Faculty of Science, Technology and Communication

Making social graphs resistant to active attacks

Thesis Submitted in Partial Fulfillment of the Requirements for the Degree of Master in Information and Computer Sciences

Author:
Bochuan Xuan

Supervisor:
Prof. Dr. Sjouke Mauw

Reviewer:
Dr. Gabriele Lenzini

Advisor:
Dr. Rolando Trujillo-Rasua

September 2015

Abstract

The growing popularity of social networks has generated interesting data analysis problems. An important concern in the release of these data for study is their privacy, since social networks usually contain personal information. Unfortunately, most of the previous studies on privacy preservation can deal with the attacking by adversaries with structural knowledge only, and cannot resist the attacks by stronger adversaries who can affect the structure of the social network graphs actively which is named active attack. In this thesis, based on the privacy metric (k, ℓ)-anonymity against active attacks, we introduce two anonymization methods to transform graphs to $(k, 1)$-anonymous graphs which can preserve the privacy of individuals and enable useful researches. The two methods are: *Connectivity-Preserving Approach* which devotes to preserve the structure of the original graph to a larger extent, and *Edge-Preserving Approach* which reaches $(k, 1)$-anonymity fast in terms of the number of added edges. The empirical study on both synthetic data and a social network data, illustrates that anonymized social networks generated by our methods can still resist against active attacks where the adversary controls more than one node in the network.

Acknowledgements

Since the beginning of March, I have been working on my master thesis in the group of Security and Trust of Software Systems in University of Luxembourg for six months. During this period, I got a lot of help from my supervisor, my advisor and my friends. I would like to express my gratitude to them one by one in order to memorize their direct or indirect contributions to my master thesis.

First of all, I want to thank my supervisor Prof. Sjouke Mauw. One year ago, he accepted me as an exchange student in University of Luxembourg. This one-year exchange experience not only taught me the knowledge in information security but also developed me into an independent, optimistic, sunshine girl, which could be the priceless wealth in my life. In addition, Prof. Mauw helped me review my thesis for the final check before submission and gave me some useful comments.

I am also grateful to my advisor, Dr. Rolando Trujillo Rasua, for his patience, motivation, and immense knowledge. His guidance helped me in all the time of the research and the writing of this thesis. I was always welcomed in his office no matter how busy he was. He could usually thought of new ideas when I got stuck and guided me to the right direction when I was lost. During my thesis writing, he usually read it carefully and gave me useful feedback.

I also want to thank my friends, Qixia Yuan, Yang Zhang and Weiwei Li. Qixia taught me how to use HPC and helped me review my thesis. Yang gave me some valuable advices on my thesis. Although Weiwei is not majored in Computer Science, she gave me advice from her point of view, which is also helpful for my thesis. They all encouraged me to be positive when I met problems. With them I do not feel lonely and homesick.

Last but not least, I want to thank my parents for their unconditional love and support.

Bochuan Xuan

Luxembourg, September, 2015

Contents

List of Tables

List of Figures

List of Algorithms

Chapter 1

Introduction

In this chapter, we present the motivation that led us to develop this research. We provide an overview of the (k, ℓ)-anonymity concept, which is the privacy metric for social graphs we use in this thesis. We also summarize our contributions. Finally, the structure of this thesis is detailed.

1.1 Motivation

A social network is a social structure which is made up of actors and their interactions. Each actor in the social network can search and check the profile of the social network members, post comments on their profiles, publish their status, etc. Without any constraint of the physical spaces, a social network makes it easier to communicate, interact and socialize for web users than face-to-face communications.

An article[1] by Sharon Guadin shows that in 2013 nearly 30% of the world's population use social networks like Facebook, Instagram, Google+ each month, which is an increase by 14.2% from 2012. This increase is predicted to continue by eMarket, a market research company. Their forecast is shown in Figure 1.1. The growth in the number of social network users around the world may be slowing but it shows little sign of stopping. By 2017, 2.33 billion people will use social networks.

A recent statistics[2] in August, 2015, also provides information on the most popular worldwide networks, ranked by the number of active accounts as shown in Figure 1.2. Market leader Facebook is the first social network to surpass 1 billion registered accounts.

The success of these social networks has attracted the attention of the media (e.g., [2][7][25][34][37]) and researchers. The research is often built upon the existing literatures on social network theory (e.g., [17][30][31][47]). To enable researchers' useful analysis such as community detection, link prediction, identifying prominent actors, social network characterization, social network data is usually published in the form of graphs consisting of nodes and edges. Nodes are representations of either individuals or organizations who have one or more attributes. The edges denote relationships or interactions between these nodes, such as financial exchange, friend relationship, web links, disease transmissions, which is the sensitive information the individuals do not want to reveal. For example, in Facebook, nodes indicate individuals and edges specify friendships. Users on Facebook cannot only view the profile of their friends, neighbors of them in the corresponding graph, but also send instant messages with them, which corresponds to edges in the graph.

[1] http://www.emarketer.com/Article/India-Leads-Worldwide-Social-Networking-Growth/1010396

[2] http://www.statista.com/statistics/272014/global-social-networks-ranked-by-number-of-users/

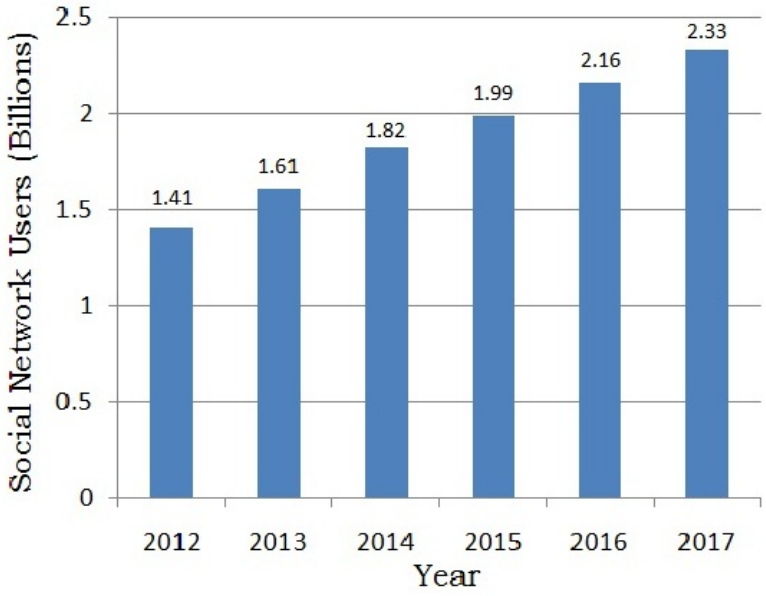


Figure 1.1: Social Network Users Worldwide[1], 2012-2017

The structure of social network graphs is usually published for useful analysis which can be used to fields as varied as marketing [16], sociology [8] and even counter-terrorism [22][39]. As is shown above, the usage of social networks has become widespread, everyone can get access to these data after the social network graph is published, thus the privacy in social networks becomes a serious concern [40].

Our goal is to enable the useful analysis of social network data while protecting the privacy of individuals. Many works focused on managing the balance between privacy and utility in data publishing, but they only deal with how to prevent the adversaries who know the structural background knowledge from attacking the social network after the social network is released. For example, if the adversary knows the degree of a target individual and the degree of all the neighbors of this individual, then the target individual can be easily identified.

The utility we consider in this thesis is the structural property of the graph, such as the degree of nodes, eccentricity of nodes, diameter of the graph, connectivity of the graph and so on. It is a property of the graph itself instead of a specific representation of the graph.

As anonymization is a conventional technique to preserve the privacy, we present in this thesis new anonymization approaches which preserve both the utility and privacy of the social network. We consider the simple connected graphs where nodes model individuals and edges their relationships. Every relationship is unlabeled, in other words, all the relationships have the same meaning. To protect the social network data, we mask it according to the privacy metric (k, ℓ)-anonymity proposed in [44], specifically $(k,1)$-anonymity. It can evaluate the resistance of social networks against active attacks where the adversaries can actively affect the structure of the published social network graphs and make them easier to decipher. Our transformation approaches anonymize the graph with a few added edges and disturbs as little as possible the social network structural information.

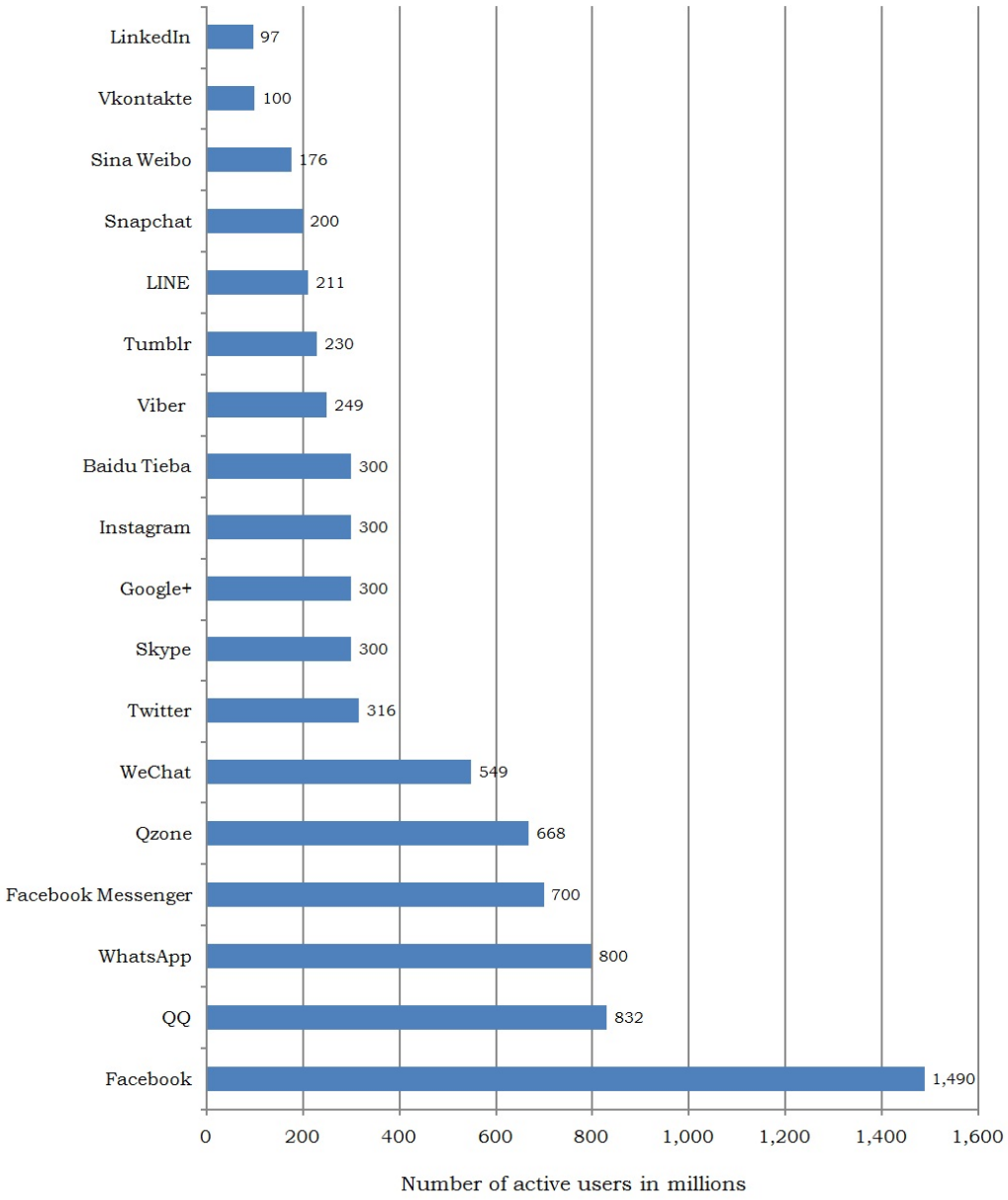


Figure 1.2: Leading social networks worldwide as of August 2015, ranked by number of active users (in millions)[2]

1.2 Our contribution

The general goal of the present thesis is to provide obfuscation techniques aimed at transforming social graphs into graphs resistant to active attacks. In particular, we will focus on social graphs satisfying (1,1)-anonymity, which is the simplest privacy guarantee possible and transform them into graphs satisfying $(k, 1)$-anonymity for $k > 1$. Our contributions are the followings:

- We show theoretical properties of graphs that satisfy (k,1)-anonymity, particularly $k = 1$.

- Based on these properties, we provide a fundamental principle of transforming graphs to ones without any vertices of whom the metric representation with respect to a considered one-vertex subset are different from the other vertices in the same graph. And according to this principal we propose two obfuscation techniques, *Edge-Preserving Approach* (*EPA* for short), which preserves the original connectivity of the graph to a large extent, and *Connectivity-Preserving Approach* (*CPA* for short), which reaches to the goal fast.
- We perform experiments on both synthetic graphs and real-life social graphs in order to evaluate the proposed solutions. The results are that both methods indeed preserve the privacy of users based on the privacy metric $(k, 1)$-anonymity. EPA requires less edge addition operations while CPA arouses less connectivity loss.
- We evaluate one existing active attack, walk-based attack, to social networks against the proposed obfuscation techniques. The results are that the success rate of attacking social networks which are transformed by either EPA or CPA is sharply decreased comparing to the success rate of attacking the original social networks.

