1.498	1.621	1.556	1.332
$C_D^-(e^{-4})$	P-k	0	1.119	1.615	2.157	2.246	2.527	2.461	2.805	3.203	3.475	2.161
	I-k		0.125	0.450	0.707	0.897	1.171	1.348	1.734	2.046	2.053	1.053
$C_D^+(e^{-4})$	P-k	0	1.148	2.984	3.576	4.165	5.154	5.883	6.780	7.534	8.213	4.544
	I-k		0.612	2.403	3.423	3.884	4.906	5.748	6.607	7.369	8.042	4.299
Clustering	**Model**	$k=1$	2	3	4	5	6	7	8	9	10	$\overline{p}$
Infomap	P-k	1	0.994	0.991	0.982	0.983	0.967	0.951	0.946	0.947	0.946	0.970
	I-k		0.999	0.997	0.992	0.989	0.982	0.979	0.976	0.974	0.971	0.985
Walktrap	P-k	1	0.940	0.855	0.822	0.770	0.794	0.837	0.808	0.838	0.763	0.843
	I-k		0.965	0.950	0.875	0.809	0.889	0.849	0.865	0.823	0.827	0.885

ally, information regarding degree distribution is provided in Figure 5a, where several nodes do not fulfill k-degree anonymity both on in- and out-degree.

The first two rows in Table 2 correspond to edge intersection and edge addition. Edge intersection is the percentage of edges on the anonymous networks which are also present in the original network. Figure 5b shows that this metric is linear on the k value on both methods. It is important to underline that more than 90% of the arcs in $\widetilde{A}$ are also present in A. The next metric is closely related to this one and the behaviour is similar, as depicted in Figure 5c. Edge addition indicates the number of arcs added to anonymize the network. The difference between these metrics rely on the edge switch and edge extension operations, which can modify some arcs to fulfill the anonymous degree sequences. We note that usually the number of arcs added by Independent k-degree is half the number of Paired k-degree. Average distance is pointed out in the third row. The value



(a) Degree distribution (b) EI (c) EA

(d) $\overline{dist}$ (e) d (f) In-degree C_C

(g) Out-degree C_C (h) In-degree C_D (i) Out-degree C_D

(j) C_B (k) *Infomap* (l) *Walktrap*

Fig. 5: Generic and clustering information loss evaluation on Paired k and Independent k-degree anonymity on DBLP-CITE dataset.

of the original network (denoted by $k = 1$) is 5.427. Thus, values close to this one indicate low information loss. Although both methods achieve good results, values of Independent k-degree anonymity remain closer to the original one over all anonymization range than values of Paired k-degree, as can be seen in Figure 5d. Indeed, the average error computed over all range is 0.996 for Paired k-degree

and 0.439 for Independent k-degree. A similar behaviour can be seen within the second metric, diameter (see Figure 5e).

The centrality measures are computed by Equation 9, as described previously. Therefore, the values in Table 2 are 0 for the original network, i.e. k=1. Clearly, the lower the value, the better the performance of the corresponding method. Betweenness centrality is an important measure for some clustering and community detection algorithms. We remark that the error remains almost stable for values of $k \geq 3$, as can be seen in Figure 5j. Figures 5f and 5g depict the in- and out-degree closeness centrality, respectively. We can observe that these values remain low, except from the out-degree values of Paired k-degree anonymity which present slightly high values. Finally, the in- and out-degree centrality measures, which are depicted in Figures 5h and 5i, indicate very similar values and behaviour, independently of the method used to anonymize the network. We should note here that the results shown in Fig. 5 and Table 2 correspond to a single run of the proposed models. As we can observe, some data utility criteria are not monotonous with respect to the anonymization level k. For example, in Fig. 5e, the diameter of the graph presents a spike at $k = 8$, which turns out to be an effect of the edge modification process (edge switch and edge extension).

