Social Network Anonymization and Influence Preservation

Alina Campan[*] and Yasmeen Alufaisan[*]

Abstract — **Social media has grown rapidly in the past few years. Facebook, Twitter, LinkedIn, and many other social media sites contain public and confidential information about their users. In order to protect the users' privacy, social network graphs are anonymized before being published or released to a third party for data mining or statistical analysis. Many social network anonymization models have been proposed, each with different assumptions and settings regarding the information that needs protection and possible privacy attack scenarios. The ultimate goal of all the anonymization models is to preserve the privacy of the social network's users and, at the same time, preserve enough information to enable a good analysis of the social network. In this work, we study how well we can preserve the important features in a social graph, specifically the nodes' influence in the network (as quantified by influence spread measures) while preserving privacy with different anonymization models.**

I. Introduction

As with other types of data (microdata, streams, location-based data etc.), social network graphs can be subjected to an anonymization process, before the social network data can be publicly released; the goal is to ensure the privacy of the social actors. Up until now, there are no standard models and algorithms for social network anonymization. Various solutions have been developed in the recent past, for different problem settings. Different anonymity approaches vary in their assumptions about: data available about the social actors and their relationships; private information that needs protection; background knowledge of an attacker [15]. Consequently, different anonymity models and methods to achieve them have been created corresponding to these problem settings. The resulting anonymized networks are very dissimilar, and so is the extent to which they preserve information inherent in the original network. For example, recent studies investigated how structural properties such as diameter, centrality measures, clustering coefficients, and topological indices are preserved between the original networks and their anonymized versions [14]. In this paper, we investigate how influence is preserved in social networks that undergo an anonymization process. Influence modeling has been studied with applications in understanding information diffusion, viral marketing ([6]), outbreak detection in networks ([9]). Influence spread was modeled and analyzed so that to find a small set of nodes in a network such that: their overall influence in the network is maximized (viral marketing), or they are able to detect most effectively the spreading of a process over a network (outbreak detection). We used two distinct anonymization approaches to mask several real and synthetic social networks: *k-anonymity for social networks* ([3]), which can be enforced on a network by using the *Sangreea* algorithm, and *k-degree anonymity*, enforced by the *Fast K-Degree Anonymization* algorithm ([11]). We measured and compared influence spreading in the original networks and in the anonymized networks. For networks masked with *Sangreea*, we had to do de-anonymization prior to measuring influence: this to make comparison with the original networks feasible, as we will explain later.

The paper is structured as follows. Next section reviews the two models we used for social network anonymization: *k-anonymity* and *k-degree anonymity*, and their respective anonymization methods. Section 3 presents our approaches to de-anonymize networks masked with *Sangreea*. We describe in Section 4 the influence spread measure we analyzed and the method we used to approximate influence. Section 5 describes how we measure influence preservation between an original network and its anonymized / de-anonymized version. Section 6 describes our experimental setup and results. The paper ends with conclusions.

II. Anonymity Models for Social Networks

K-degree anonymity was proposed for protection against identity disclosure due to attacks that use background information about nodes' degrees. A social network modeled as a simple graph $\mathcal{G} = (\mathcal{N}, \mathcal{E})$, where $\mathcal{N}$ is the set of nodes and $\mathcal{E}$ is the set of edges, is said to be *k-degree anonymous*, for a given k value (5, 7 etc.), if for every node X in $\mathcal{N}$, there are at least k-1 other nodes with the same degree as X [10]. Lu et. al. proposed in [11] an efficient solution for enforcing k-degree anonymity on social graphs: *FKDA* (*Fast K-degree Anonymization Algorithm*). *FKDA* works by trying to anonymize groups of at least k nodes in one step. Nodes to be next in an anonymized group are selected in decreasing order of their degree, among the nodes which haven't been yet anonymized in previous steps. The anonymization consists in wiring new edges to nodes in the group, until all have the same degree, equal to the largest degree in the group at the beginning of the step. Wiring is attempted with nodes with smaller degrees than the highest one in the group, and which haven't therefore been put through anonymization before. If anonymization cannot be achieved for a group by following this procedure, a more relaxed wiring is allowed, which can destroy the anonymity of nodes processed in previous steps (then, the whole process is re-started). We used *FKDA* for enforcing k-degree anonymity.

[*] Department of Computer Science, Northern Kentucky University, USA {campana1, alufaisany1}@nku.edu

.

K-anonymity for social networks, introduced in [3], can protect against identity disclosure and against content (or attribute) disclosure. According to this model, both the data and the structure associated to nodes are anonymized such that a node becomes undistinguishable from at least k-1 other nodes in the network. The key to the anonymization process as applied by *Sangreea* consists in clustering the nodes into a partition with sets of cardinality at least k, and which are as similar as possible to each other in terms of their attributes and their neighborhoods. Nodes in each cluster are merged into a supernode in the masked network. For each supernode, there is some information that will be released: its cardinality (is k or greater), the number of edges internal to the cluster, and the generalized attribute values describing all nodes in the cluster. Connectivity information between supernodes is also released: for each pair of supernodes, a weight representing the number of edges with ends in the two clusters is published. *Sangreea* can be geared towards preserving more the attributes of the nodes or the structure of the graph, by using two user defined parameters. We used *Sangreea* to take into account only the structure of the network, and not the nodes' attributes.