1.3 Structure of the thesis

The rest of the thesis is structured as follows. Chapter 2 provides some definitions and notations we will use later. Chapter 3 reviews the literature on privacy-preserving publication of social network data. Chapter 4 shows the theoretical properties of $(k, 1)$-anonymous graphs, particularly (1,1)-anonymous graphs. Chapter 5 presents the formulation of the problem and our obfuscation techniques to solve the problem. Experimental results and the evaluation of our proposed methods CPA and EPA are shown in Chapter 6. Chapter 7 draws conclusions and future work.

Chapter 2

Preliminaries

In this section, we provide some definitions and notations which we will use later and show these properties on a complete graph as an example.

We model a social network graph $G = (V, E)$ as a simple graph where V represents individuals and E their relationships. G is a connected and undirected graph without self-loops, and multiple edges. We often label the vertices with letters v_1,...,v_n. The edges could also have some labels and weights but in this thesis we only consider the simplest form which has neither labels nor weights on both nodes and edges. The graph is connected if all vertices are connected to each other.

Two vertices are adjacent if they are connected by an edge. The two vertices forming an edge are named the *endpoints* of this edge.

The *distance* $d_G(v, u)$ between two vertices v and u in G is the number of edges in the shortest path connecting them. Notice that there may be more than one shortest path between two vertices [51]. If there is no path connecting the two vertices, i.e., if they belong to different connected components, then conventionally the distance is defined as infinite.

The *degree* of a vertex is the number of edges connected to it. An end-vertex is a vertex with degree one. A node with the highest degree in the graph is often called a *hub*.

Connectivity is another important concept of graph theory, which is used in this thesis to measure the information loss when transforming graphs. Globally, the connectivity of a graph is an important measure of its robustness as a network. Formally, it is defined in [35]:

Definition 1. *(Connectivity) The connectivity of a graph is the minimum number of elements (nodes or edges) whose removal makes the graph disconnected.*

If the removed element is a vertex, then *vertex-connectivity* is considered. If the removed element is an edge, then *edge-connectivity* is considered. The vertex-connectivity of a graph is less than or equal to its edge-connectivity. Both are less than or equal to the minimum degree of the graph, since deleting all neighbors of a vertex of minimum degree will disconnect that vertex from the rest of the graph. The local edge-connectivity of two vertices x, y is the size of a smallest edge cut disconnecting x from y. The edge-connectivity of the graph is the size of the smallest edge cut. In this thesis, we regard the edge-connectivity as a metric of information loss and all the mentioned connectivity points to edge-connectivity.

Definition 2. *(Eccentricity) The eccentricity $\epsilon(v)$ of a vertex v in a connected graph is the greatest number of edges in a shortest path between v and any other vertex in the graph.*

The maximum *eccentricity* of any vertex in a graph is the *diameter* of the graph while the minimum is the *radius* of the graph. For a disconnected graph, all vertices are defined to have infinite eccentricity [52].

Definition 3. *(Eccentricity path) Let $G = (V, E)$ be a simple connected graph and v a vertex in $V(G)$. Let u be another vertex in $V(G)$ such that $d_G(v, u) = \epsilon(v)$. The shortest path from v to u is called an eccentricity path of v.*

As there might be more than one u such that $d_G(v, u) = \epsilon(v)$, there might exist more than one *eccentricity path* of v.

Definition 4. *(Graph density) For undirected simple graphs, the graph density is defined as:*

$$Density = \frac{2|E|}{|V|(|V| - 1)}$$

where $|E|$ is the number of edges and $|V|$ is the number of vertices in the graph.

For undirected graphs, the maximum number of edges is $\frac{|V|(|V|-1)}{2}$, so the maximum density is 1 and the minimum is 0 [12]. If the density of G is near 0, then we say that G is a sparse graph. If the density of G is near 1, then we say that G is a dense graph. But there is no absolute limit between sparse graphs and dense graphs. We judge a graph sparse or dense relatively to each other.

The definition of *metric representation* in [44] is an important concept when defining the adversary's background knowledge. Let $G = (V, E)$ be a simple connected graph and $S = \{u_1, ..., u_t\}$ be an ordered subset of vertices of G. The metric representation $r(v|S)$ of every v in $V(G) - S$ with respect to S is the vector of all the distances between v and each vertex in S, i.e., $r(v|S) = (d(v, u_1), ..., d(v, u_t))$. When there is only one vertex w in S, then the metric representation of v with respect to $\{w\}$ is the vector $r(v|\{w\}) = (d(v, w))$.

The concept of *k-antiresolving set* and *metric-antidimension* is the basis of (k, ℓ)-*anonymity*. A subset S of $V(G)$ is called a k-antiresovling set if k is the greatest positive integer such that for every $u \in V(G) - S$ there exists at least $k - 1$ other different vertices who have the same metric representation with respect to S. We call S *1-antiresolving set* if there exists at least one vertex whose metric representation with respect to S is different from the metric representation of any other vertex in G. Another similar concept is 1-resolving set. We call S a *1-resolving set* if the metric representation of every vertex in G with respect to S is different from each other. The *k-metric antidimension* of G, $adim_k(G)$ for short, is the minimum cardinality amongst the k-antiresolving sets in G. Based on these fundamental concepts, the definition of (k, ℓ)-anonymity is following:

Definition 5. *((k, ℓ)-anonymity) A graph G meets (k, ℓ)-anonymity with respect to active attacks if k is the smallest positive integer such that the k-metric antidimension of G is lower or equal than ℓ.*

For every vertices v, u in a complete graph K_n and a subset S of vertices in K_n where $|S| = \{1, ..., n - 1\}$, the structural properties and privacy evaluation is shown in the table 2.1.

Properties	Values
$d_{K_n}(u,v)$	1
Degree of v	$n-1$
Connectivity of K_n	$n-1$
$\epsilon(v)$	1
Diameter of K_n	1
Radius of K_n	1
Density of K_n	1
Metric representation of v $w.r.t$ S	$\overbrace{((1),...,(1))}^{\lvert S\rvert}$
S is a k-antiresolving set	$k = n - \lvert S\rvert$
K_n meets (k,ℓ)-anonymity	$k = n - \lvert S\rvert$, $\ell = \lvert S\rvert$

Table 2.1: The structural properties and the privacy evaluation of K_n

Chapter 3

State of the art

In this chapter, we perform a thorough study of literature review, with emphasis on anonymization methods, two types of attacks and several proposed privacy metrics. Based on these previous work, we show that our study is a new step towards anonymization methods against active attacks.

3.1 Anonymization methods

Social network graphs are published in order to perform useful analysis, such as the prediction of disease transmission or community detection. However once it is released, users data are available to both legitimate researchers and adversaries. It may result in a privacy breach if the adversaries with some auxiliary or background information about the graph have access to the published social network data [13]. The privacy concerns associated with data analysis over social networks have aroused recent research works such as [59][11][57][55][9]. Anonymization is a conventional technique to preserve the privacy of the users in social networks. As mentioned in Chapter 1, there exist some prior work on privacy-preserving techniques of social network graphs with respect to different adversary's background knowledge.

A simple method to anonymize graphs is to remove all the identifiable attributes of individuals such as names, social security numbers and emails before the graph is released and replace them with meaningless identifiers. This simple anonymization method is referred to as *naive anonymization* of a social network [14]. It is named naive anonymization because later this anonymization method was proved by Backstrom et al. to be insufficient to preserve the privacy of social networks. It can only prevent the adversaries who have no auxiliary knowledge about individuals from re-identifying which node corresponds to which individual.

However, in practice the adversary may have access to external information about the individuals in the graph and their relationships. This information may be available through a public source or by the adversary's malicious actions. In [33] Narayanan et al. proposed a re-identification approach showing that one third of users who use both Flickr and Twitter can be re-identified in the completely anonymized Twitter graph with only 12% error rate.

In addition to the adversaries with external information, researchers are not interested in which individual corresponds to which node, instead, the structural properties of the social network graphs. Useful analysis can be carried out even without identifiers. Publishing the naive anonymized graph is roughly akin to providing the original graph, both of which compromise the privacy of individuals.

A study [43] estimated that 87% of the population of the United States can be uniquely identified by means of the combination of seemingly innocuous attributes gender, date of birth and 5-digit zip code. These kinds of attributes whose combination can be used to identify an individual with a significant probability are named quasi-identifiers [32][6].

As naive anonymization is insufficient to protect users' privacy. k-anonymity [36] was proposed to solve the unsolved problem of naive anonymization and amended in [42] for relational data release. It prevents the adversaries from inferring sensitive information of individuals through making the individuals indistinguishable from each other. A released table satisfies k-anonymity if each record on the quasi-identifier attributes is indistinguishable from at least $k-1$ other records in the release. The larger the value of k is, the better the privacy is preserved.

Privacy-preserving methods for relational databases have been studied extensively, several models such as approximation k-anonymity[1], t-closeness[27] , m-invariance [54], ℓ-diversity[29], (k, e)-anonymity [56], privacy skyline[10], anatomy[53] as well as efficient algorithms such as [15][20][23][24][5] have been proposed. But those methods cannot be used to social network data straightforwardly because they fail to account for the interconnectedness of the entities and the background knowledge for the adversaries in social networks is diverse because of its complex structure compared with relational data.

In a social network, nodes with strong structural similarity are indistinguishable by the adversary, even if the adversary is rich in external information. This strong form of structural similarity between nodes is called *automorphic equivalence*. Two different nodes $u, v \in V(G)$ are automorphically equivalent if there exists an isomorphism from the graph to itself where u maps to v. In social networks, k-anonymity was later used for social network data with some variations. A social network graph satisfies k-anonymity if every node in the graph has at least $k-1$ other nodes with identical structural properties with itself. Depending on various structural background knowledge assumption on the adversary, different k-anonymity anonymization methods have been presented:

- **k-subgraph anonymity[50]** This is an anonymization method combining label anonymization with structural anonymization, which was proposed to limit the risk of privacy disclosure in social network data publication. The original graph is partitioned into m unconnected k-subgraphs. In order to maintain the structure of the original graph and reduce the structural information loss in the process of k-subgraph partition, they use a k-subgraph connectivity to record how to connect the separate subgraphs together. Every node in a subgraph has the same label and same degree, so even if the adversary having the knowledge of the degree or the label of certain nodes cannot distinguish one node in a subgraph from the other $k-1$ nodes.

- **k-degree anonymity[28]** Liu et al. considered that the adversary with the background knowledge of the degree of certain nodes can re-identify individuals. They proposed an anonymization model for social networks - a graph meets k-degree anonymity if for every node v, there exist at least $k-1$ other nodes in the graph with the same degree as v. To preserve the utility of the original graph they required to make minimum edge addition operations to the original edges instead of transforming to a complete graph which is useless for any study. They devise simple and efficient algorithms for transforming a graph to k-degree anonymous graph with minimum number of edge modification operations.

- **k-neighbourhood anonymity** Zhou and Pei [58] found that an adversary who

has the knowledge of the neighbours of a target victim and the relationship among the neighbours can re-identify this target victim even though the victim's identity is preserved using conventional anonymization techniques such as the naive anonymization. Zhou and Pei gave the definition of k-neighbourhood anonymity which is that a graph is k-neighbourhood anonymous if for every node v in the graph there exist at least $k-1$ other nodes $v_1, ...v_{k-1}$ of which the subgraph of the neighbours are all isomorphic to the subgraph constructed by the neighbours of v. They presented a practical solution to battle neighborhood attacks, but they only handled 1-neighbourhoods.

It is criticized by Narayanan et al. that all these defenses impose arbitrary restrictions on the information available to the adversary and make arbitrary assumptions about the properties of the social network. Narayanan et. al. argued that the auxiliary knowledge which is likely to be available to the attacker is not restricted to the neighborhood of a single node and the existing models fail to capture self-reinforcing, feedback-based attacks, where re-identification of some nodes provides the attacker with more auxiliary information which is then used for further re-identification.

3.2 Two types of attacks

The extent of attacking depends on what kind of knowledge the adversary owns. The more powerful the knowledge the adversaries hold, the more extent of attacking they can achieve and the more individuals they can re-identify. Now let us see two kinds of attacks.

3.2.1 Passive attack

In a passive attack, an adversary simple observes data as it is presented. In the case of anonymized social networks, passive attacks are carried out by adversaries who try to re-identify individuals only after the anonymized network has been released. In the passive attack described in [3], regular users are able to discover their locations in the graph with the knowledge of the local structure of the network around them. Backstrom et al. imagine that a few existing passive attackers in the graph, who are also able to discover their locations, collude to construct a small coalition X. The users in the coalition know the edges amongst themselves and the names of their neighbors outside X [21][41]. After the graph is released, the coalition runs a search algorithm to find X which is a subgraph consisting of a single node connected to all others. According to the number of vertices outside X the coalition are connected, they locate other users. It is possible that they cannot locate any specific users other than themselves except that a coalition is moderately-sized. However, this attack only works on a small scale: the colluding users can only compromise the privacy of some of their neighbors.

In [33] Narayanan et al. develop another passive attack which takes self-reinforcing, feedback of the re-identified nodes into consideration. This passive attack runs in two stages, which are seed identification and propagation. First, the attacker identifies a small number of seeds, users who have accounts both in the anonymized targeted graph and attacker's auxiliary graph, and maps them to each other. Second, according to the topology of the network and the previously constructed mappings, the propagation step is an extension of seed mapping to new nodes and the new mapping is fed back to the algorithm. Compared to the passive attack in [3], seed indentification and propagation passive attack can be successfully deployed on a very large scale.