Table 3 presents the same analysis on the other tested networks. However, due to space constraints, only the average error (last column of Table 2) is pointed out for each metric, method and network. POLBLOGS is a medium size network with some important hubs. Hence, anonymization process is harder than other networks, adding an average of 19% of total arcs to fulfill Paired k-degree anonymity. When Independent K-degree is considered, less than 5% of arcs have to be added. Nonetheless, the average distance and diameter show relatively small distortion after the anonymization process. On the contrary, EPINIONS is our largest network and the results are very encouraging. Less than 6% of the total number of arcs need to be added to fulfill Paired k-degree anonymity. The same value reduces to less than 2% in case of Independent k-degree anonymity. Moreover, the average distance and diameter show very small perturbation. Indeed, there is no perturbation in diameter using Independent k-degree anonymity.

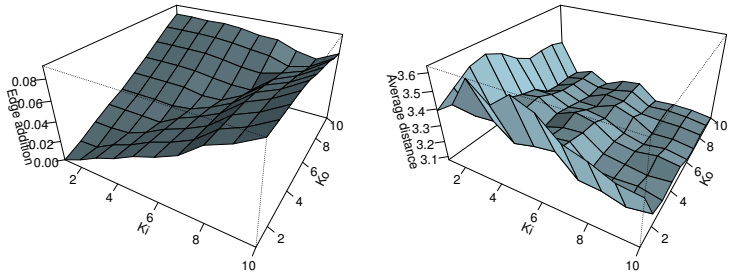
In the previous paragraph we have considered and compared Independent k-degree when $k = k_i = k_o$. This is an specific case, but also the most probable, interesting and useful one. Nevertheless, we want to analyze the general case, i.e. Independent (k_i, k_o)-degree anonymity where $k_i \neq k_o$. In the following experiments, we will consider all possible combinations of $k_i, k_o \in [1, \ldots, 10]$ on POLBLOGS, which implies 100 anonymous datasets. Note that, 10 of these datasets are the same in Independent k-degree anonymity.

Results of Independent (k_i, k_o)-degree anonymity are depicted in Table 3 (third row) and in Figure 6. As it can be seen, the average error of Independent (k_i, k_o)-degree anonymity on generic information loss measures remains higher than Independent k but lower than Paired k. It is interesting to underline that the number of edges added when $k_i \approx 10$ or $k_o \approx 10$ is very similar to the one when $k_i, k_o \approx 10$, but the privacy level is not. Indeed, the best ratio between number of arcs added and privacy level is achieved when $k_i = k_o$, as can be seen in Figure 6a. The average distance (Figure 6b) also shows that the error is greater when considering very different values of k_i and k_o.

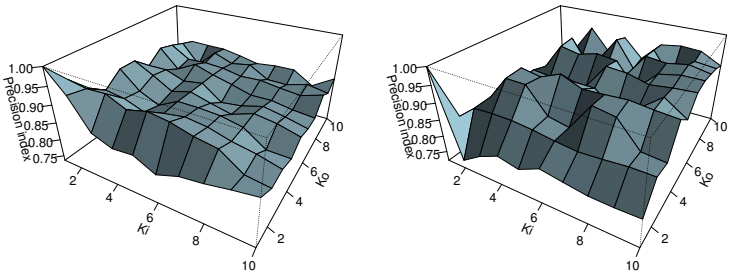
Finally, we have considered a baseline comparison to our methods. It is a naïve approach based on converting the digraphs to undirected graphs, applying a k-

Table 3: Results for Paired k (P-k) and Independent k (I-k) degree anonymity over 10 levels of anonymization, and Independent (k_i, k_o) (I-(k_i, k_o)) degree anonymity over 100 levels of anonymization. For each dataset and method, we compare the results obtained on *edge intersection* (EI), *edge addition* (EA), *average distance* ($\overline{dist}$), *diameter* (d), *betweenness centrality* (C_B), *closeness centrality* based on the in-degree (C_C^-) and out-degree (C_C^+), *in-degree* (C_D^-) and *out-degree* centrality (C_D^+) and *precision index* using Infomap (IM) and Walktrap (WT) clustering algorithms.