A network masked with *Sangreea* will obviously have a number of supernodes at most the size of the original network divided to k. Such an aggregated network cannot be fairly compared w.r.t. influence preservation with the original network or the *FKDA* anonymized network. To be able to inspect how influence is preserved through *Sangreea* anonymization, we need to execute an extra-step: we try to reverse the anonymization process and create a replica of the original network. We called this process *de-anonymization*. De-anonymization is implemented based on the information packaged in the aggregated network and assumes certain statistical distribution of the nodes' degrees. The de-anonymization process is described next.

Let $\mathcal{G} = (\mathcal{N}, \mathcal{E})$ be an initial social network and $\mathcal{MG} = (\mathcal{MN}, \mathcal{ME})$ be a corresponding k-anonymous social network masked with *Sangreea*, where $\mathcal{MN} = \{Cl_1, Cl_2,\ldots, Cl_v\}$, and $Cl_j = [gen(cl_j), (|cl_j|, |\mathcal{E}_{clj}|)]$, $j = 1..v$. This anonymized network was built based on a partition $\mathcal{S} = \{cl_1, cl_2, \ldots , cl_v\}$ of the node set $\mathcal{N}$, $\bigcup_{j=1,v} cl_j$ such that = $\mathcal{N}$; $\mathrm{cl_i} \cap \mathrm{cl_j} = \varnothing$; $i, j = 1..v$, $i \neq j$; where nodes were grouped such that nodes within every cluster cl_i were as similar to each other as possible w.r.t. their attributes and neighbors. The corresponding ***masked social network*** $\mathcal{MG} = (\mathcal{MN}, \mathcal{ME})$ has:

- $\mathcal{MN} = \{Cl_1, Cl_2,\ldots, Cl_v\}$, node Cl_j corresponds to cluster $cl_j \in \mathcal{S}$ and is described by a "tuple" $gen(cl_j)$ (the generalization information of cl_j, w.r.t. quasi-identifier attribute set) and an intra-cluster generalization pair $(|cl_j|, |\mathcal{E}_{clj}|)$;
- $\mathcal{ME} \subseteq \mathcal{MN} \times \mathcal{MN}$; $(Cl_i, Cl_j) \in \mathcal{ME}$ iif $Cl_i, Cl_j \in \mathcal{MN}$ and $\exists\ X \in cl_j$ and $Y \in cl_j$, such that $(X, Y) \in \mathcal{E}$. Each generalized edge $(Cl_i, Cl_j) \in \mathcal{ME}$ is labeled with the inter-cluster generalization value $|\mathcal{E}_{cli,clj}|$.

For both anonymity models, we make the assumption that nodes' identities are released, as follows. For k-degree anonymity, we assume that identities of the nodes are released together with those nodes' degree in the anonymized network; for example, nodes *John*, *William*, and *Mary* have an anonymized degree of 4 – in this example, we assume k to be 3, therefore each group of nodes with the same degree has cardinality at least 3. We will call these groups of nodes with the same anonymized degree ***anonymity clusters***. Obviously, for random networks, the anonymity clusters will contain a rather large number of nodes from $\mathcal{N}$. However, for scale-free networks, it is expected that anonymity clusters for large degree values will have cardinality close to k, while anonymity clusters for small degree values will still have large cardinality.

For k-anonymity for social networks, we assume that the identities of the entities in each supernode are disclosed; for example, the supernode Cl_i in the 3-anonymous network consists of the nodes *John*, *William*, and *Mary*. In this model's case, each supernode is an anonymity cluster.

These assumptions about nodes' identities are not against the definitions of the two models, nor do they weaken the models' strength. These assumptions are necessary; otherwise a masked network would be unusable, for example, for viral marketing. Identifying, even accurately, the most influential nodes in an anonymized network would be useless if the nodes were unidentified, since they could not be targeted with different promotions without knowing who are the people represented by those nodes.

III. De-anonymization for *Sangreea* Networks

We used two procedures to de-anonymize a network masked with *Sangreea* to try to reconstruct the original social graph $\mathcal{G}$. Each one of these two procedures assumes a certain type of degree distribution for the nodes in the original network.

The first de-anonymization method, ***uniformReconstruct***, is based on the assumption that the node degrees, and therefore edges in the graph, are uniformly distributed among nodes. ***uniformReconstruct*** will then randomly reconnect with edges nodes that belong within each cluster, and then nodes in every pair of clusters. We are omitting the algorithm for ***uniformReconstruct*** due to space constraints.