3.2.2 Active attack

In contrast, an adversary in an active attack can actively try to affect the data to make it easier to identify individuals by creating new user accounts and links before the anonymized network is released. After the anonymized graph is published, theses new nodes and edges will be presented in the graph. The adversary recovers the subgraph constructed by these nodes and compromises individuals' connections with the help of these links. As users provide large amounts of private information in social networks, it is easy for adversaries to create fake user accounts imitating the data they have and make it undetectable. A research in [38] done by Sophos in 2007 shows that sending a friend request to 200 random Facebook users by a fabricated account named Freddi Staur receives 87 users responses, among which 82 users leak their personal information, such as email address, date of birth, education or current workplace. The reason is that the default settings of Facebook enable friends to view the profiles of each other and most of the users do not change the default settings. The adversary may even pay a handful of users for information about themselves and their friends [26]. All the information helps the adversary to forge trusty users to mislead the legitimate users in the social network.

In [3], Backstrom et al. propose two active attacks, which are, the walk-based attack and the cut-based attack. In both attacks, the adversary plants a well-constructed and uniquely identifiable subgraph in the social network graph, and creates links to arbitrary users which are named targeted users, before the graph is released. For every targeted vertex there is a corresponding subset which dedicates the connecting knowledge between this targeted node and attacker nodes and is different from the other targeted nodes. After the release of the social graph, the adversary retrieves the planted subgraph and identifies the targeted nodes with the help of the connecting knowledge. Normally the success rate is high. There are trade-offs between the two active attacks, which are, the cut-based attack uses fewer new accounts to carry out the attack but is easier to be detected by the curator of the data than the walk-based attack. Furthermore, with $k = \Theta(\log n)$ new accounts where n is the number of vertices in the graph the walk-based attack has the potential to compromise $\Theta((\log n)^2)$ users, while the cut-based attack can only compromise $O(\log n)$. However, in that paper, there is no practical method proposed to counter those attacks.

Another active attack against anonymized social networks, named *Seed-and-Grow*, was proposed in [49]. Different from the active attack in [3], they drop the assumption that the adversary has complete control over the connection between new created subgraph and the rest of the graph, instead, the adversary has a background graph which is in terms of the social connection similar to the target social network graph but the meaning of such connections are different. The adversary also creates a uniquely identifiable subgraph into a social network graph and some link to vertices in the graph(the initial seeds) before it is released and retrieves it after the graph is published. The second *grow* stage is a structure-based vertex matching process from the background knowledge graph to the published graph. *Seed-and-Grow* is a progressive and self-reinforcing strategy starting with the initial seeds and extending the mapping to other vertices after each round.

3.3 Privacy metrics

Wang et al. [46] adopted description logic as the underlying knowledge representation formalism and proposed several formal metrics of anonymity which makes it possible to assess the risk of privacy breach after the social network is published. However, they did

not propose any preventative method if the assessment of the resistance against attacks shows that the release of the data is not safe.

Hay. et al.[19][18] also presented a framework for assessing the privacy risk of publishing anonymized network data. The adversary's knowledge modeled in their studies are vertex refinement queries and subgraph knowledge queries. They proposed a privacy metric k-candidate anonymity which is similar to k-anonymity in relational data to evaluate the risk of re-identification in real data sets.

Trujillo-Rasua and Yero [44] first presented a privacy metric for evaluating the resistance of social graphs under active attacks, which is named (k, ℓ)-anonymity. This privacy measure is created under the circumstance of the active attack, where the ability of adversaries is more powerful than that under passive attacks. The adversary' s background knowledge is modeled as a distance vector of each vertex with respect to attacker nodes, named metric representation.

In that paper, k is a privacy threshold and ℓ is an upper bound on the number of attacker nodes in the network which is much lower than the total number of nodes in the network. For each subset S_i of ℓ attacker nodes there exists a correspondingly greatest k_i such that every vertex in $V(G) - S_i$ has at least $k_i - 1$ other vertices who have the same metric representation with respect to S_i. A graph meets (k, ℓ)-anonymity with respect to active attacks if k is the smallest positive integer among all these k_i when the number of attacker nodes is equal or lower to ℓ.

Considerable works have been done to anonymize social network graphs against adversaries with auxiliary information which can preserve both the privacy of users and the utility of the graphs. However, there is no study on anonymizing social network graphs against active attacks. In this thesis, we focus on transforming a graph to a privacy-preserving graph resisting against active attack and at the same time preserve as much utility of the original graph as possible. The previous mentioned k-neighborhood anonymity [58] imposes a strong structural requirement on the graph, which is difficult to achieve without a huge information loss and a large amount of computation. So Zhou and Pei only deal with the particular case of d-neighborhoods where $d = 1$. Similarly, in this thesis we focus on $(k, 1)$-anonymity rather than on the more general (k, ℓ)-anonymity concept driven by utility concerns.

Chapter 4

Theoretical properties of $(k, 1)$-anonymous graphs

In this thesis, we consider that there is only one attacker node, which is the simplest case of (k,ℓ)-anonymity proposed in [44]. In this chapter, we show the theoretical properties of (1,1)-anonymous graphs and the conditions to be (k,1)-anonymous graphs.

4.1 (1,1)-anonymous graphs

A graph G meets $(1, 1)$-anonymity with respect to active attack if $admin_1(G) = 1$. If the metric representation of v in G with respect to S is different from that of any other nodes in $V - S$, we name v as an eye-catching node:

Definition 6. *(Eye-catching node) Let $G = (V, E)$ be a simple connected graph and let $S = \{u_1, ..., u_t\}$ be an ordered subset of vertices of G. A node $v \in V(G)$ is called an eye-catching node with respect to S if there does not exist another node u in $V(G) - S$ such that $r(v|S) = r(u|S)$.*

Note that, if v is an eye-catching node with respect to S, then S is a 1-antiresolving set. If a graph has a 1-antiresolving set, then it satisfies $(1, 1)$-anonymity. Then we have the next proposition:

Proposition 1. *Given a simple connected graph $G = (V, E)$ and a vertex u in $V(G)$, if there exists an eye-catching node in $V - \{u\}$ with respect to $\{u\}$, then G satisfies $(1, 1)$-anonymity.*

Proof. As v is an eye-catching node with respect to $\{u\}$, then $\{u\}$ is a 1-antiresolving set. The 1-metric antidimension of G is 1. Then 1 is the smallest positive integer such that 1-metric antidimension of G is 1. According to the definition of $(1, 1)$-anonymity, G satisfies $(1, 1)$-anonymity. □

As the presence of eye-catching nodes with respect to the set of a single node implies $(1, 1)$-anonymity, let us see several lemmas below which provides the conditions of vertices to be eye-catching nodes.

Lemma 1. *For every end vertex u and its neighbour v, it holds that v is an eye-catching node with respect to $\{u\}$.*

Proof. If u is an end vertex with a neighbour vertex v, then for all the vertices in the graph there is only one vertex whose distance to u is 1 and that is its neighbour v. □

Lemma 2. *Let $G=(V,E)$ be a simple connected graph $G=(V,E)$ and v a vertex in $V(G)$. Let another vertex u in $V(G)-\{v\}$ such that $d(v,u)=\epsilon(v)$. If there does not exist $w\in V(G)$ such that $d(v,w)=\epsilon(v)$, then u is an eye-catching node with respect to $\{v\}$.*

Proof. Let v be a vertex in $V(G)$. According to the definition of eccentricity in a graph, for every vertex $x\in V(G)$ the eccentricity of V satisfies that $\epsilon(v)\geq d_G(v,x)$. If there exists one and only one vertex u whose distance to v is equal to $\epsilon(v)$, then for $\forall w\in V(G)-\{u\}$ it holds that $d_G(v,w)<\epsilon(v)=d_G(v,u)$, which implies that u is an eye-catching node with respect to $\{v\}$. □

Observation 1. *In any cycle graph C_n with even order, for every pair of diametral vertices u,v in $V(C_n)$, u is an eye-catching node with respect to $\{v\}$.*

Observation 2. *Any complete bipartite graph $K_{r,2}$ with two disjoint sets U and V, where $|U|=r>2$ and $|V|=2$. Let u,v be the vertices in V, then it holds that u is an eye-catching node with respect to $\{v\}$.*

Observation 3. *In a rooted tree, if there is only one leaf in a level, then this leaf has the longest depth to the root and is an eye-catching node with respect to the root.*

4.2 $(k,1)$-anonymous graphs

After knowing the conditions of vertices to be eye-catching nodes, it can be inferred easily the conditions for graphs to be $(k,1)$-anonymous for $k>1$.

Proposition 2. *Given a simple connected graph $G=(V,E)$, if there is no eye-catching node with respect to any one-vertex subset of $V(G)$, then G satisfies $(k,1)$-anonymity for $k>1$.*

The following theorem shows the conditions of a graph with a bridge to satisfy $(k,1)$-anonymity.

Theorem 1. *Let $G_1=(V_1,E_1)$ and $G_2=(V_2,E_2)$ be two graphs satisfying (k_1,1)-anonymity and (k_2,1)-anonymity for $k_1,k_2>1$ respectively. Let $G=(V,E)$ be the graph obtained by adding an edge between two random vertices $u\in V_1$ and $v\in V_2$. Then G satisfies (k,1)-anonymity for $k>1$ if and only if for every vertex $x\in V_1$, and $y\in V_2$ it holds that $\epsilon_{G_1}(x)>d(x,u)$ and $\epsilon_{G_2}(y)>d(y,v)$.*

Proof. ($\Rightarrow$) Let us assume that G is $(k,1)$-anonymous for $k>1$. In order to find a contradiction let us say that, without loss of generality, there exists $x'\in V_1$ such that $\epsilon_{G_1}(x')\leq d_{G_1}(x',u)$.

According to the definition of eccentricity in graph, the eccentricity of x satisfies that $\epsilon_{G_1}(x)\geq d_{G_1}(x,u)$ for every $x\in V_1$. Therefore, $\epsilon_{G_1}(x)=d_{G_1}(x,u)$. Let w be a vertex in V_1 different to x. Then $d_{G_1}(x,w)\leq\epsilon_{G_1}(x)=d_{G_1}(x,u)$. Because there is a single path from any vertex in V_1 to v in G and such path passes through u, then $d_G(x,v)=d_G(x,u)+1$, which implies that $d_G(x,v)>d_G(x,w)$. Consequently, there does not exist a vertex in V_1 whose distance is equal to $d_G(x,v)$. It is easy to note that, for every vertex $w'\in V_2-\{v\}$, it holds that $d_G(x,w')=d_G(x,v)+d_G(v,w')>d_G(x,v)$. As a result, $\{x\}$ is a 1-antiresolving set, which is a contradiction with the assumption that G is $(k,1)$-anonymous for $k>1$.
($\Leftarrow$) If for every vertex $x\in V_1$, $y\in V_2$ it holds that $\epsilon_1(x)>d(x,u)$ and $\epsilon_2(y)>d(y,v)$, then it can imply that G is $(k,1)$-anonymous for $k>1$.

Because for every vertex $x \in V_1$ it holds that $\epsilon_{G_1}(x) > d_{G_1}(x,u)$ and in G it holds that $d_G(x,v) = d_G(x,u) + 1$, then $\epsilon_{G_1}(x) \geq d_G(x,v)$ and there exists $y \in V_1$ such that $\epsilon_{G_1}(x) = d_{G_1}(x,y) \geq d_G(x,v)$ and exists z in the $x-y$ path such that $d_G(x,z) = d_G(x,v)$, which implies that v is not eye-catching node with respect to any vertex in V_1. The same with u with respect to any vertex in V_2. Because G_2 is (k_2,1)-anonymous for $k_2 > 1$, then for every vertex $w' \in V_2 - \{v\}$ there exists at least $k_2 - 1$ different vertices in $V_2 - \{v\}$ whose distances to v is equal to $d_{G_2}(v,w')$. Because there is a single path from any vertex in V_1 to w' in G and such path passes through u, v, then for every $x \in V(G_1)$, it holds that $d_G(x,w') = d_G(x,u) + 1 + d_G(v,w')$, which implies that for every vertex $w' \in V_2 - \{v\}$ there exists at least $k_2 - 1$ different vertices in $V_2 - \{v\}$ whose distances to x in G equal to $d_G(x,w')$. Given that G_1 satisfies $(k_1,1)$-anonymity, for every vertex $w \in V_1 - \{x\}$ there exists at least $k_1 - 1$ different vertices in $V_1 - \{x\}$ whose distances to x equal to $d_{G_1}(x,w)$. As a result, for every vertex $w'' \in V - \{x\}$, there exists at least $\min\{k_1,k_2\}$ different vertices in $V - \{x\}$ whose distances to x equal to $d_G(x,w'')$. This will reach to the same result if $x \in V_2$. So according to the definition of (k,1)-anonymity for $k > 1$, G satisfies (k,1)-anonymity where k=$\min\{k_1,k_2\}$. □

Chapter 5

Obfuscation techniques

In this chapter, we first specify adversaries' background knowledge and the privacy metric we consider in this thesis in section 5.1 and 5.2. In section 5.3, we propose the problem on which we are focusing and which is solved by our solutions in section 5.4.

5.1 Adversary's background knowledge

Before defining the problem of privacy preservation in the published graph, we need to formulate the background knowledge that an adversary may use to attack the privacy of users. According to this knowledge, we find the corresponding method to preserve the privacy. In a passive attack, the adversary's background knowledge is modeled as structural relations on the network, such as vertex degrees, or neighborhood.

For example, the adversary's knowledge modeled in [28] is the degree of target nodes. In order to preserve the privacy, the graph is anonymized by edge addition and deletion operations to make every vertex in the graph to have at least $k-1$ other vertices who have the same degree with it.

The adversary's knowledge modeled in [58] is the neighborhood of a target victim and the relationship among the neighbors. Zhou and Pei propose an improved k-anonymity model, that is, for every vertex u in the graph, there are at least $k-1$ other vertices whose neighborhoods are isomorphic to the neighborhood of u. Their new model prevents the adversary from attacking the privacy with neighborhood knowledge.