Network	**Model**	EI	EA	$\overline{dist}$	d	C_B	C_C^-	C_C^+	C_D^-	C_D^+	IM	WT
Polblogs	P-k	0.160	19.45%	0.484	1.5	$2.50e^{-3}$	$7.33e^{-4}$	$4.50e^{-4}$	$7.14e^{-3}$	$5.76e^{-3}$	0.835	0.882
	I-k	0.064	4.26%	0.180	0.1	$1.33e^{-3}$	$1.47e^{-4}$	$1.02e^{-4}$	$3.38e^{-3}$	$4.37e^{-3}$	0.930	0.925
	I-(k_i, k_o)	0.067	5.40%	0.186	0.7	$1.76e^{-3}$	$1.63e^{-2}$	$1.37e^{-2}$	$3.20e^{-3}$	$4.23e^{-3}$	0.901	0.885
	U-k	0.395	68.16%	0.594	1.1	$2.90e^{-3}$	$1.60e^{-3}$	$1.68e^{-3}$	$1.09e^{-2}$	$1.64e^{-2}$	0.744	0.723
UC-Irvine	P-k	0.098	11.27%	0.113	0.6	$5.53e^{-4}$	$1.50e^{-4}$	$9.65e^{-3}$	$1.76e^{-3}$	$2.58e^{-3}$	0.951	0.710
	I-k	0.022	2.19%	0.023	0.0	$3.33e^{-4}$	$0.79e^{-4}$	$5.51e^{-3}$	$0.74e^{-3}$	$1.15e^{-3}$	0.950	0.785
Wiki-vote	P-k	0.078	8.78%	0.018	1	$5.68e^{-4}$	$2.54e^{-4}$	$3.94e^{-5}$	$9.75e^{-4}$	$1.18e^{-3}$	0.683	0.786
	I-k	0.024	1.88%	0.049	0	$1.29e^{-4}$	$0.07e^{-4}$	$0.11e^{-5}$	$4.81e^{-4}$	$0.82e^{-3}$	0.796	0.709
Epinions	P-k	0.056	5.96%	0.200	1.3	$6.12e^{-5}$	$4.26e^{-6}$	$2.79e^{-6}$	$1.73e^{-4}$	$7.66e^{-5}$	0.846	0.675
	I-k	0.026	1.94%	0.016	0.0	$1.74e^{-5}$	$0.90e^{-6}$	$0.41e^{-6}$	$1.63e^{-4}$	$6.24e^{-5}$	0.901	0.702



(a) EA (b) $\overline{dist}$



(c) *Infomap* (d) *Walktrap*

Fig. 6: Independent (k_i, k_o)-degree anonymity on Polblogs dataset.

degree anonymity algorithm (with k in the same range as the other methods) and transforming again to directed graphs to make the comparison fair. For this

analysis we have used the UMGA algorithm [9], which demonstrated to preserve data utility better than other k-degree-based methods.

We named this method Undirected k-degree anonymity (U-k), and its results are depicted in Table 3 (fourth row). It is important to underline that the privacy level achieved is the same as in the Paired k-degree anonymity, since the in- and out-degree of the anonymous graphs are the same. However, the error values are higher than the ones for Paired k on all metrics, except the diameter.

6.2 Clustering information loss evaluation

As we have stated previously, clustering-specific information loss measures are important to consider data utility and information loss on real graph-mining processes. Even though the generic information loss measures are a good way to assess the data utility, specific information loss measures can help us to quantify data utility and information loss associated to a data publishing process. The last two rows in Table 2 and the last two columns in Table 3 present the precision index computed using the Infomap and Walktrap algorithms. As we have previously commented, the precision index takes the value of zero when there is no overlap between the sets and the value of one when the overlap between the sets is complete.