Many real-world networks do not have a uniform distribution of the nodes' degrees. Instead, they are scale-free, and their node degree distribution follows a power-law. Our second de-anonymization method, ***rmatReconstruct***, is based on this assumption about node degree distribution. We use an *R-MAT* generation procedure ([4]) to de-anonymize an anonymous network $\mathcal{MG} = (\mathcal{MN}, \mathcal{ME})$.

```
Algorithm rmatReconstruct is
Input: $\mathcal{MG} = (\mathcal{MN}, \mathcal{ME})$ – a $k$-anonymous social
        network for $\mathcal{G} = (\mathcal{N}, \mathcal{E})$
        $\mathcal{MN} = \{Cl_1, Cl_2, \ldots , Cl_v\}$, where $Cl_j$ has
        cardinality $|cl_j|$, and the identities of
        the nodes in $cl_j \subseteq$ N  are known (*)
        $\mathcal{ME} \subseteq \mathcal{MN} \times \mathcal{MN}$ and each edge $(Cl_i, Cl_j) \in$
        $\mathcal{ME}$ has a weight $|\mathcal{E}_{cli,clj}|$, which is the
        number of edges in $\mathcal{E} \cap (Cl_i \times Cl_j)$
Output: $\mathcal{G}' = (\mathcal{N}, \mathcal{E}')$ a de-anonymized network with
        the same node set as $\mathcal{G}$ and $|\mathcal{E}'| = |\mathcal{E}|$
```

```
Set the adjacency matrix of $\mathcal{G}'$, $\mathcal{AM}'$, to be the
zero matrix; this is equivalent to $\mathcal{E}' = \emptyset$;
For every $Cl_j \in \mathcal{ME}$ do:
  count = 0
  While count < $|cl_j|$:
    Use rmatEdgeGeneration on the restriction
    of $\mathcal{AM}'$ to the rows & columns representing
    nodes in $cl_j$ to generate a random edge
    $(X,Y)$: $X,Y \in cl_j$, $X \neq Y$, $(X,Y) \notin \mathcal{E}'$
    $\mathcal{E}' = \mathcal{E}' \cup \{(X,Y)\}$
    Update $\mathcal{AM}'$ to reflect the newly added edge
    count++
For every $(Cl_i, Cl_j) \in \mathcal{ME}$ do:
  count = 0
  While count < $|\mathcal{E}_{cli,clj}|$:
    Use rmatEdgeGeneration on the restriction
    of $\mathcal{AM}'$ to the rows & columns representing
    nodes in $cl_i \cup cl_j$ to generate a random edge
    $(X,Y)$: $X \in cl_i$, $Y \in cl_j$, $(X,Y) \notin \mathcal{E}'$
    $\mathcal{E}' = \mathcal{E}' \cup \{(X,Y)\}$
    Update $\mathcal{AM}'$ to reflect the newly added edge
    count++
End uniformReconstruct;
```

```
Algorithm rmatEdgeGeneration is
Input:  An adjacency matrix $\mathcal{AM}$
        Parameters $a$, $b$, $c$, $d$: $a + b + c + d = 1$
Output: A (row, column) location in $\mathcal{AM}$, chosen
        according to parameters $a$, $b$, $c$, $d$, that
        indicates a new edge $(X,Y)$ to be added
        to the graph represented by $\mathcal{AM}$.
If $\mathcal{AM}$ has a single row and column:
  Return that position in the matrix

Generate a random number $r$, in range [0, 1].
Divide $\mathcal{AM}$ in 4 equal-size partitions, top-left,
top-right, bottom-left, and bottom-right
If $r < a$:
  rmatEdgeGeneration(top-left, a, b, c, d)
Else If $r < a + b$:
  rmatEdgeGeneration(top-right, a, b, c, d)
Else If $r < a + b + c$:
  rmatEdgeGeneration(bottom-left, a, b, c, d)
Else:
  rmatEdgeGeneration(bottom-right, a, b, c, d)
End rmatEdgeGeneration
```

Note (*): we explained before why we assume that the identities of the nodes in the original network that belong to each supernode in $\mathcal{MN}$ are known for the corresponding released k-anonymous network $\mathcal{MG} = (\mathcal{MN}, \mathcal{ME})$.

The *R-MAT* procedure takes 4 probabilities, called a, b, c, d as input parameters, where $a + b + c + d = 1$. It works on a submatrix of the adjacency matrix of $\mathcal{G}$' which is: a restriction of it to a cluster (to generate internal edges in that cluster), or a restriction of it to two clusters (to generate inter-cluster edges). ***rmatEdgeGeneration*** recursively determines the location of a new edge in this matrix: the algorithm divides the adjacency matrix into 4 equal-sized partitions and the location of the new edge is probabilistically selected in one of the 4 locations, based on the 4 probability parameters. Once a partition is found, it is again divided into 4 sub-partitions until there will be only one location left in the partition. If an edge was already placed on that location, we will repeat this procedure from the beginning (multiple edges between the same pair of nodes are not allowed in our graph model). For all our tests we used the following values for the 4 probabilities: 0.45, 0.15, 0.15, and 0.25. This choice seems to model better many real-world graphs that follow power-law degree distributions [4]. As explained in [4], this generation technique will create 2 large well-connected “communities” in the graph: one among the nodes in the first “half” of the node set (the top-left quadrant in the adjacency matrix), the other among the nodes in the second half of the node set (the bottom-right quadrant in the adjacency matrix). Edges are created with higher probability among nodes in those respective halves, since parameters a and d are higher. The two communities are more loosely connected, as decided by the lower probabilities b and c that command the placement of edges between nodes belonging to different halves. The process is repeated recursively in each quadrant such that larger communities are divided in smaller and smaller communities.

Since we need a symmetric adjacency matrix to reflect that our social network graph is undirected, the adjacency matrix produced with ***rmatReconstruct*** is finally processed once more. The matrix entries above (or below) the main diagonal are discarded and the other half is copied over it to make it symmetric. Since parameters b and c are equal, the number of edges that result by applying this transformation is fairly equal to the number of edges in the uncut matrix.

IV. Influence in Social Networks

Influence spreading, or propagation, has been studied in a number of fields for a while now: sociology, viral marketing ([6], [8]), outbreak detection in networks ([9]). The linear threshold influence model (*LTM*) and the independent cascade influence model (*ICM*) are among the most used models for influence spreading ([8]). Influence models are used in solving the influence maximization problem: given a network and a parameter k, find a set of k nodes in the network that, when activated, can spread their influence to more network nodes than any subsets of nodes of size k. Please note that k in this context has a different meaning, totally unrelated, from k as in k-anonymity.

We chose to use the *LTM* for influence spreading, and the degree-discount algorithm ([5]) for determining the subset of nodes that could maximize the spread of influence.