Our work is based on the privacy measure proposed in [44], so we model the adversary's background knowledge as in [44]. The adversaries in [44] not only have a global view of the network, such as the structural properties of the whole graph, but also a local view, such as the properties from the perspective of a single static vertex. The adversary's background knowledge about a target node u is defined as the metric representation of u with respect to any subset of potential attacker nodes. Since we only consider the case that there is only one attacker node in the published graph, in this thesis, the adversary's background knowledge is the metric representation of a target node u with respect to a potential set $\{v\}$ of the attacker node v. This background knowledge, the metric representation of certain nodes, means that the adversary knows the partial topological structure of the network.

5.2 Privacy metric

The privacy metric (k, ℓ)-anonymity considers the worst case if there exist ℓ potential attacker nodes in the graph, what the value of the smallest k is such that every vertex

cannot be distinguished with probability higher than $1/k$. The means to distinguish vertices is the metric representation of vertices, which is a distance vector. For example, if the distance between two vertices x and y is 3, then the metric representation of x with respect to $\{y\}$ is a vector (3), symmetrically, the metric representation of y with respect to $\{x\}$ is also (3).

Table 5.1 shows the metric representation of each vertex in Figure 5.1 with respect to every possible attacker set S_i whose size is 1. From the second column, we can see that every vertex has another one and only one vertex who has the same metric representation with respect to each S_i. When ℓ is 1 the smallest positive integer k among all the k_i is 2, which means that the graph is (2,1)-anonymous.

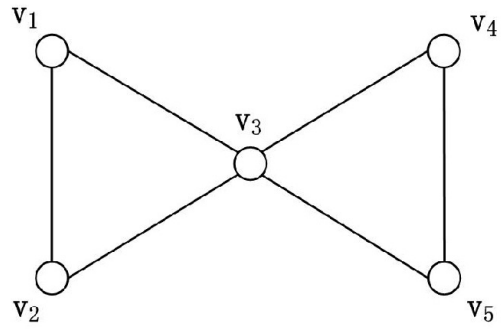


Figure 5.1: An example to explain why we choose ℓ=1

S_i	metric representation of each vertex in $V(G) - S_i$ *w.r.t* S_i	k_i
$\{v_1\}$	(1) (1) (2) (2)	2
$\{v_2\}$	(1) (1) (2) (2)	2
$\{v_3\}$	(1) (1) (1) (1)	2
$\{v_4\}$	(2) (2) (1) (1)	2
$\{v_5\}$	(2) (2) (1) (1)	2

Table 5.1: Adversary's knowledge for graph in Figure 5.1 when ℓ=1

However, when the expected number of attacker nodes increases to 2, as it is shown in Table 5.2, the smallest value k among all these k_i is 1. In this case, the graph satisfies (1,2)-anonymity. Therefore, the graph meets (1,2)-anonymity and (2,1)-anonymity.

Aiming at improving the privacy level of this graph when there are two attacker nodes, the value of k_i with respect to the set of two attacker nodes should be increased. Take the subset $\{v_1, v_2\}$ for example, the metric representation of v_3 with respect to $\{v_1, v_2\}$ is (1,1) while that of v_4 and v_5 with respect to $\{v_1, v_2\}$ are the same (2,2). Without adding any nodes, in order to make the metric representation of v_3, v_4, v_5 with respect to $\{v_1, v_2\}$ the same we need to add four edges, $v_1 - v_4$, $v_1 - v_5$, $v_2 - v_4$ and $v_2 - v_5$. After that, every vertex in $V(G) - \{v_1, v_2\}$ has the same metric representation (1,1) with respect to $\{v_1, v_2\}$. However, it becomes a complete graph, in Figure 5.2, which is useless for deeper study.

Therefore, we think the privacy measure proposed in [44] is so strong that transforming a graph to satisfy (k, ℓ)-anonymity costs a lot of structural breach to the original graph. $(k, 1)$-anonymity for $k > 1$ is a simple form of (k, ℓ)-anonymity which can not only protect

S_i	metric representation of each vertex in $V(G) - S_i$ *w.r.t* S_i	k_i
$\{v_1\}$	(1) (1) (2) (2)	2
$\{v_2\}$	(1) (1) (2) (2)	2
$\{v_3\}$	(1) (1) (1) (1)	2
$\{v_4\}$	(2) (2) (1) (1)	2
$\{v_5\}$	(2) (2) (1) (1)	2
$\{v_1, v_2\}$	(1,1) (2,2) (2,2)	1
$\{v_1, v_3\}$	(1,1) (2,1) (2,1)	1
$\{v_1, v_4\}$	(1,2) (1,1) (2,1)	1
$\{v_1, v_5\}$	(1,2) (1,1) (2,1)	1
$\{v_2, v_3\}$	(1,1) (2,1) (2,1)	1
$\{v_2, v_4\}$	(1,2) (1,1) (2,1)	1
$\{v_2, v_5\}$	(1,2) (1,1) (2,1)	1
$\{v_3, v_4\}$	(1,2) (1,2) (1,1)	1
$\{v_3, v_5\}$	(1,2) (1,2) (1,1)	1
$\{v_4, v_5\}$	(2,2) (2,2) (1,1)	1

Table 5.2: Adversary's knowledge when ℓ=2 for the graph in Figure 5.1

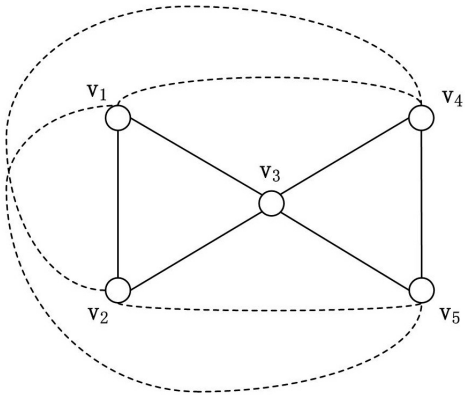


Figure 5.2: Transforming the graph in Figure 5.1 to satisfy $(k, 2)$-anonymity.

the privacy of the users in social network but also preserve the utility of the graphs. So in this thesis, we focus on $\ell = 1$ as a first step to satisfy some sort of privacy against active attacks. Indeed, we show later in Chapter 6 that graphs satisfying $(k, 1)$-anonymity also resist attacks where the adversary controls more than one node in the network, i.e., $\ell > 1$. For future work, we plan to address the problem with $\ell > 1$. However, as shown in the example above, the added noise to the graph seems to significantly increases in this case. Thus, we will also study the relaxed notions of (k, ℓ)-anonymity. We discuss more on this in Section 7.

5.3 Problem formulation

After defining the adversary's background knowledge, it is easy to note that if the distance from a vertex u in the graph to another attacker node v in the graph is unique, then the adversary can easily re-identify u after publication, which compromises the privacy of

the corresponding user of u. The more such unique distance exists, the more privacy is likely to be compromised.

In the *Seed and Grow* algorithm [49], the adversary plants a node v in the graph before it is released. After the graph is published, the adversary retrieves v and identifies its neighbors as the initial seeds, which provides a firm ground for further identification. Given the background knowledge defined previously, eye-catching nodes with respect to v, i.e., u, can also be identified and serves as the initial seeds to be grown, which increases the number of initial seeds. From the comparative study in that paper, we know that the more initial seeds, the higher ratio of correctly identified nodes. It implies that a graph satisfying $(1, 1)$-anonymity helps the attack.

To avoid the augment of initial seeds which is caused by eye-catching nodes, we can modify the graph to make every vertex in the graph has at least $k - 1$ other vertices who have the same distance to v with its distance to v, which can reduce the number of initial seeds to some extent. This is the privacy measure we are going to use in this thesis named $(k, 1)$-anonymity. According to the definition of (k,ℓ)-anonymity in [44], a graph meets $(k, 1)$-anonymity for $k > 1$ with respect to active attack if k is the smallest positive integer such that the k-metric antidimension of G is equal to 1. If every node in the graph has at least $k - 1$ nodes who has the same metric representation with respect to $\{v\}$, then this graph satisfies $(k,1)$-anonymity.

We use the $(k, 1)$-anonymity privacy metric mentioned previously to define the graph anonymization problem. The input to the problem is a simple connected graph $G = (V, E)$ which is $(1, 1)$-anonymous. The requirement is to use a set of graph-modification operations on G to construct a $(k, 1)$-anonymous graph $G' = (V', E')$ for $k > 1$ which is structurally similar to G. We restrict the graph-modification operations to edge additions without any edge deletions, which is more probable to break the connectivity of G. Furthermore, we require that the new graph has the same set of nodes as the original graph, that is, $V' = V$.

Formally, we define the graph anonymization problem as follows:

Problem 1. *Given a graph $G = (V, E)$ satisfying $(1, 1)$-anonymity, modify G via a relatively minimum number of edge-addition operations in order to construct a new graph $G' = (V, E')$ such that G' satisfies $(k,1)$-anonymity with $k > 1$.*

It is usually possible to transform G to the complete graph K_n, in which the metric representation of all the nodes in G with respect to attacker node is $(n - 1)$-dimensional vector (1,1,...,1) and K_n is $(n - 1,1)$-anonymous. Although this kind of transformation can preserve privacy, it would make the anonymized graph useless for any study. As any edge addition operation breaks the structure of the original graph, considering utility concerns we ask for as few edges as possible to be added. So we proposed an additional requirement that the relatively minimum number of edge-additions is made. In this way, we preserve the utility of the original graph while at the same time we meet the $(k, 1)$-anonymity requirement.

5.4 Our solutions

In this section, before solving the problem we first show two preparatory work on end-vertices elimination and locating eye-catching nodes. Secondly, we provide two solutions to the proposed problem and show the tradeoffs between them.

5.4.1 End-vertices elimination

According to Lemma 1, a graph with at least one end-vertex does not satisfy $(k,1)$-anonymity where $k > 1$. So before transforming a graph into a private graph, we need to deal with these end-vertices.

Firstly, we need to know why end-vertices exist and who they are. In real life social network, *i.e.*, *Facebook*, where nodes and edges are on behalf of people and friendship respectively, these end-vertices represent people who have only one friend. There are many reasons for this situation. One is that they are new on Facebook. Another may be that they just created an account, added one friend and gave up using it. Those accounts which have been given up are valueless for analysis. In this thesis, we treat end-vertices as new accounts on social network. At the same time, we speculate that they will know new friends soon by recommendation from both the Internet and real life.

A trivial solution to deal with end-vertices is to delete all of them in the graph. However, this approach fails in low density graphs such as paths and trees, because every vertex in the paths and trees will be deleted in this case. Another approach is to make the degree of these end-vertices higher than 1 by adding new edges to each end-vertex. To achieve this, we should decide how to establish these new connections optimally.

According to [4], there is a high probability that a new vertex will be linked to a vertex that already has a large number of connections, which is named hub afterwards in [18]. A hub is a highly connected node in the network. In a graph of email connections, a hub represents an influential individual. For example, in a graph of email connections, it is more possible that the end-vertex which represents another ordinary individual sends an email to this influential person than to other ordinary people.

The studies on high *"clustering"* in [48] give us more idea to deal with end vertices. If the distance between this end-vertex and the hub is far in the network, it looks unrealistic that there is an edge between far-away nodes, even in the future published network. *Watts* proposed that there is a high probability that two vertices will be connected directly to each other if they have another common vertex.

It holds also in real life situation. If two strangers have a common friend, then it is probable that they will be recommended by their common friend to each other. It is also possible these two strangers meet each other by accident without any recommendation. But in general, this possibility of recommendation is higher than the latter case.

Let v be an end-vertex in G and u the neighbor of v. We do not consider the case that there exist no other neighbors of u except v because the size of social network graphs is usually large. Taken both statements in [4] and [48] into consideration, we choose the neighbor of u who has highest degree as the other endpoint of the new edge, since it has one common neighbor with the end-vertex v and the highest possibility to be linked. If all the other neighbors of u except v are also end-vertices, then we select one randomly from them as the other endpoint of the new edge. So our decision is to add an edge between v and the neighbor of u with highest degree except v (Algorithm 1).

5.4.2 Locating eye-catching nodes

The neighbor u of an end-vertex v is just one of the eye-catching nodes with respect to $\{v\}$ in the graph. It is because the distance between u and v is different from the distance from other vertices to v, more concretely in this case it is smaller. While in other cases, graphs without end-vertices may still have eye-catching nodes, e.g., Figure 5.3 is a graph without end-vertices. With respect to $\{v_0\}$, the metric representation of all the other nodes are (1), (2), (3), (4), (5), (6), (7), (7), (1), (2), (2) corresponding the vertices from v_1 to v_{11}. For v_1 there exists another vertex v_9 such that $r(v_1|\{v_0\}) = r(v_9|\{v_0\})$. For v_2 there

Algorithm 1 Given a graph $G = (V, E)$, this algorithm outputs a graph $G' = (V, E')$ which has no end-vertex

Let S_e be the set of all the end-vertices in G.
Let $E'(G') = E(G)$.
while $S_e \neq \emptyset$ **do**
  for every v in S_e **do**
    Let u be the neighbor of v.
    Let w be a neighbor of u in $V(G) - \{v\}$ with the largest degree.
    Add an edge e between v and w.
    $S_e = S_e - \{v\}$
    $E'(G')$=$E'(G') \cup \{e\}$
  end for
end while
Output G'.

exist another two vertices v_{10} and v_{11} such that $r(v_2|\{v_0\}) = r(v_{10}|\{v_0\}) = r(v_{11}|\{v_0\})$. For v_7 there also exists another vertex v_8 such that $r(v_7|\{v_0\}) = r(v_8|\{v_0\})$. However, for v_3, v_4, v_5 and v_6 there is no other vertex whose metric representation is the same with its metric representation, respectively. Therefore, in this example v_3, v_4, v_5 and v_6 are eye-catching nodes with respect to $\{v_0\}$. So this graph only satisfies $(1, 1)$-anonymity.