Analyzing our in-depth experiment on the DBLP-CITE network, we can point out that more than 94% (Paired k) and 97% (Independent k) are correctly clustered using Infomap after $k = 10$ anonymization process, as shown in Figure 5k. Precision index average values are 0.97 and 0.985, respectively. Similar results can be seen for our other tested networks in Table 3. Precision index on Independent (k_i, k_o)-degree anonymity on Infomap can be seen in Figure 6c. As in the previous experiments, Infomap seems to be more stable and less sensitive to data perturbation. The average precision index keeps close to 90% well-classified vertices using both clustering algorithms. These results allow us to claim that data utility is preserved using our methods to anonymize directed networks.

Lastly, it is interesting to point out that the precision index achieved using an undirected k-degree approach (U-k) is far worse than the methods specifically developed to deal with directed networks when considering the clustering-specific information loss.

Summarizing the results, it is interesting to stress out that both methods achieve good results on generic and clustering-specific information loss measures. Specifically, Independent k-degree anonymity gets the lowest average error on all analyzed metrics and datasets, and the highest precision index values on all clustering algorithms. It is also important to underline that although Paired k-degree anonymity imposes the strongest privacy levels, it achieves very good results on all analyzed metrics.

7 Performance and Scalability

In this section, we aim to improve the scalability of the proposed methods towards being able to anonymize large-scale directed networks. To this direction,

Table 4: Large datasets used to test the scalability of our methods. For each network we present the number of vertices (n), number of edges (m), independent (k_i,k_o)-degree anonymity and paired k-degree anonymity.

Dataset	n	m	(k_i, k_o)	k
DBLP-2006	484,161	1,422,263	(1,1)	1
Pokec	1,632,803	30,622,564	(1,1)	1

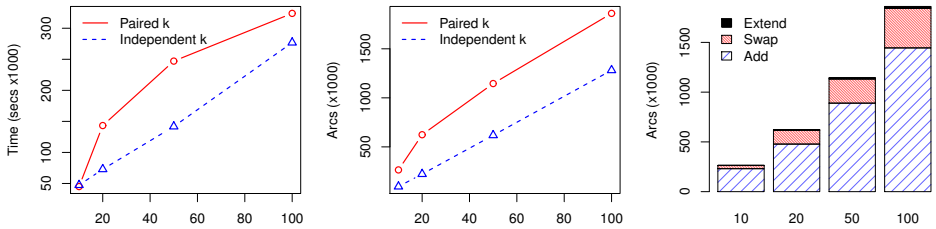
Table 5: Results for Paired k (P-k) and Independent k (I-k) degree anonymity for $k \in \{10, 20, 50, 100\}$. For each dataset, method and k value, we present the main values for each step of our method: computation time (in sec.) and number of new arcs for degree sequence anonymization; and number of edge addition, edge switch and edge extension, and computation time (in seconds) for graph modification process.

Network	**Model**		**Deg. seq. anon.**		**Graph modification**			
	Meth.	k	Time (s)	Arcs	Add	Switch	Extend	Time (s)
DBLP-2006	P-k	10	13	18,107	18,106	1	0	15
		20	15	33,915	33,915	0	0	44
		50	105	70,365	70,286	68	11	118
		100	457	130,447	130,417	10	20	326
	I-k	10	81	1,143	956	187	0	12
		20	65	2,885	2,346	539	0	30
		50	48	8,053	6,769	1,284	0	118
		100	39	18,485	14,971	3,514	0	298
Pokec	P-k	10	503	265,095	230,875	31,675	2,545	44,502
		20	357	623,930	477,875	138,646	7,409	142,926
		50	400	1,145,660	889,369	243,467	12,824	246,668
		100	456	1,861,159	1,444,475	396,106	20,578	323,473
	I-k	10	60	95,281	53,659	39,209	2,413	47,307
		20	73	222,653	54,731	166,545	1,377	72,807
		50	604	618,259	233,867	383,892	500	141,137
		100	1,880	1,281,275	517,036	763,910	329	274,976

we have applied two preprocessing techniques for obtaining the k-anonymous degree sequences. The first one, which is tailored for the Independent (k_i,k_o)-degree anonymity model, is based on a lossless representation of the degree sequences d_{in}, d_{out} with a considerable reduction in size [35].