Under *LTM*, a social network is modeled as a directed graph, $\mathcal{G} = (\mathcal{N}, \mathcal{E})$. Note: the two anonymity models we are studying both employ undirected graphs; we cope with this difference between the influence model and the anonymity models by simply considering each undirected edge in the anonymity models to be equivalent to two directed edges between the same nodes, when computing the spread of influence. Each node in $\mathcal{G}$ can be either active or inactive. Nodes that are active (i.e. have adopted a product or embraced a new idea) can further activate other nodes, which are currently inactive. Each node is influenced in a certain degree by each one of its neighbors. The influence that a node w exerts over its neighbor node v (this means that (w, v) is a directed edge in $\mathcal{E}$) is denoted by $b_{v,w}$ where $b_{v,w} \geq 0$ and $\sum_{\text{w is a neighbor of v}} b_{v,w} \leq 1$. A choice for the weights $b_{v,w}$ for a node v is $1 / |\mathcal{N}_v|$, where $\mathcal{N}_v = \{w \in \mathcal{N}, (w, v) \in \mathcal{E}\}$ is the set of all nodes in $\mathcal{N}$ that are connected to v through edges pointing to v. This means that all v's neighbors have the same influence on v. Each node v chooses an activation threshold θ_v, uniformly at random from the interval [0, 1]; the node v will become active when the overall strength of all its active neighbors passes its threshold. In other words, v will become active when $\sum_{\substack{w \text{ is a neighbor of } v, \\ w \text{ is active}}} b_{v,w} \geq \theta_v$. The randomness in choosing the activation thresholds of the network nodes models our lack of knowledge regarding how susceptible to influence are the social actors in a network.

Given randomly selected thresholds for all nodes in $\mathcal{G}$, and a set of initially active nodes $\mathcal{S}$ (= the ***seeds***), the activation process proceeds in steps. In each step, the previously active nodes remain active, and inactive nodes that have enough active neighbors will be activated as well. The spreading process stops when no further nodes can be activated.

The influence maximization problem can be stated as follows. If $\sigma(A)$ denotes the expected number of nodes that will be influenced if the set A is initially activated, find the set of seeds $\mathcal{S}$, of size k, that has the maximum influence in the network. This set, called the ***seed set***, is a solution for the optimization problem $\max_{B \subseteq N} \sigma(B)$, such that $|B| = k$.

Kempe et al. showed in [8] that finding the optimum seed set under the *LTM* is NP-hard. They also proposed a greedy algorithm that is able to find an (1-1/e) approximation of the optimum solution; i.e. the solution found by the greedy algorithm will be at least 63% of the optimal one. This result is based on two significant properties that the influence function $\sigma(\cdot)$ has been proven to have: $\sigma(\cdot)$ is monotone and submodular (see [8] for definitions and proofs).

This greedy algorithm for the influence maximization problem has unfortunately a drawback, its efficiency. We therefore chose to use a different algorithm for the influence maximization problem, which is based on heuristics and has been proven to reduce the running time by more than six orders of magnitude ([13]). Chen et al. proposed the ***degree discount*** heuristic for estimating the most influential nodes in a network. Selecting a seed set based on the degree discount heuristics has been shown to be very efficient and to achieve, under the *ICM*, an influence spread almost as large as the one produced by the greedy algorithm; for other influence models (*LTM* included), degree discount has been said to have an improved performance compared to other heuristics, such as the pure degree heuristic.

Under the degree discount heuristic, the best k seeds for initial activation in the network are selected as follows. The selection proceeds in k steps, in each step a new seed is chosen, that has the highest discounted degree among the nodes not chosen yet. The discounted degrees of the nodes are initially, before the first selection is made, equal to the actual degrees of the nodes. After each seed selection, the discounted degrees of the nodes that are neighbors of that seed are decremented by 1. This alteration reflects the basic idea that it is not worth it to make a seed (i.e. initially active) a node that already has seed(s) in its neighborhood; this because that node will be potentially activated by the neighboring seeds, and then it will itself further spread its influence to its inactive neighbors.

V. Measuring Influence Preservation

In our experiments, we compared influence for the seed sets of the original social networks with seed sets of the corresponding *FKDA* anonymized network and the de-anonymized *Sangreea* networks. We describe next how the comparison can be performed, taking into consideration the point of view of a user attempting to do marketing targeted to the most influential nodes in an anonymized network.

Assume a user disposing of a budget for promoting products or services to p% of the network nodes. Of course, they would want the nodes they target to be the most influential in the network. Let's first assume the network they have has been anonymized with *Sangreea*. They can de-anonymize this network and determine the k most influential nodes in the de-anonymized network, where k is chosen to be a certain percentage of the network size. How is the k value to be chosen? k cannot be $|\mathcal{N}| * p / 100$, for the following reason. When we de-anonymize a network, we use information that we recorded during anonymization about the composition of each cluster: what node IDs belong to which cluster. However, the nodes within a cluster are anonymous and cannot be distinguished from each other, so a node restored from cluster $cl_j = \{X_j^1, X_j^2, \ldots, X_j^{|clj|}\}$ with id X_j^r will not necessarily be the same one that was identified by X_j^r in the original cluster; it could instead be anyone of the other nodes assigned to cluster cl_j. This further means that if X_j^r is determined as one of the seeds in the de-anonymized network, any one of the nodes in its cluster could actually be the real influential node, not necessarily X_j^r. Therefore, someone who wants to be sure they do not miss the real influential node(s) in a cluster containing a seed / seeds will have to basically target all the nodes in that cluster.