It can be inferred from Proposition 2 that only graphs without any eye-catching nodes with respect to each one-vertex set can satisfy $(k, 1)$-anonymity for $k > 1$. So removing all eye-catching nodes with respect to each vertex in the graph is a means to preserve the privacy of the individuals.

In Figure 5.3, all the eye-catching nodes (shaded vertices) with respect to $\{v_0\}$ have unique distance values because the distance of the vertices on other paths to v_0 is smaller than these values and the distance of vertices on the same path to v_0 is either larger or smaller than these values. The following lemma shows a predication of the locating path of eye-catching nodes and gives the proof of the correctness of our predication.

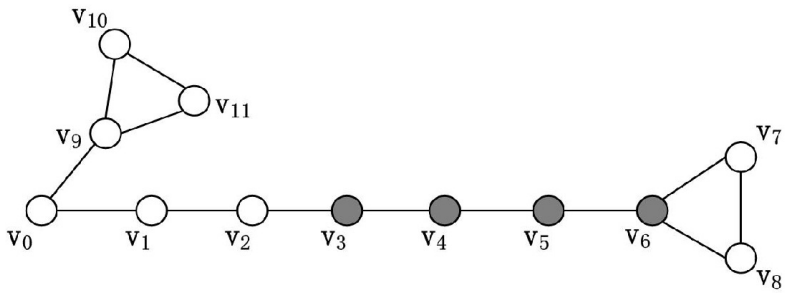


Figure 5.3: An example

Lemma 3. *Let $G = (V, E)$ be a simple connected graph and v a vertex in $V(G)$. If there exists an eye-catching node with respect to $\{v\}$, then it is located in one of the eccentricity paths of v.*

Proof. Let us assume that there exists one eye-catching vertex w in $V(G)$ with respect to $\{v\}$ such that it is not located in any eccentricity path of v. According to the definition

of eccentricity in graph, the eccentricity of v satisfies that $\epsilon(v) \geq d(v, y)$ for every $y \in V(G)$, then it holds that $\epsilon(v) \geq d(v, w)$. It means that there exists one vertex w' in the eccentricity paths of v such that $d(v, w') = d(v, w)$, which implies that w is not an eye-catching node with respect to $\{v\}$, which is a contradiction with the assumption that w is an eye-catching vertex with respect to $\{v\}$. □

According to the definition of the eccentricity path, for a vertex v it is possible that there exist more than one vertex u such that $d(v, u) = \epsilon(v)$. In this case, assume w is the other vertex such that $d(v, w) = \epsilon(v)$, if there still exist one eye-catching node x with respect to $\{v\}$, then x is located in the common path of the shortest $v - u$ path and the shortest $v - w$ path. This is reasonable: If x is located only in the shortest $v - u$ path instead of the common path of two eccentricity paths of v, there exists another vertex y in the shortest $v - w$ path such that $d(v, x) = d(v, y)$ which leads to x not an eye-catching node with respect to $\{v\}$.

5.4.3 Our solutions

After knowing where the eye-catching nodes with respect to a subset $\{v\}$ of $V(G)$ are located, to achieve our goal the next step is to eliminate them.

In order to continue our study we need to introduce some terminology and notation.

When it comes to the *predecessor* and the *descendant* of a vertex, the corresponding path and the direction is required. The predecessors of v in the shortest path from x to y are all the vertices in this shortest $x - y$ path whose distance to x is smaller than the distance between x and v. The immediate predecessor of v from the shortest path from x to y is the vertex u in this shortest $x - y$ such that $d(x, u) = d(x, v) - 1$. Similarly, the descendants of a vertex v in the shortest path from x to y are all the vertices in this path whose distance to x is larger than the distance between x and v. The immediate descendant of v in the shortest path from x to y is the vertex u such that $d(x, u) = d(x, v) + 1$. In this thesis, when we mention the predecessor or descendant of a vertex in the eccentricity path of v it means the direction is from v to the vertex u such that $d(v, u) = \epsilon(v)$ instead of from u to v.

Let $d_{f_{eye}}(v)$ and $d_{n_{eye}}(v)$ be two distances to v from the farthest eye-catching node v_f and nearest one v_n with respect to $\{v\}$ respectively and generally it holds that

$$d_{f_{eye}}(v) \geq d_{n_{eye}}(v). \tag{5.1}$$

If there is no eye-catching node with respect to $\{v\}$, we say that

$$d_{f_{eye}}(v) = d_{n_{eye}}(v) = 0. \tag{5.2}$$

If there is only one eye-catching node with respect to $\{v\}$, we say that

$$d_{f_{eye}}(v) = d_{n_{eye}}(v) \neq 0. \tag{5.3}$$

Since all the graphs we use now are ones without any end-vertices, it holds that

$$d_{n_{eye}}(v) > 1 \quad \text{and} \quad d_{f_{eye}}(v) > 1. \tag{5.4}$$

Given that we are aiming to remove all eye-catching nodes with respect to a potential set of one attacker node, which is a one-vertex subset of the graph, it is in accordance with Equation 5.2 for every v in the graph.

To achieve Equation 5.2, we need to know what kinds of graphs does not have eye-catching nodes with respect to each one-vertex subset. It can be inferred from Observation 1 that any cycle graph C_n with odd order does not have any eye-catching node with respect to any one-vertex subset of $V(C_n)$.

Lemma 4. *A cycle graph C_n with an odd order satisfies (2,1)-anonymity.*

Proof. For every vertex v in $V(C_n)$ the distances from other $n-1$ even number of vertices to v are pair-wise from 1 to $(n-1)/2$ with an interval of 1. So every vertex in C_n has another vertex who has the same metric representation with respect to $\{v\}$, C_n satisfies (2,1)-anonymity. □

This gives us inspiration of eliminating all the eye-catching nodes with respect to each subset $\{v\}$ of $V(G)$. If all the eye-catching nodes with respect to $\{v\}$ are included into a cycle with odd order, then they are not eye-catching nodes with respect to $\{v\}$ any more. In order to create an odd number circle in the graph, an new edge is needed to be added. It is required that before adding this new edge the distance between two endpoints of the new edge is even with which the cycle has odd number of vertices.

Theorem 2. *Let $G = (V, E)$ be a simple connected graph and v a vertex in G. Let S_e be the set of all the eye-catching nodes in $V(G)$ with respect to $\{v\}$ where $|S_e| \neq 0$. Let $G' = (V, E')$ be a graph obtained by creating a cycle through one edge addition operation to G. If all the following conditions are satisfied*

- *The number of vertices in the cycle is odd,*
- *The cycle includes all the vertices in S_e,*
- *v_n is not an endpoint of new edge,*

then in G' all the vertices in S_e are not eye-catching nodes with respect to $\{v\}$.

Proof. Let x, y be two endpoints of this new edge. There are two possibilities for the relationship between v and the cycle depending on whether v is included in the cycle:

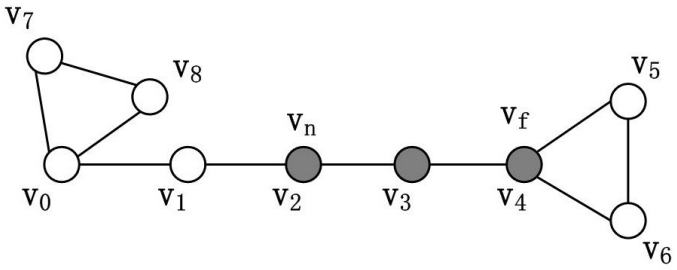
- v is in the cycle. As is proved in Lemma 4, with respect to $\{v\}$ every vertices in the cycle except v can not be distinguished with probability higher than $1/2$. So in G' all the vertices in S_e are not eye-catching nodes with respect to $\{v\}$.
- v is not in the cycle. According to Lemma 3, all the vertices in S_e are located in one of the eccentricity paths of v. Since the cycle includes all the vertices in S_e, in G the vertices in S_e are also located in the shortest $x - y$ path. As v is not in the cycle and x, y are also located in one of the eccentricity paths of v in G, both x and y are descendants of v. Let us assume that $d(v, x) < d(v, y)$. In G', x is the nearest vertex to v compared with other vertices in the cycle. As v_n is not one endpoint of the new edge, x is one predecessor of v_n. With respect to $\{x\}$ there is no eye-catching nodes in the cycle. If all the distances to x from the other vertices in the cycle plus $d(v, x)$ are the distances from the vertices in the cycle to v. The distances are also pair-wise. For every vertex in the cycle, there exists another vertex who has the same metric representation with respect to $\{v\}$. So in G' all the vertices in S_e are not eye-catching nodes with respect to $\{v\}$.

□

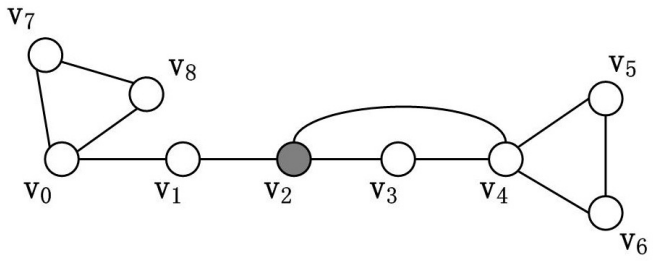
The example below shows the reason why v_n cannot be the endpoint of the new edge:

Example 1. *In Figure 5.4(a) there are three eye-catching nodes with respect to $\{v_0\}$ which are v_2, v_3 and v_4. Among these three vertices v_2 is the nearest eye-catching node with respect to $\{v_0\}$ while v_4 is the farthest one. If v_2 is an endpoint of the new edge, then after adding an edge between v_2 and v_4 as is shown in Figure 5.4(b), the metric*

representation of v_2 with respect to $\{v_0\}$ is still different from that of other vertices. This new circle $v_2 - v_3 - v_4 - v_2$ does not have the function of eliminating all the eye-catching nodes with respect to $\{v_0\}$.



(a) The original graph



(b) After adding an edge

Figure 5.4: An example to show v_n is not a suitable endpoint of the new edge.

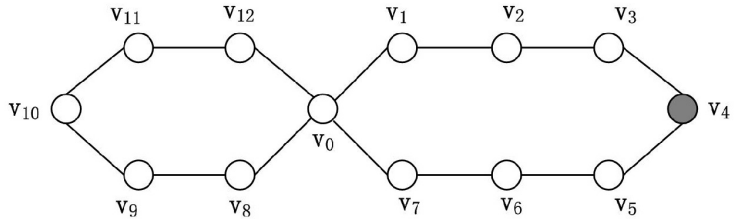
If we create a cycle as mentioned in Theorem 2 for the eye-catching nodes with respect to every one-vertex subset $\{v\}$ in the graph, this will eliminate all the existing eye-catching nodes locally with respect to the considered $\{v\}$. Globally this action may create new eye-catching node with respect to any one-vertex subset. Even with respect to $\{v\}$ there may appear new eye-catching nodes in the other paths except any eccentricity paths of v, let alone with respect to other one-vertex subset. Figure 5.5 is an example to show this.

Example 2. *In Figure 5.5(a), v_4 is the only eye-catching node with respect to $\{v_0\}$. We create a cycle according to the statement in Theorem 2 by adding an edge between v_0 and v_4 as is shown in Figure 5.5(b). In the new obtained graph v_4 is not an eye-catching node with respect to $\{v_0\}$, instead, v_{10} is a new eye-catching node with respect to v_0.*

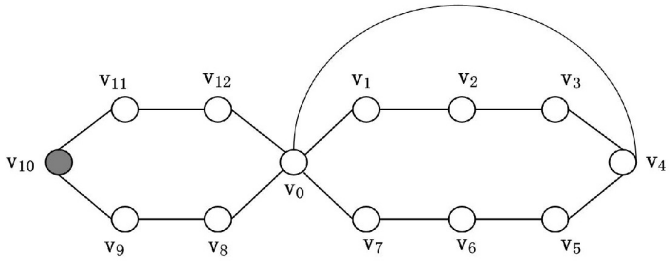
Based on this feasible solution we propose two specific approaches to solve the proposed problem in order to achieve our goal. They are *Edge-Preserving Approach*, *EPA* for short and *Connectivity-Preserving Approach*, *CPA* for short.

Edge-preserving approach (EPA)

Solution 1. *(EPA) Given a simple connected graph $G = (V, E)$ with an attacker node v. Let x be one of the immediate descendant of v in the eccentricity paths of v, and w be*



(a) The original graph



(b) After creating a cycle

Figure 5.5: An example to show Theorem 2 can only eliminate eye-catching nodes locally not globally.

a vertex such that $d(v, w) = \epsilon(v)$. Our proposal is to create a cycle by adding an edge between v and w if $\epsilon(v)$ is even or between x and w if $\epsilon(v)$ is odd.

Algorithm 2 Given a graph $G = (V, E)$ this algorithm outputs a graph $G' = (V, E')$ which satisfies (k,1)-anonymity for $k > 1$

Let $E'(G') = E(G)$.
for $\forall v \in V(G)$ **do**
  BFS[3](v)
  Let w be the vertex such that $d(v, w) = \epsilon(v)$ and x an immediate descendant of v in one of the eccentricity paths of v.
  if $\epsilon(v)$ is even **then**
    Add an edge e between v and w
    $E'(G')$=$E'(G') \cup \{e\}$
  else
    Add an edge e between x and w
    $E'(G')$=$E'(G') \cup \{e\}$
  end if
end for
Output G'.