For the case of Paired k-degree anonymity, since the microaggregation technique is not scalable (e.g., the MDAV algorithm has time complexity proportional to $O(n^2)$), we have applied k-means as a partitioning method (preprocessing step of $O(n)$ complexity). In particular, we have used Lloyd's algorithm in a hierarchical way to obtain partitions of manageable size.

More precisely, we start by applying k-means to obtain a partition for the entire data set. If the parts are not small enough (smaller than a threshold s), we further apply k-means on each of them, until we satisfy the required threshold. Notice that our solution could have been executed in parallel, yielding an even faster algorithm in practice. The method is presented in Algorithm 2.



(a) Computation time (b) New arcs (c) Detail of new arcs

Fig. 7: Details of our experiments on Pokec network. The horizontal axis represents the k-degree anonymity value.

Function `graph_partition_process`
  Input: $(\delta_{in}, \delta_{out})$, $threshold = s$
  Output: Partition $\mathcal{C}$
  Apply k-means clustering to obtain clusters $\mathcal{C} = C_1, \ldots, C_{m_0}$
  while $|C_i| > s$ *for some* $C_i \in \mathcal{C}$ **do**
    for $C_1 : |C_1| = max\{|C_i| : C_i \in \mathcal{C}\}$ **do**
      Apply k-means clustering to obtain clusters $\mathcal{C}_1 = C_{11}, \ldots, C_{1m_1}$
      Update $\mathcal{C} = (\mathcal{C} \setminus C_1) \cup \mathcal{C}_1$
    end
  end
  return $\mathcal{C}$
end
Function `MDAV_parallel`
  Input: Partition $\mathcal{C}$
  Output: Anonymized sequence $(\delta^k_{in}, \delta^k_{out}) = \mathcal{P}$
  for $C_i \in \mathcal{C}$ **do**
    Apply MDAV to obtain k-anonymous subsequences P_i for each C_i
  end
  return $\mathcal{P} = \cup P_i$
end

Algorithm 2: Scalable Paired k-degree anonymity

In order to examine the scalability of our methods, we have used two large scale real networks. The first one is the DBLP-2006[2], which corresponds to the co-authorship network of the DBLP computer science bibliography in 2006. The second one, Pokec [38], is the most popular online social network in Slovakia. Table 4 provides the main properties of these networks. All the experiments reported in this section have been performed on a 4 CPU Intel Xeon X3430 at 2.40GHz with 32GB RAM, running Debian GNU/Linux.

Table 5 depicts the results of the scalability experiments. For each network and method, we have considered values of $k \in \{10, 20, 50, 100\}$. As a summary of the first step of our method, we provide the computation time (in secs.) and the number of new arcs that is needed to create a k-degree anonymous sequence. Regarding the second step of our method, we report the computation time, as well as the number of edge addition, switch and extension that is performed.

[2] DBLP Bibliography Server: http://dblp.uni-trier.de/xml/

As it can be seen in Table 5, the computation time of degree sequence anonymization (step 1) is negligible compared to the one of graph modification process (step 2). Consequently, the Paired k model is more time-consuming compared to the Independent k model, mainly due to the fact that the Paired k involves the insertion of more new edges in order to reach the desired privacy level. Figure 7a also shows that the total running time grows linearly with respect to the value of k. Additionally, the number of arcs that need to be added also grows linearly with the value of k, as shown in Figure 7b. Finally, as depicted in Figure 7c, the number of edge additions, swaps and extensions grows proportionally while the value of k increases.