To stay in the allowed budget, one has to find less than or at most equal to p% most influential nodes. One would first search for the p% most influential nodes. If they happen to populate exhaustively their clusters, then the process would stop. If however, and this is much more likely, the clusters containing seeds also contain other nodes, one has to reduce the target percent, repeat the seed set determination, and check if they are in the allowed budget. The process will stop at the first p*% found for which $\left|\bigcup_{s \in \mathcal{S}^*} cl^s\right| * 100 \Big/ |\mathcal{N}| \leq p$, where $\mathcal{S}$* is the seed set with cardinality $|\mathcal{N}|$ x p^* / 100, and cl^s is the cluster containing the seed s. In our experiments, we computed p^* as follows: we started with p^* being equal to p; we then found the most influential p*% nodes in the anonymized network; next, we determined the set of all nodes found in clusters containing seeds, $T(\mathcal{S}^*) = \left|\bigcup_{s \in \mathcal{S}^*} cl^s\right|$ - we call this set the ***targeted set***; if its size is greater than the budget of $|\mathcal{N}|$ x p / 100 nodes, then p* is reduced to 0.95 x p*; this adjustment process is repeated until the targeted set fits into the budget.

Once the seed set $\mathcal{S}$* is found, we can estimate and compare the spread of the most influential p% nodes in the original un-anonymized network, with the spread of the targeted set $T(\mathcal{S}^*)$. The loss incurred by targeting the nodes in $T(\mathcal{S}^*)$ instead of targeting the most influential p% nodes in the original network can also be computed as $loss = \sigma(\mathcal{S}) - \sigma(T(\mathcal{S}^*))$, where σ is the influence function defined under *LTM* and $\mathcal{S}$ is the seed set of size $|\mathcal{N}|$ * p / 100, determined in the original network. σ is computed for both $T(\mathcal{S}^*)$ and $\mathcal{S}$ in the original network. The *loss* measure represents the estimated number of nodes that can be reached when activating $\mathcal{S}$ but cannot be reached when activating $T(\mathcal{S}^*)$. Theoretically, *loss* should be a positive measure, since $\mathcal{S}$ is the most influential set that could be found by the degree discount procedure; this set is obviously not the optimum solution for the influence maximization problem, but it is very likely to still be better than $T(\mathcal{S}^*)$.

The algorithm we used to estimate the spread of influence in a network, for an initial set of active seeds, under *LTM*, is based on a Monte-Carlo simulation.

```
Algorithm estimateSpread is
Input:   $\mathcal{G} = (\mathcal{N}, \mathcal{E})$ and a set of seeds $\mathcal{S} \subseteq \mathcal{N}$
Output:  An estimate of $\sigma(\mathcal{S})$ in $\mathcal{G}$
R = 10000; spread = 0;
For i = 1, R do:
   Select random thresholds for nodes in $\mathcal{G}$
   Perform a LT spread simulation in $\mathcal{G}$, with
   seed set $\mathcal{S}$; let count be the number of nodes
   activated in that simulation
   spread += count
Return spread/R
End estimateSpread.
```

For *Sangreea*, *loss* can be computed as the difference between the result of *estimateSpread*($\mathcal{G}$, $\mathcal{S}$) and that of *estimateSpread*($\mathcal{G}$, $T(\mathcal{S}^*)$). For *FKDA* networks, the loss in influence due to anonymization can be computed similarly as for *Sangreea*, with a modification: when determining p*% and the seed set $\mathcal{S}$*, the targeted set $T(\mathcal{S}^*)$ is computed as

$$T(\mathcal{S}^*) = \bigcup_{s \in \mathcal{S}^*} \{X \in \mathcal{N} \mid X \text{ has same degree as } s\}.$$

Since every seed in the *FKDA* anonymized network is undistinguishable from the other nodes in the network with the same degree, when a seed is selected in $\mathcal{S}$*, all nodes with the same degree should be targeted, to be sure that the true influential node is targeted. Compared to *Sangreea*, the clusters of anonymous nodes that *FKDA* creates are the subsets of nodes in $\mathcal{G}$' with the same degree. Once $T(\mathcal{S}^*)$ and $\mathcal{S}$ are determined, *loss* could be again computed as the difference between the result of *estimateSpread*($\mathcal{G}$, $\mathcal{S}$) and that of *estimateSpread*($\mathcal{G}$, $T(\mathcal{S}^*)$).

VI. Experiments and Results

We study influence preservation (with degree discount) in the original, anonymized (for *FDKA*), and de-anonymized (for *Sangreea*) versions of three datasets.

The **Enron** dataset is a network of email exchanges available online at [7]. It is an undirected network with 36,692 nodes and 183,831 edges. Each node in this network represents an email address. And edge exists between two nodes if at least one email was sent from one node to the other from that edge. The **Random** dataset is synthetically generated using the Erdos-Renyi random network model [1] using the social network analysis program Pajek [12]. We used as input parameters for the social network generator 10,000 nodes and an average vertex degree of 20. The resulting network has 100,314 edges. The **ScaleFree** dataset is an undirected network generated based on the scale-free model [13]. This approach models real world social networks that follow a power-law degree distribution [2]. We generated this dataset using Pajek with the following parameters: the number of nodes 10,000, the average degree of nodes of 33, the number of nodes in the initial Erdos-Renyo graph 10. The generated graph has a significant number of multiple edges which were eliminated in a post-processing step. The final scale-free network that we used in experiments had 10,000 nodes and 152,909 edges.