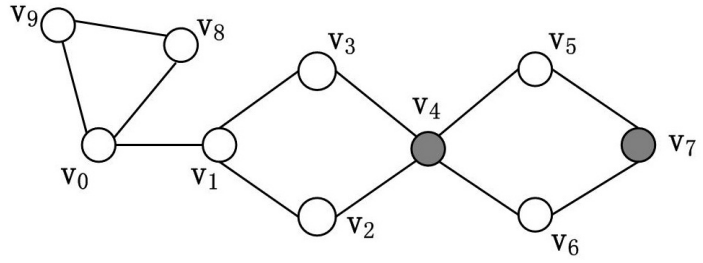
EPA consists in creating a big cycle with respect to each potential one-vertex subset in its eccentricity path. The biggest cycle in the eccentricity path of v is created by adding an edge between v and w if the eccentricity of v is even. If it is odd, we need to remove x out of the cycle instead of w to reach an odd number of vertices in the cycle.

[3]Breadth-First Search, BFS for short, is an algorithm for traversing or searching tree or graph data structure.

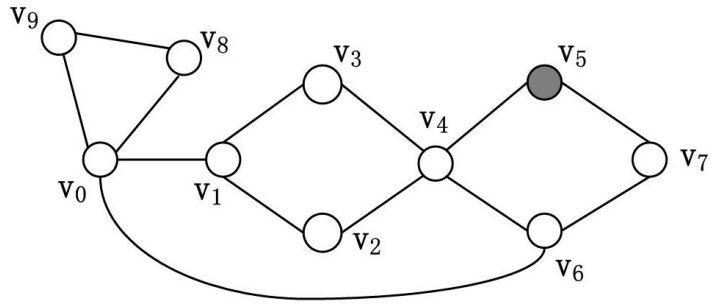
Since it is possible that w is the farthest eye-catching node with respect to $\{v\}$. If we add an edge between v and the immediate predecessor of w, then the cycle does not include all the eye-catching nodes with respect to $\{v\}$ which has the possibility of not reaching the goal of eliminating all the eye-catching nodes with respect to $\{v\}$.

Example 3. *This is an example to show what happens if we remove w out of the new created cycle when the eccentricity of v is even. In Figure 5.6(a), v_7 is the vertex such that the distance between v_0 and v_7 is equal to the eccentricity of v_0. The distance is 5, which is odd. In order to create a biggest circle with odd number of vertices if we add an edge between v_0 and one immediate predecessor of v_7, we choose v_6 for example in Figure 5.6(b), then v_5 becomes a new eye-catching node with respect to $\{v_0\}$.*

So when the eccentricity of v is odd, it is better to add the edge between one of the immediate descendant of v in the eccentricity paths of v, name it x, and w . No matter whether x or v is chosen as the endpoint of the new edge, both cases satisfy that the new created cycle includes all the eye-catching nodes with respect to $\{v\}$ and has odd number of vertices. So according to Theorem 2, all the local eye-catching nodes with respect to $\{v\}$ are eliminated.



(a) The original graph



(b) After adding an edge

Figure 5.6: An example to show that if w is not the endpoint of the new edge when $\epsilon(v)$ is odd, then there still exists an eye-catching node v_5.

Theorem 3. *Let $G = (V, E)$ be a simple connected graph and v a vertex in $V(G)$. After creating one cycle with respect to $\{v\}$ using EPA, the eccentricity of v decreases.*

Proof. Let w be a vertex such that $d(v, w) = \epsilon(v)$ and z the farthest vertex to v which is not located in any eccentricity paths of v. After adding an edge between v or x and w there are three possibilities for the new eccentricity paths of v:

- The shortest path from v to the middle vertex of the original $v - w$ path where the distance is $\lfloor(\epsilon(v) + 1)/2\rfloor$ such that $\lfloor(\epsilon(v) + 1)/2\rfloor < \epsilon(v)$.
- The shortest path from v to z where the distance is $d(v, z)$ such that $\lfloor(\epsilon(v)+1)/2\rfloor < d(v, z) < d_{n_{eye}}(v) \leq \epsilon(v)$.
- The shortest path from v to w where the distance is $d''(v, w)$ such that $d''(v, w)$ is either 1 or 2.

So the new eccentricity of v is $max\{\lfloor(\epsilon(v)+1)/2\rfloor, d(v, z)\}$ where both of them are smaller than $\epsilon(v)$. □

Intuitively, one single edge in EPA not only eliminates all the eye-catching nodes with respect to a one-vertex subset but also make the graph more connected. According to the definition of edge-connectivity, adding an edge between two vertices improves the size of smallest edge cut. So it is sensible to confirm that EPA tends to increase the edge-connectivity of the graph globally. When the endpoint is v and w it really increases the edge-connectivity of the $v - w$ path while if the endpoint is x and w then it does not. The more connected the graph is, the less possibility eye-catching nodes exist, less edges are needed to be added. That is why we name this approach *Edge-Preserving Approach.*

The following two examples show two circumstances when transforming a graph with EPA. One is the eccentricity of v_0 is odd, i.e., Example 4. The other is the eccentricity of v_0 is even, i.e., Example 5

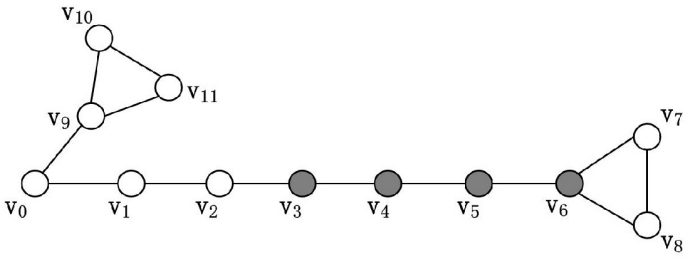
Example 4. *In Figure 5.7(a), v_7 and v_8 are vertices such that $d(v_0, v_7) = d(v_0, v_8) = \epsilon(v_0) = 7$. The distance between v_0 and both of v_7 and v_8 is odd. So the immediate descendant of v_0 in the eccentricity paths of v_0 which is v_1 is one endpoint of the new edge. The other is either v_7 or v_8. If we add an edge between v_1 and v_8, then a cycle with odd number of vertices $v_1 - v_2 - v_3 - v_4 - v_5 - v_6 - v_8 - v_1$ is created. In Figure 5.7(b), there is no eye-catching node with respect to $\{v_0\}$. The eccentricity of v_0 is changed from 7 to 4.*

Example 5. *Figure 5.8 is an example for another circumstance that the eccentricity of v_0 is even. In Figure 5.8(a) there are three eye-catching nodes with respect to $\{v_0\}$ which are v_3, v_4 and v_5. The farthest vertices to v_0 are v_6 and v_7. According to EPA adding an edge between v_0 and v_6 or v_7 eliminates all the local eye-catching nodes with respect to $\{v_0\}$, e.g., Figure 5.8(b).*

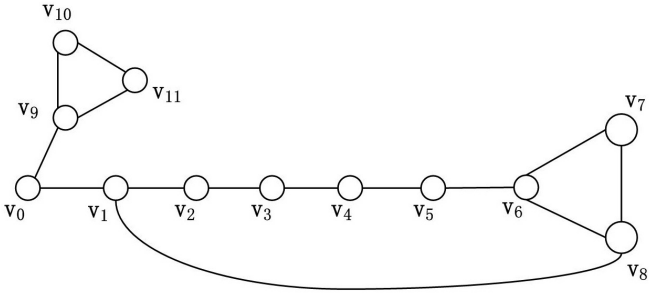
Connectivity-preserving approach (CPA)

Although EPA removes eye-catching nodes with respect to each one-vertex subset of $V(G)$ and tends to make the graph more connected. Adding an edge between two far away vertices is less probable in reality than between two nearer vertices, where the previous operation also compromises the utility of the graph. In order to maintain the structure of the original graph, we try to create a smallest cycle which seems to make the least structural breach to the original graph.

Solution 2. *(CPA) Given a simple connected graph $G = (V, E)$ with an attacker node v. Let x be the nearest predecessor of v_n in the eccentricity paths of v whose distance to v_f is even. Our proposal is to create a cycle by adding an edge between v and x.*



(a) The original graph



(b) Carry out EPA with respect to $\{v_0\}$

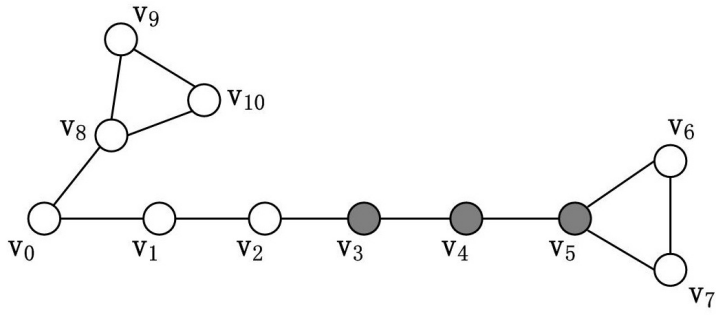
Figure 5.7: Carry out EPA with respect to $\{v_0\}$ when $\epsilon(v)$ is odd.

To create a smallest cycle, one endpoint is easy to be fixed, that is, v_f. To satisfy that the number of vertices in the cycle are odd we need to find the nearest predecessor of v_n in the eccentricity paths of v whose distance to v_f is even. Let x be this vertex. Then the cycle created by adding an edge between x and v_f is the smallest cycle with respect to $\{v\}$ because no more vertex can be removed in the cycle to satisfy the statement in Theorem 2.

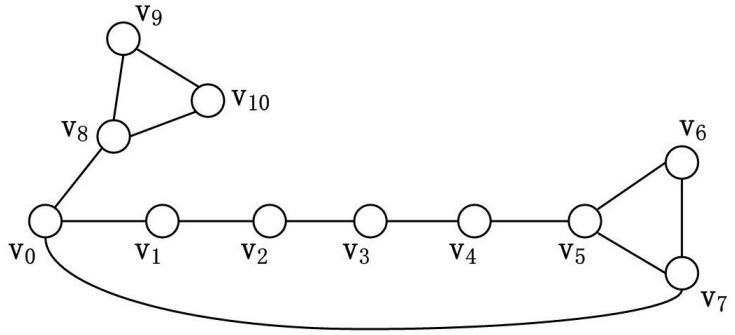
There are two possibilities for x. If the distance between v_n and v_f are odd, then x is the immediate predecessor of v_n in the eccentricity paths of v whose distance to v is $d_{n_{eye}}(v) - 1$. If the distance between v_n and v_f is even, x is a predecessor of v_n in the eccentricity paths of v whose distance to v_n is 2 and distance to v is $d_{n_{eye}}(v) - 2$. The smallest cycle is created by adding an edge between v_f and x.

There are also three possibilities of the new eccentricity paths of v. Let w be one of the vertices such that $d(v, w) = \epsilon(v)$ and z the farthest vertex to v who is not located in any eccentricity paths of v, and the eccentricity after creating a circle with respect to $\{v\}$ are:

- The shortest path from v to one of the vertex in the middle of $x - v_f$ where the distance is $d(v, x) + d(x, v_f)/2$ such that $d(v, x) + d(x, v_f)/2 < d(v, x) + d(x, v_f) < d(v, x) + d(x, v_f) + d(v_f, w) = \epsilon(v)$
- The shortest path from v to z where the distance is $d(v, z)$ such that $d(v, z) < d_{n_{eye}}(v) \leq \epsilon(v)$
- The new shortest from v to w where the distance is $d''(v, w) = d(v, x) + d(x, v_f) + d(v_f, w) = d(v, x) + 1 + d(v_f, w)$ such that $d(v, x) + 1 + d(v_f, w) < d(v, x) + d(x, v_f) + d(v_f, w) = \epsilon(v)$.



(a) The original graph



(b) Carry out EPA with respect to $\{v_0\}$

Figure 5.8: Carry out EPA with respect to $\{v_0\}$ when $\epsilon(v)$ is even.

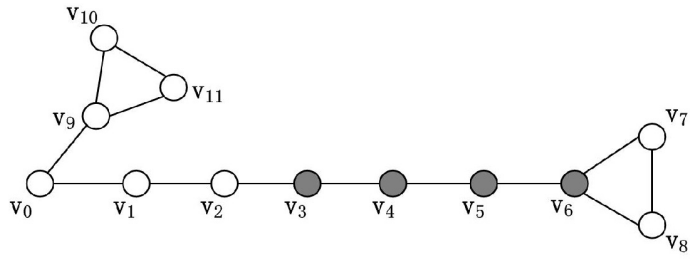
Example 6. *Figure 5.9 shows the circumstance that the distance between v_n and v_f is odd. In the original graph in Figure 5.9(a), v_3 is the nearest eye-catching node with respect to $\{v_0\}$ while v_6 is the farthest. The distance between v_6 and v_3 is odd. So x is the immediate predecessor of v_3 in the eccentricity paths of v_0 which is v_2. After adding an edge between v_2 and v_6, a smallest cycle $v_2 - v_3 - v_4 - v_5 - v_6 - v_2$ is created by CPA, there is no eye-catching node with respect to $\{v_0\}$.*

Example 7. *The graph in Figure 5.10 shows the other circumstance that the distance between v_n and v_f is even. In Figure 5.10(a) v_3 is the nearest eye-catching node with respect to $\{v_0\}$ while v_5 is the farthest and the distance between them is 2. In this case x is the predecessor of v_3 whose distance to v_3 is 2 which is v_1. The new transformed graph by CPA is shown in Figure 5.10(b).*

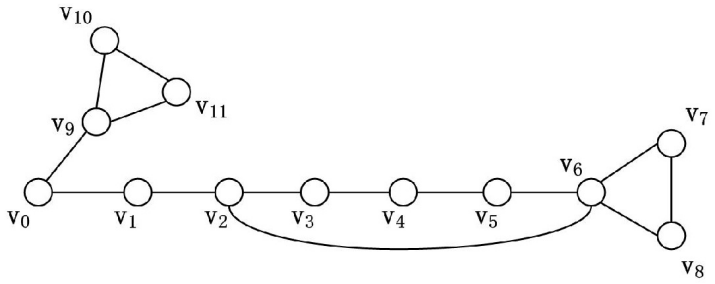
Tradeoffs

Locally, based on Theorem 2 both approaches create a cycle with an odd number of vertices which includes all eye-catching nodes with respect to each one-vertex subset of the graph. Both of them eliminate all the existing eye-catching nodes with respect to the subset $\{v\}$ of the graph with a single edge. Globally, adding an edge to a graph tends to make the graph more connected where fewer eye-catching nodes exist, there exists one ending moment that all the eye-catching nodes are removed.