However, the computation time needed by the Paired and Independent k anonymity models is quite similar in the DBLP-2006 graph. Although the number of arcs that need to be created is much lower in the Independent k model, the number of edge switch operations is higher; edge switches, along with edge extensions, are more time-consuming compared to edge additions which are computationally easy to be performed.

Finally, regarding data utility and information loss, we underline that the preprocessing technique on Independent (k_i,k_o)-degree anonymity model preserves the quality of the solution, as demonstrated by authors in [35]. On the contrary, the preprocessing technique on Paired k-degree anonymity can slightly reduce the quality of the solution compared to the case where no preprocessing step is applied. We measure this divergence as the number of added extra arcs to the k-anonymous degree sequences, which is between 0.1% and 1.5% according to our experiments on DBLP-2006. Specifically, the values are 0.13%, 0.33%, 0.93% and 1.57% for $k \in \{10, 20, 50, 100\}$.

8 Conclusions

In this paper, we have defined two different k-degree anonymity models specifically designed for directed networks. Furthermore, we have introduced different algorithms to achieve the desired privacy levels, based on the proposed models. An empirical evaluation of these models have been conducted on several real networks, comparing information loss based on different graph properties and also on clustering-specific criteria. We have demonstrated that our anomymization models aim to reduce information loss, while simultaneously retain data utility. As we have seen throughout our experimental framework, the Independent k-degree anonymity model demands fewer edge additions and switches in order to meet the desired privacy level. Nevertheless, the Paired k-degree model gives good results in several generic information loss measures and also achieves excellent precision index values in our clustering-specific information loss framework. Furthermore, we have demonstrated that our edge modification technique is scalable to large scale networks.

Many interesting directions for future research have been uncovered by this work. Firstly, a deeper analysis of the Independent (k_i,k_o)-degree anonymity model have to be performed in order to better understand how these parameters can be used according to network's specific properties in order to achieve good privacy levels, while preserving the underlying graph structure. Secondly, it would be thought-provoking to also consider edge deletion in order to better preserve data

utility. Moreover, other information loss measures based on real graph-mining processes can be considered, such as information flow. Lastly, it would be also very interesting to extend those models to other rich graph types, including weighted, signed and multilayer [32] directed networks.

Acknowledgements Jordi Casas-Roma was partially supported by the Spanish MCYT and the FEDER funds under grants TIN2011-27076-C03 "CO-PRIVACY" and TIN2014-57364-C2-2-R "SMARTGLACIS". Julián Salas acknowledges the support of a UOC postdoctoral fellowship and TIN2014-57364-C2-2-R "SMARTGLACIS".