The flow of our experiments is shown in Figure 1. This experimental framework consists of 6 steps. We start from the initial social networks (Enron, Random, and ScaleFree) previously described. First, the initial social networks are anonymized into k-anonymous social networks, using *FKDA* (step 1a) and *Sangreea* (step 1b) as described in Section 2. For each dataset we used the following values for k: 2, 3, 4, 5, 6, 7, 8, 9, 10, 15, 20, 25, and 50. Second, from each k-anonymous *Sangreea* network we generated two de-anonymized social networks, one following the *Uniform* de-anonymization strategy (step 2a) and the other the *R-MAT* de-anonymization strategy (step 2b). The need for performing de-anonymization on *Sangreea* networks was explained in Section 2. In Step 3, we computed the seed set $\mathcal{S}$ of the most influential p% nodes in the original networks, where p has values 2, 4, 6, 8, 10. In Step 4, we computed the seed set $\mathcal{S}^*$ of the most influential p^*% nodes in the *FKDA* networks and the de-anonymized *Sangreea* networks, where p^* is computed as described in section 5 for p values 2, 4, 6, 8, 10. Each of these sets $\mathcal{S}^*$ have the corresponding $T(\mathcal{S}^*)$. In Step 5, we also consider random selections for the seed sets in the original networks, denoted by $\mathcal{S}_{\text{random}}$, for the same p values 2, 4, 6, 8, and 10 – 5 random seed sets of each size, for each of the networks. In Step 6, we compare the influence of seed sets $\mathcal{S}$, $T(\mathcal{S}^*)$, and $\mathcal{S}_{\text{random}}$: the influence of these seed sets in the original network is estimated using the ***estimateSpread*** procedure, and is reported as a percentage of the network size. Since we generated 5 random seed sets for each original network, the influence determined in those cases is averaged.

Figures 2 a-f show the results of steps 5 and 6, for the Random, ScaleFree, and Enron datasets.

For the Random network, *FKDA* preserved reasonably well the spread of influence (Figures 2 a-b). The only situation where *FKDA* dropped rapidly is when p = 8 and k was 25 or 50 (not shown here). The reason for that behavior is that the targeting set $T(\mathcal{S}^*)$ kept decreasing by 5% of its size multiple times, until its size reached 578 nodes with

k=25 and 581 nodes with k=50 – by comparison, the size of the seed set for the original network was 800 (= 8% of 10000). Any significant difference in the size of the targeting set, compared to the current p% budget size, will definitely decrease the spread of influence for the targeted set; this happens regardless of the anonymity model. In this particular case, for k=25 and k=50, the size of the targeted set before the final 5% reduction might have been just a little bit over 800, and the last 5% adjustment was too drastic. With *Sangreea R-MAT*(*Reconstruct*), the spread of influence was well preserved especially when k got larger. *Sangreea Uniform*(*Reconstruct*), on the other hand, was the weakest in influence preserving, which indicates that it is not worth it to de-anonymize *Sangreea* networks with the *Uniform* algorithm for any random network.

For the ScaleFree network (Figures 2, c-d), *FKDA* and the original networks have almost identical spread of influence for all p values. *Sangreea* de-anonymization with *R-MAT* and *Uniform* have the same spread of influence until the anonymity parameter k reaches 10, for 2% the size of the seed set, and until k reaches 7, for the remaining p values.

For Enron, *FKDA* preserved well the most influential nodes with all the p values (similar behavior was recorded for the p values 4 and 8, which are not illustrated here).

However, *R-MAT* and *Uniform* de-anonymization for *Sangreea* have a similar behavior for all p values: the spread of influence decreased almost linearly, with p. In all these cases *R-MAT* had much better results than *Uniform*.

So, overall, *FKDA* preserved well the spread of influence in all networks. De-anonymized *Sangreea* networks weren't as good as *FKDA* networks, except for the Random network, where *R-MAT* over performed *FKDA* in about ½ of the cases. But always *R-MAT* behaved better than *Uniform,* even for the Random network. We also noticed that the random selection of the seed set didn't preserve the most influential nodes in any of the networks.

After all, the preservation of the spread of influence under the user's point of view assumption is almost entirely dependent on the purity of the anonymity clusters w.r.t. the most influential nodes in the network. If anonymity clusters do not contain any residual nodes, meaning $T(\mathcal{S}^*)$-$\mathcal{S}^*$ is close to $\varnothing$, then $\mathcal{S}^*$ is as big as the p% budget, and only real influential nodes are targeted. That would really ensure preservation of spread of influence compared to the original un-anonymized network. The question about the preservation of the influence spread is now reduced to how pure are the anonymity clusters produced by *Sangreea* or *FKDA* (purity from the point of view explained before). The *FKDA* anonymity clusters are induced by the groups of nodes anonymized together, which are nodes that have similar degrees. For high degrees, the *FKDA* anonymity clusters are small, since there are few nodes with high degrees, especially in scale-free networks. For smaller degrees and scale-free networks, the *FKDA* anonymity clusters could be bigger – and the chance of them becoming impure grows. On the other side, the degree discount procedure identifies the most influential seeds to be, more or less, the nodes with the highest degrees. Therefore, the most influential nodes will correspond to the anonymity clusters with the highest node degree, which are, as we discussed, small; their size should be about k, where k is the anonymity parameter, as in k- degree anonymity. Since we just look for p% of the most influential nodes, with p having small values in general, this means we never reach to the anonymity clusters with low node degree values, which could be larger than k and impure. Since the principle based upon which *FKDA* anonymizes nodes is so similar to the principle based upon which degree discount finds the most influential nodes, clearly the *FKDA* anonymity clusters, at least the ones that will be considered within the p% budget, tend to be very pure, for scale-free networks. For random networks, where nodes are more uniform in terms of their degrees, the *FKDA* clusters are not that pure anymore; this is reflected in smaller influence preservation scores. Only for the Random network had *FKDA* much smaller influence preservation; for Enron and ScaleFree, *FKDA*'s influence preservation was almost 100%. For Random though, *FKDA* had in 2 cases smaller preservation compared to *Sangreea R-MAT* (for p: 2, 10).