With respect to a one-vertex subset $\{v\}$ of the graph, EPA creates the biggest cycle in the eccentricity path of v. It sharply shortens the eccentricity of v which makes the



(a) The original graph



(b) Carry out CPA with respect to $\{v_0\}$

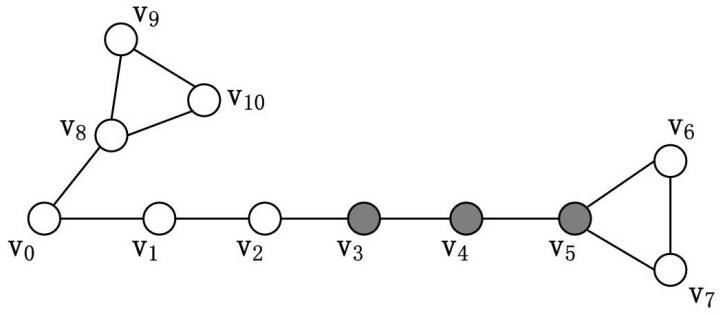
Figure 5.9: Carry out CPA with respect to $\{v_0\}$ when $d(v_n, v_f)$ is odd.

Algorithm 3 Given a graph $G = (V, E)$ this algorithm outputs a graph $G' = (V, E')$ which satisfies $(k,1)$-anonymity for $k > 1$

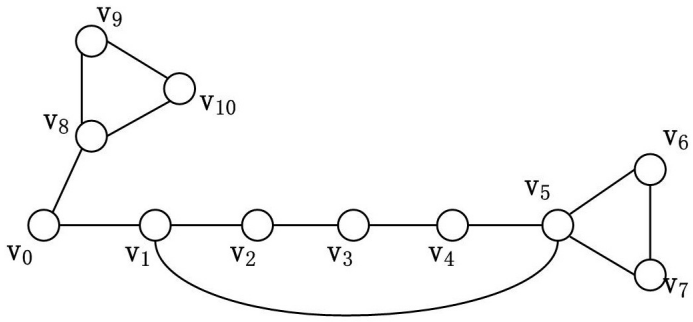
Let $E'(G') = E(G)$.
for $\forall v \in V(G)$ **do**
  BFS(v)
  Let x be the nearest predecessor of v_n in the eccentricity paths of v whose distance to v_f is even.
  Add an edge e between x and v_f,
  $E'(G')$=$E'(G') \cup \{e\}$
end for
Output G'.

graph more connected. As is known from Theorem 1, it does not mean only graphs with high-connectivity have no eye-catching nodes. A graph with a bridge can still satisfy $(k, 1)$-anonymity for $k > 1$. The graph in Figure 5.11 is such an example. For every vertex in the graph there exists at least another one vertex that has the same metric representation with respect to a subset $\{v\}$ of the graph, so the graph satisfies (2,1)-anonymity. But as there is a bridge $v_4 - v_5$, the edge-connectivity of the graph is 1.

There is no direct relationship between the edge-connectivity of the graph and eye-catching nodes with respect to a subset of the graph. But it is more possible that the unique distance to v appears when the graph is less connected. For example, compared with a complete graph K_n which is maximumly connected, a path P_n with the same order has more eye-catching nodes with respect to any one-vertex subset of the graph. A more connected graph has less possibility of containing eye-catching nodes with respect to every one-vertex subset of the graph.



(a) The original graph



(b) Carry out CPA with respect to $\{v_0\}$

Figure 5.10: Carry out CPA with respect to $\{v_0\}$ when $d(v_n, v_f)$ is even.

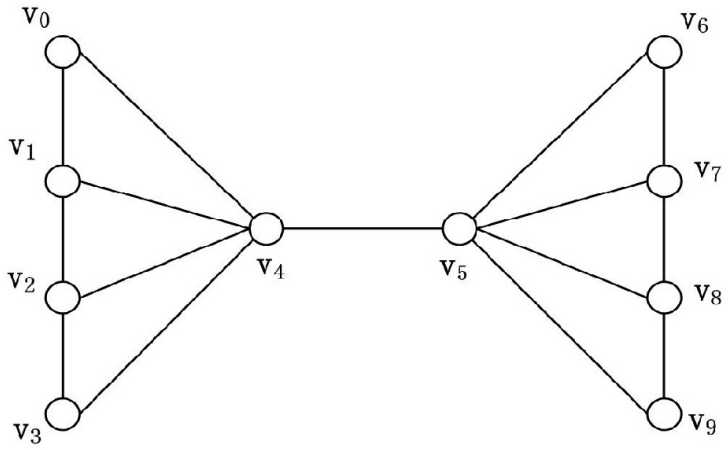


Figure 5.11: A graph satisfies (2,1)-anonymity whose edge-connectivity is 1.

While CPA removes all the eye-catching nodes with respect to every one-vertex subset $\{v\}$ of the graph by means of creating a smallest circle with odd number of vertices. The advantage is obvious that the smallest circle affects the graph partially in terms of the graph connectivity, which preserves the original structural property of the graph. Because the connectivity out of the cycle is not changed, the smaller the cycle is the fewer changes the structure of the graph is made.

The final aim for the proposal approaches is for the sake of publishing the anonymized graph instead of the original graph which preserves both the utility of the graph and the privacy of the individuals for deep study. EPA achieves the goal faster with big information loss while CPA preserves the utility of the graph to a larger extent, however, with lower speed.

Chapter 6

Experiment

In this section we report a systematic empirical study to evaluate our anonymization method using both synthetic data sets and real data sets. All the experiments were performed on the UL HPC platform [45].

We first report the anonymization quality on both data sets, show the anonymization cost in terms of the number of added edges and the information loss evaluated by a metric named *connectivity loss*.

It is challenging to evaluate the information loss in anonymizing social network data. As the structural relationship should be considered, we cannot compare two social networks by simply comparing the vertices and edges individually. Even though two graphs have the same number of edges and nodes, they may be quite different structurally. Therefore, in this thesis, we use a network-wise property, connectivity loss, to measure the information loss after the anonymization.

Definition 7. *(Connectivity Loss) Given an original graph G and its anonymized version G', the connectivity loss in G' is defined as*

$$ConLoss(G, G') = \frac{Con(G') - Con(G)}{Con(G)}$$

where $Con(G)$ and $Con(G')$ are the connectivity of G and G', respectively.

The rationale of using connectivity loss to measure the information loss in G' is that a lower connectivity loss indicates that fewer structural changes have been made to the original graph G. Furthermore, some other statistical network measures such as degree difference, average shortest-paths and cluster coefficient, are also used to evaluate the utility of the released network[58][18][28].

Secondly, we consider what happens if an adversary with weaker background knowledge plants more than one attacker node in the original graph. We used the walk-based active attack proposed in [3] and show that the success rate of attacking decreases after anonymizing graphs with our transformation methods.

Recalling the walk-based attack. An attacker first chooses an arbitrary set $W = \{w_1, w_2, ...w_b\}$ of users in G as targeted individuals. Secondly, without knowing what G looks like, the attacker creates a new graph $H = (V, E)$ where $V(H) = \{x_1, x_2, ...x_k\}$ and k is greatly smaller than the size n of G. Each x_i for $i \in \{1, 2, ...k\}$ has an external degree $\triangle_i$ indicating the number of edges x_i will have to nodes in $G - H$. For each vertex w_j for $j \in \{1, 2, ...b\}$ it has a distinct set $N_j \subseteq \{x_1, ...x_k\}$ containing all the nodes in H connected to w_j, which will be used to identify each w_j once G is released. The attacker generates the random internal edges in H by including the edge $(x_i, x_i + 1)$ for

$i \in \{1, ...k-1\}$ and the edge (x_i, x_j) with probability 1/2. In this way, a subgraph H of G is created which is in general identified efficiently and uniquely. Once G is released, the adversary performs a search algorithm to identify H and then re-identifies w_j with the knowledge of N_j for $j \in \{1, 2, ...b\}$. As a consequence, with $k = \Theta(\log n)$ new accounts the attacker can reveal the identities of $\Theta((\log n)^2)$ targeted nodes.

In [3], Backstrom et al. recommend to create $k = (2 + \delta) \log n$ new accounts for a small constant $\delta > 0$. In our thesis, we test the success rate of attacking random graphs when $\delta \in \{0.1, 0.5, 1, 1.5\}$ and the average success rate of attacking one social network graph 100 times when δ is 0.1 which is the smallest among $\{0.1, 0.5, 1, 1.5\}$. We choose $(\log n)^2$ random vertices as the targeted vertices. Each $\triangle_i$ is chosen independently and uniformly at random from the interval $[1, \log n]$. The size of each N_j is limited at most 3. We plant H in G as mentioned in the walk-based attack and compare the new success rate of attacking both random graphs and a real-life social network graph which are anonymized by CPA and EPA with the success rate of attacking their original forms, respectively.

6.1 Empirical evaluation on random graphs

In the following two subsections, we first show the quality of our anonymization methods which is measured by $(k, 1)$-anonymity, the cost of our anonymization on random graphs which is measured by the number of added edges and the information loss evaluated by connectivity loss. In the second subsection, we show how the anonymized random graphs perform facing an active attack when the number of the attacker nodes is bigger than one. Normally, as our approach changes the metric representation of nodes with respect to attacker nodes when anonymizing graphs, it tends to make it harder for the adversary to identify the targeted vertices. If there are more than one vertex connecting the same subset of attacker nodes, then the adversary cannot distinguish them and this leads to the decline of the success rate.

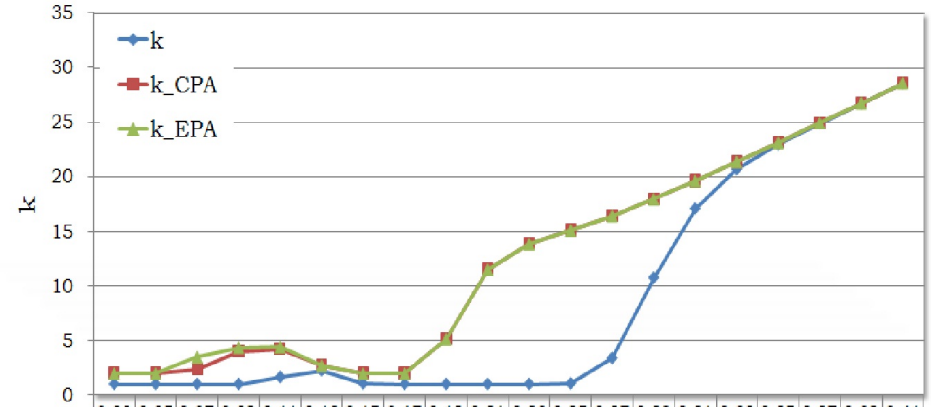
6.1.1 Anonymization quality and cost

We measure the performance of both approaches by means of the value of k for $(k, 1)$-anonymity which corresponds to the anonymization quality and the number of added edges corresponding to the anonymization cost. $(k, 1)$-anonymity is a privacy metric where any vertex in a $(k, 1)$-anonymous graph cannot be distinguished with a probability higher than $1/k$. The larger the value of k is, the harder the vertices can be distinguished. As an edge addition operation changes the structural properties of the graph such as the vertex degree and the eccentricity of the vertex, the fewer edges are added by the approach, the fewer changes are done to the graphs, the better performance the approach owns.

To check the performance of CPA and EPA we ran experiments on random graphs under different density. We fix 100 as the number of vertices in each random graph. The number of edges distribute uniformly in the interval $[99, \frac{100*99}{2}]$ and the edges are added randomly to the graph. Without loss of generality, for each density of the graph we created 100,000 random graphs and transformed them with EPA and CPA, respectively, examine how much extent of the privacy they can preserve based on $(k, 1)$-anonymity and how many edges are needed for the transformation.

With respect to the anonymization quality, before transforming the graphs we examine the original value of k based on $(k, 1)$-anonymity for the graphs. Transform the graphs with the two approaches and record their new value of k. The curve graph in Figure

6.1 manifests that the privacy preserving level in terms of $(k, 1)$-anonymity is improved by both approaches. The horizontal axis indicates the density of the graph and the vertical axis represents the average value of k for the graphs. We can see from the chart that when the random graph is sparse, i.e. the graph density is smaller than 0.17, both methods can increase the value of k slightly. However, when the graph density is between 0.17 and 0.27, even if the original value of k is low, i.e., 1, it is significantly increased by both methods. As the graph becomes denser, the graph density is larger than 0.27, the majority of graphs have already satisfied $(k, 1)$-anonymity for high value of k, the growth of k becomes slower and finally reaches to 0 when the graph density is 0.35.