References

1. L. Backstrom, C. Dwork, and J. Kleinberg, "Wherefore art thou r3579x?: anonymized social networks, hidden patterns, and structural steganography" in WWW '07. New York, NY, USA: ACM, pp. 181–190, 2007.
2. S. Bhagat, G. Cormode, B. Krishnamurthy, and D. Srivastava, "Class-based graph anonymization for social network data" in VLDB '09, vol. 2, no. 1, pp. 766–777, 2009.
3. R. Bredereck, V. Froese, M. Koseler, M.G. Millani, A. Nichterlein, and R. Niedermeier, "A Parameterized Algorithmics Framework for Degree Sequence Completion Problems in Directed Graphs". In Proceedings of the 11th International Symposium on Parameterized and Exact Computation (IPEC '16), pp. 10:1–10:14. Schloss Dagstuhl - Leibniz-Zentrum fr Informatik, 2017.
4. R. Bredereck, V. Froese, M. Koseler, M.G. Millani, A. Nichterlein, and R. Niedermeier, "A Parameterized Algorithmics Framework for Digraph Degree Sequence Completion Problems". arXiv:1604.06302v3, 2018.
5. B.J. Cai, H.Y. Wang, H.R. Zheng, and H. Wang, "Evaluation repeated random walks in community detection of social networks" in ICMLC '10. Qingdao, China: IEEE, pp. 1849–1854, 2010.
6. J. Casas-Roma, J. Herrera-Joancomartí, and V. Torra, "An Algorithm For k-Degree Anonymity On Large Networks" in ASONAM '13. Niagara Falls, ON, Canada: IEEE, pp. 671–675, 2013.
7. J. Casas-Roma, J. Herrera-Joancomartí, and V. Torra, "Anonymizing graphs: measuring quality for clustering". Knowledge and Information Systems, vol. 44, no. 3, pp. 507–528, 2015.
8. J. Casas-Roma, "An Evaluation of Edge Modification Techniques for Privacy-Preserving on Graphs" in MDAI '15. Skövde, Sweden: Springer, 2015, pp. 180–191.
9. 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.
10. S. Chester, B.M. Kapron, G. Ramesh, G. Srivastava, A. Thomo, and S. Venkatesh, "Why Waldo befriended the dummy? k-Anonymization of social networks with pseudo-nodes". Social Network Analysis and Mining, vol. 3, no. 3, pp. 381–399, 2013.
11. K.L. Clarkson, K. Liu, and E. Terzi, "Towards identity anonymization in social networks". In Link Mining: Models, Algorithms, and Applications, pp. 359–385. Springer, 2010.
12. G. Cormode, D. Srivastava, T. Yu, and Q. Zhang, "Anonymizing bipartite graph data using safe groupings". The VLDB Journal, vol. 19, no. 1, pp. 115–139, 2010.
13. S. Das, Ö. Egecioglu, and A. Abbadi, "Anonymizing weighted social network graphs" in ICDE '10. Long Beach, CA, USA: IEEE, 2010, pp. 904–907.
14. J. Domingo-Ferrer, and V. Torra, "Ordinal, continuous and heterogeneous k-anonymity through microaggregation". Data Mininig Knowledge Discovery, vol. 11, no. 2, pp. 195–212, 2005.
15. J. Domingo-Ferrer, J.M. Mateo-Sanz, "Practical data-oriented microaggregation for statistical disclosure control". IEEE Trans. Knowl. Data Eng., vol. 14, no. 1, pp. 189–201, 2002.
16. F. Ferri, P. Grifoni, and T. Guzzo, "New forms of social and professional digital relationships: the case of Facebook". Social Network Analysis and Mining, vol. 2, no. 2, pp. 121–137, 2011.