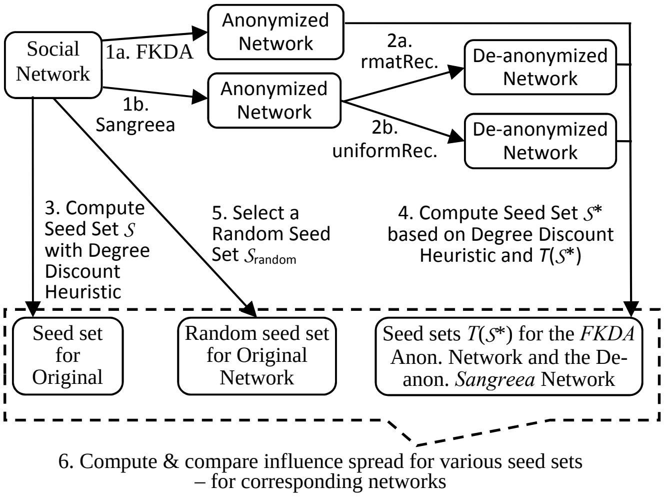
Social Network
1a. FKDA
Anonymized Network
1b. Sangreea
Anonymized Network
2a. rmatRec.
De-anonymized Network
2b. uniformRec.
De-anonymized Network
3. Compute Seed Set $\mathcal{S}$ with Degree Discount Heuristic
5. Select a Random Seed Set $\mathcal{S}_{\text{random}}$
4. Compute Seed Set $\mathcal{S}^*$ based on Degree Discount Heuristic and $T(\mathcal{S}^*)$
Seed set for Original
Random seed set for Original Network
Seed sets $T(\mathcal{S}^*)$ for the FKDA Anon. Network and the De-anon. Sangreea Network
6. Compute & compare influence spread for various seed sets – for corresponding networks

Fig. 1. Flow of Experiments

Sangreea attempts to put together in a supernode some of the original nodes that have the same neighbors, as much as possible. This could go somewhat against the way the degree discount procedure identifies the most influential nodes, and therefore it is expected to get more impure anonymity clusters compared to the ones created by *FKDA*. Also, in *Sangreea*'s case, more error is added to the one introduced by the anonymization itself, due to the de-anonymization process. As expected, *Sangreea* followed by *R-MAT* or *Uniform* de-anonymization will not preserve influence spread as well as *FKDA*.

VII. Conclusions

Anonymization models have been used to ensure the privacy of social networks. A conflicting goal with maintaining the privacy of a network's information is the preservation of the structural properties of the social network. The goal of this work was to investigate whether we can preserve privacy in social networks using anonymization techniques and in the same time preserve