17. S. Hanhijärvi, G.C. Garriga, and K. Puolamäki, "Randomization techniques for graphs" in SDM '09. Sparks, Nevada, USA: SIAM, 2009, pp. 780–791.
18. S.L. Hansen and S. Mukherjee, "A Polynomial Algorithm for Optimal Univariate Microaggregation". IEEE Trans. Knowl. Data Eng., vol. 15, no. 4, pp. 1043–1044, 2003.
19. S. Hartung, C. Hoffmann, and A. Nichterlein, "Improved upper and lower bound heuristics for degree anonymization in social networks" in SEA '14. Copenhagen, Denmark: Springer, pp. 376–387, 2014.
20. S. Hartung, A. Nichterlein, R. Niedermeier, and O. Suchý, "A refined complexity analysis of degree anonymization in graphs". Information and Computation, vol. 243, pp. 249–262, 2015.
21. M. Hay, G. Miklau, D. Jensen, D. Towsley, and P. Weis, "Resisting structural re-identification in anonymized social networks" in VLDB '08, vol. 1, no. 1, pp. 102–114, 2008.
22. L.A. Adamic, and N. Glance, "The Political Blogosphere and the 2004 US Election: Divided they Blog" in LinkKDD '05. Chicago, Illinois, USA: ACM, 2005, pp. 36–43.
23. T. Opsahl, and P. Panzarasa, "Clustering in weighted networks". Social Networks, vol. 31, no. 2, pp. 155–163, 2009.
24. L. Liu, J. Wang, J. Liu, and J. Zhang, "Privacy Preservation in Social Networks with Sensitive Edge Weights" in SDM '09. Sparks, Nevada, USA: SIAM, pp. 954–965, 2009.
25. K. Liu, and E. Terzi, "Towards identity anonymization on graphs" in SIGMOD '08. New York, NY, USA: ACM, pp. 93–106, 2008.
26. A. Lancichinetti and S. Fortunato. (2009), "Community detection algorithms: a comparative analysis". Physical Review E, vol. 80, no. 5, art. 56117, 2009.
27. M. Ley. "The DBLP computer science bibliography: Evolution, research issues, perspectives" in SPIRE 2002, Springer-Verlag London, 2002, pp. 1–10.
28. J. Leskovec, D. Huttenlocher and J. Kleinberg. "Predicting Positive and Negative Links in Online Social Networks" in WWW '10. Raleigh, North Carolina, USA: ACM, 2010, pp. 641–650.
29. X. Lu, Y. Song, and S. Bressan, "Fast Identity Anonymization on Graphs" in DEXA '12. Vienna, Austria: Springer, 2012, pp. 281–295.
30. P. Pons and M. Latapy, "Computing communities in large networks using random walks". Lecture Notes in Computer Science (LNCS), vol 3733, pp 284–293, 2005.
31. M. Richardson, R. Agrawal and P. Domingos. "Trust Management for the Semantic Web" in ISWC. 2003, pp 351–368.
32. L. Rossi, M. Musolesi and A. Torsello. "On the k-Anonymization of Time-Varying and Multi-Layer Social Graphs" in ICWSM. 2015, pp 377–386.
33. M. Rosvall, and C.T. Bergstrom, "Maps of random walks on complex networks reveal community structure". Proceedings of the National Academy of Sciences of the United States of America, vol. 105, no. 4, pp. 1118–1123, 2008.
34. J. Salas and V. Torra. "Improving the characterization of P-stability for applications in network privacy", Disc. Appl. Math., vol. 206, pp. 109–114, 2016.
35. J. Salas, "Faster univariate microaggregation for power law distributions: k-degree-Anonymity for big graphs" in ICDM Workshop on Privacy and Discrimination in Data Mining. Barcelona, Spain: IEEE, 2016.
36. P. Samarati, "Protecting Respondents Identities in Microdata Release". IEEE Trans. Knowl. Data Eng., vol. 13, no. 6, pp. 1010–1027, 2001.
37. L. Sweeney, "k-anonymity: a model for protecting privacy". International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, vol. 10, no. 5, pp. 557–570, 2002.
38. L. Takac, M. Zabovsky, "Data Analysis in Public Social Networks" in International Scientific Conference and International Workshop Present Day Trends of Innovations. Lomza, Poland, 2012, pp 1–6.
39. K. Zhang, D. Lo, E.-P. Lim and P.K. Prasetyo, "Mining indirect antagonistic communities from social interactions". Knowledge and Information Systems (KAIS), vol. 35, no. 3, pp. 553–583, 2013.
40. B. Zhou, and J. Pei, "Preserving Privacy in Social Networks Against Neighborhood Attacks" in ICDE '08. Washington, DC, USA: IEEE, 2008, pp. 506–515.
41. B. Zhou, and J. Pei, "The k-anonymity and ℓ-diversity approaches for privacy preservation in social networks against neighborhood attacks". Knowledge and Information Systems, vol. 28, no. 1, pp. 47–77, 2011.
42. L. Zou, L. Chen, and M.T. Özsu, "K-Automorphism: A General Framework For Privacy Preserving Network Publication" in VLDB '09, vol. 2, no. 1, pp. 946–957, 2009.