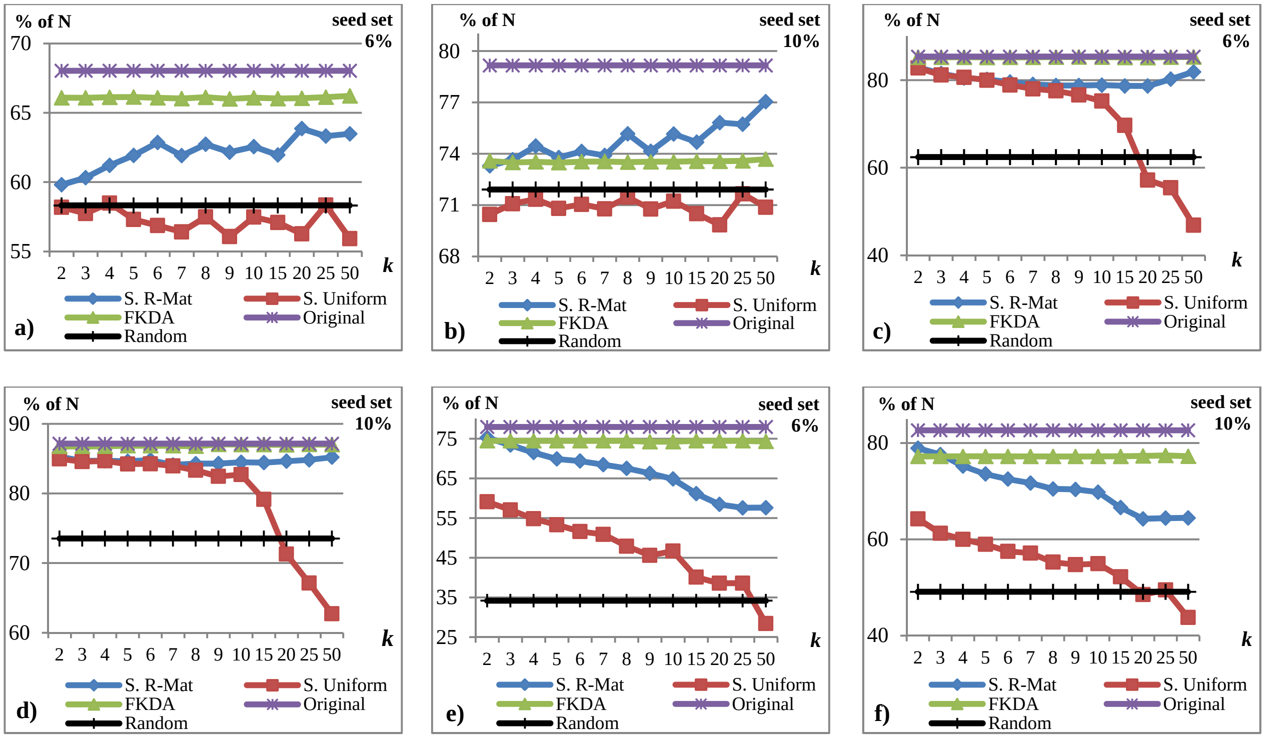

Fig. 2. Influence spread of most influential 6% (a, c, e) and 10% (b, d, f) nodes in the original network vs. influence spread of targeted sets T(S*) of size ≈6% and ≈10% in the FKDA Anonymized and the De-anonymized Sangreea networks – for the Random (a, b), ScaleFree (c, d), and Enron (e, f) networks

enough information to allow a good analysis of the properties of the social graph. We looked at how an influence spread measure changed between the original and the anonymized networks. *FKDA* had a better preserving for the spread of influence, compared with *Sangreea R-Mat* and *Uniform*. When comparing *R-Mat* and *Uniform*, the experiments showed that *R-Mat* is a stronger approach than *Uniform* – in the sense that better reconstructs the original networks, without disclosing information, but preserving well the influence spread from the original networks. This was to be expected at least for the scale-free networks, but it was also true for random networks; the explanation can be that, for small values of *k* that correspond to small clusters to be reconstructed, *R-Mat* and *Uniform* reconstruction produce sub-graph structures that are not very different, and therefore have similar influence spread. The better preservation of the spread of influence in *FKDA*'s case comes with a cost: *FKDA* is a much weaker model for preserving privacy than *Sangreea* is. An attacker with knowledge about the 2-radius neighborhood of a target node could still reidentify his target in an *FKDA* network, if the target's 2-radius neighborhood has some unique feature. *Sangreea*'s privacy preserving is stronger: an attacker won't be able to breach the privacy of a *Sangreea* network based on any structural properties knowledge.

References

[1] B. Bollobás, *Random Graphs*, Cambridge University Press, 2011.

[2] B. Bollobás, O. Riordan, J Spencer, G. Tusnady, *The Degree Sequence of a Scale-Free Random Graph Process*, Journal of Random Structures and Algorithms, Vol. 18 (3), pp. 279-290, 2011.

[3] A. Campan, T.M. Truta, *A Clustering Approach for Data and Structural Anonymity in Social Networks*, 2nd ACM SIGKDD Intl. Workshop on Privacy, Security, & Trust in KDD, USA, 2008.

[4] D. Chakrabarti, Y. Zhan, C. Faloutsos, *R-MAT: A Recursive Model for Graph Mining*, SIAM Data Mining 2004, Orlando, USA, 2004.

[5] W. Chen, Y. Wang, S. Yang, *Efficient influence maximization in social networks*, Proc. of the 15th ACM SIGKDD Intl. conference on Knowledge Discovery and Data Mining (KDD '09), 2009.

[6] P. Domingos, M. Richardson, *Mining the Network Value of Customers*, Proc. of the 7th Intl. Conference on Knowledge Discovery and Data Mining (ACM SIGKDD), pages 57-66, USA, 2001.

[7] Enron Dataset, http://snap.stanford.edu/data/email-Enron.html.

[8] D. Kempe, J. Kleinberg, E. Tardos, *Maximizing the Spread of Influence through a Social Network*, Proc. 9th ACM SIGKDD Intl. Conf. on Knowledge Discovery and Data Mining, 2003.

[9] J. Leskovec, A. Krause, C. Guestrin, C. Faloustos, J. Vanbriesen, N. Glance, *Cost-effective outbreak detection in networks*, Proc. of the 13th ACM SIGKDD international conference on Knowledge Discovery and Data Mining, KDD 2007: 420-429, USA, 2007.

[10] K. Liu, E. Terzi, *Towards identity anonymization on graphs*, in SIGMOD, 2008.

[11] L. Xuesong, Y. Song, S. Bressan, *Fast Identity Anonymization on* Graphs, Database and Expert Systems Applications - 23rd International Conference, DEXA 2012, pp. 281-295, Austria, 2012.

[12] W. de Nooy, A. Mrvar, V. Batagelj, *Exploratory Social Network Analysis with Pajek*, Revised and Expanded Second Edition, Structural Analysis in the Social Sciences, Vol 34, Cambridge University Press, 2011.

[13] D.M. Pennock, G. W. Flake, S. Lawrence, E. J. Glover, C. L. Giles, *Winners don't take all: Characterizing the competition for links on the web*, PNAS, Vol. 99, No 8, pp. 5207-5211, 2002.

[14] T.T. Truta, A. Campan, A.L. Ralescu, *Preservation of Structural Properties in Anonymized Social Networks*, the Collaborative Communities for Social Computing Workshop, USA, 2012.

[15] B. Zhou, J. Pei, W.-S. Luk, *A Brief Survey on Anonymization Techniques for Privacy Preserving Publishing of Social Network Data*, ACM SIGKDD Explorations, 10(2):12–22, 2008.