	0.03	0.05	0.07	0.09	0.11	0.13	0.15	0.17	0.19	0.21	0.23	0.25	0.27	0.29	0.31	0.33	0.35	0.37	0.39	0.41
k	1	1	1	1	2	2	1	1	1	1	1	1	3	11	17	21	23	25	27	29
k_CPA	2	2	2	4	4	3	2	2	5	12	14	15	16	18	20	21	23	25	27	29
k_EPA	2	2	4	4	4	3	2	2	5	12	14	15	16	18	20	21	23	25	27	29

Figure 6.1: The curve shows how the value of k in $(k, 1)$-anonymity changes after the graph is transformed by CPA and EPA. The table below dedicates the actual value corresponding to the node in the curve.

In regard to the anonymization cost, we record the number of added edges when anonymizing graphs with different densities. Figure 6.2 describes the number of added edges by both methods when the graph density is from 0.01 to 0.43. When the graph is very sparse, i.e., 0.07, an enormous difference happens to the added edge numbers between CPA and EPA where less edges is enough for EPA to anonymize the graphs while CPA needs more than 50 edges for a graph with 4950 edges in total. When the graph becomes denser, the graph density is higher than 0.09, both methods perform similar.

From both curves in Figure 6.1 and 6.2, when the graph density is 0.13 the original random graphs satisfy (2,1)-anonymity which leads to the average added number is 0. When the graph density is 0.19 the original random graphs satisfy (1,1)-anonymity again, more than 40 average edges are added to the graph. As the standard deviation for both methods are also high when the graph density is 0.19 and 0.07, it is reasonable to believe that if we do experiments on more than 100,000 random graphs the result will be better. When the graph density is bigger than 0.33, random graphs have already satisfied $(k, 1)$-anonymity for $k > 1$ so no edges are needed.

The experimental result corresponds to our analysis of tradeoffs between both methods. EPA removes eye-catching nodes with respect to attacker nodes faster and less edges are

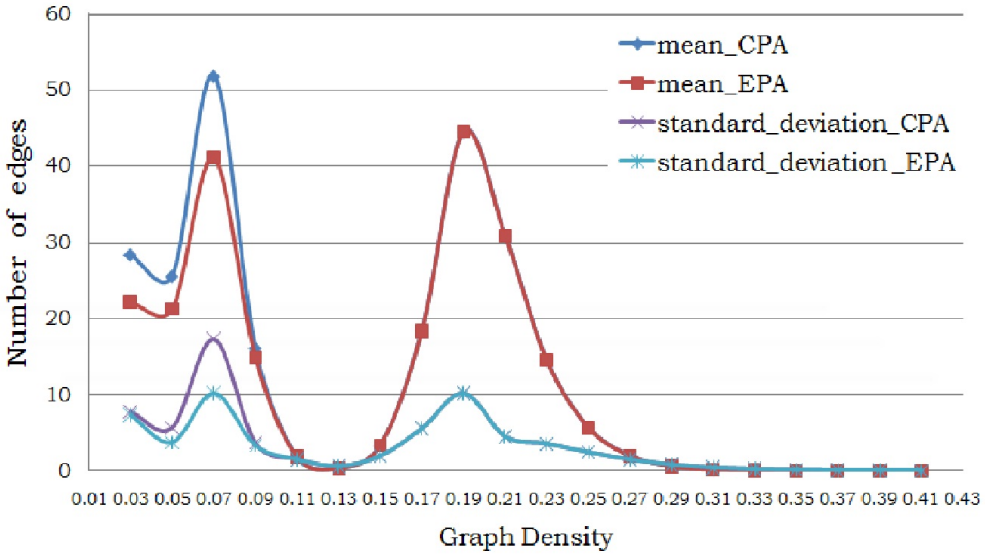


Figure 6.2: The mean and standard deviation of the added edge numbers for both EPA and CPA when the graph density differs.

needed when transforming a random graph to privacy-preserving graph which is reflected obviously when the graph is relatively sparse in the Figure 6.2.

The connectivity loss after transforming the original graphs with different density to privacy-preserving graphs using both EPA and CPA is shown in Figure 6.3. From the chart, we can see distinctly that the connectivity loss of CPA is relatively lower than that of EPA when the graph is sparse, i.e., the graph density is 0.07. As the graph becomes denser, the difference of the connectivity loss for both methods reduces slightly to 0.

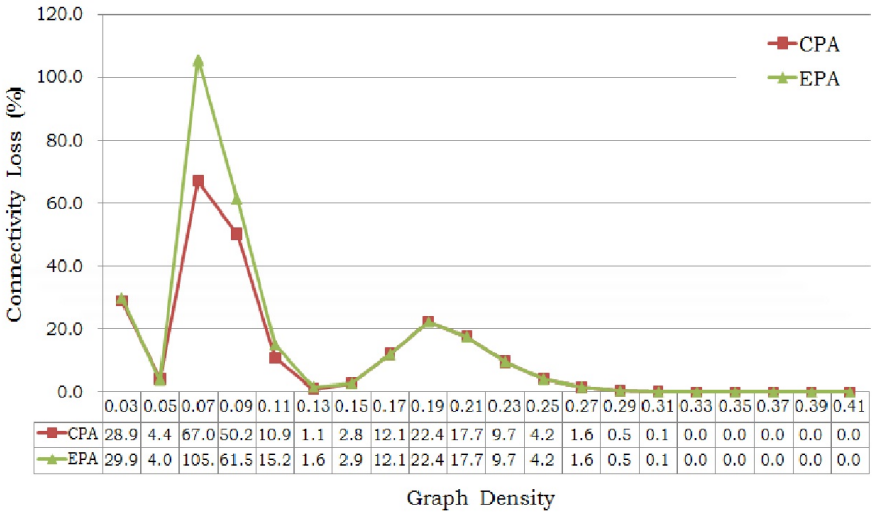
The experimental result for two connectivity loss variation validates our theoretical analysis that CPA keeps the original graph connectivity to a relatively larger extent than EPA when transforming random graphs to privacy-preserving graphs.

6.1.2 Evaluation of the anonymization against the walk-based active attack

We choose four numbers which are 4, 5, 6, 7 as the attacker nodes $m = (2 + \delta) \log n$ where $n = 100$ by changing the value of $\delta \in \{0.1, 0.5, 1, 1.5\}$, respectively.

We attack 100,000 random graphs and record the success rates for each value of m. We define success when the identified vertex corresponds to the targeted vertex. If all the identified vertices correspond to the targeted vertices, the success rate of the attack is 1, otherwise it is a ratio between 0 and 1. The closer the ratio is to 1, the better performance of attacking, the worse the ability of preserving the privacy the anonymization method owns. In order to validate whether our proposed approaches preserve the privacy of the individuals, we transform these random graphs with EPA and CPA respectively and attack them with different number of attacker nodes. Compare the success rate of attacking transformed graphs by both methods with the success rate of attacking the original graphs.

Figure 6.4 illustrates the success rate before and after transformation by both ap-



	0.03	0.05	0.07	0.09	0.11	0.13	0.15	0.17	0.19	0.21	0.23	0.25	0.27	0.29	0.31	0.33	0.35	0.37	0.39	0.41
CPA	28.9	4.4	67.0	50.2	10.9	1.1	2.8	12.1	22.4	17.7	9.7	4.2	1.6	0.5	0.1	0.0	0.0	0.0	0.0	0.0
EPA	29.9	4.0	105.	61.5	15.2	1.6	2.9	12.1	22.4	17.7	9.7	4.2	1.6	0.5	0.1	0.0	0.0	0.0	0.0	0.0

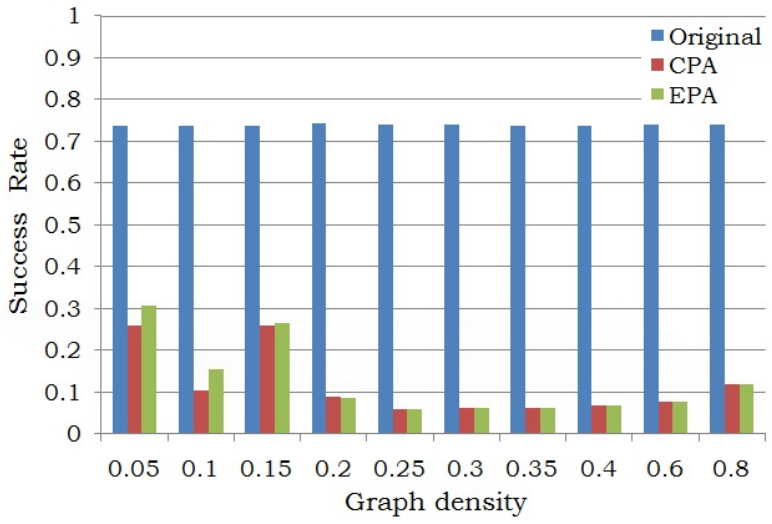
Figure 6.3: The connectivity loss of both EPA and CPA when the graph density varies.

proaches when the graph density differs for $m \in \{4, 5, 6, 7\}$. As is shown in 6.4(a), both EPA and CPA can decrease the success rate sharply to a low value comparing to that of attacking the original graphs. When the graph is sparse, i.e., the graph density is smaller than 0.1, CPA decreases the success rate to a lower value than EPA. When the graph density is larger than 0.1, the success rate of attacking the graphs transformed by either EPA or CPA is similarly low. Other three charts in Figure 6.4(b), 6.4(c), 6.4(d) for $m \in \{5, 6, 7\}$ have the same circumstance with the first chart where m=4.

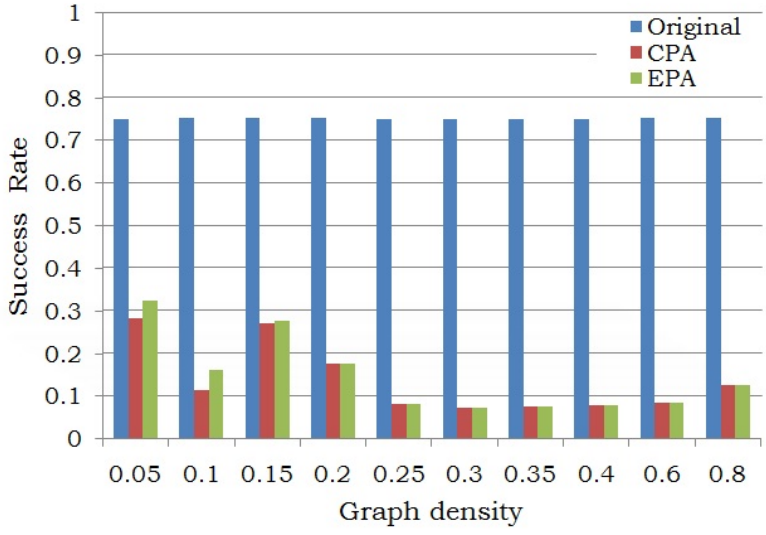
Using the same data Figure 6.5 shows the success rate before and after the graph is transformed by EPA and CPA, respectively, which is easier to evaluate them individually. From both charts, we can conclude that the walk-based attack usually succeeds with a success rate higher than 70% even the number of attacker nodes is small and the more nodes the higher the success rate is. Furthermore, both anonymization methods decrease the success rate among which the lowest decreased success rate is lower than 0.1 when the number of attacker nodes is 4. When the attacker plants fewer vertices, i.e., 4, in the original random graphs, the success rate of attacking is easier to be decreased by both methods. On the contrary, when the attacker plants a subgraph with 7 nodes in the random graphs, the success rate of attacking is also decreased but relatively to a less extent.

6.2 Empirical evaluation on real-life social graphs

In this subsection, we show the quality of our anonymization methods and the cost of the transformation on a real-life social network graph. We validate the resistance against active attack of our anonymization methods which can improve the privacy protection while preserving the utility by means of comparing the success rate of attacking the social network graph before and after it is transformed by both CPA and EPA, respectively.

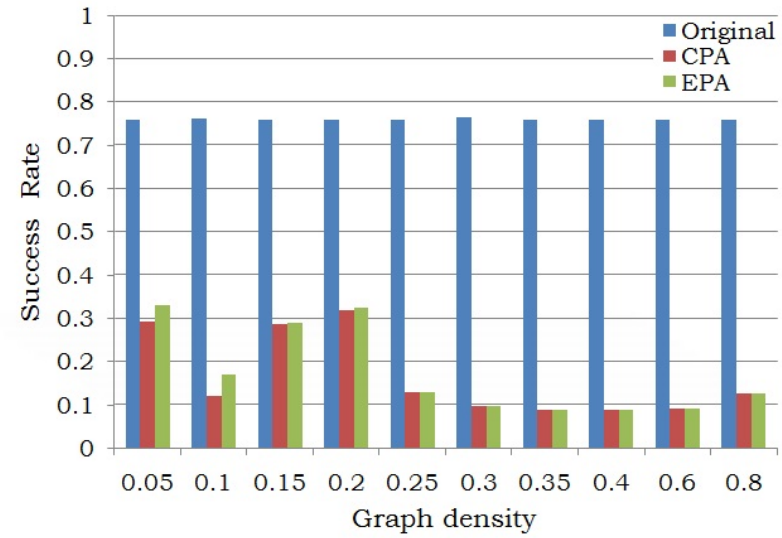


(a) m=4

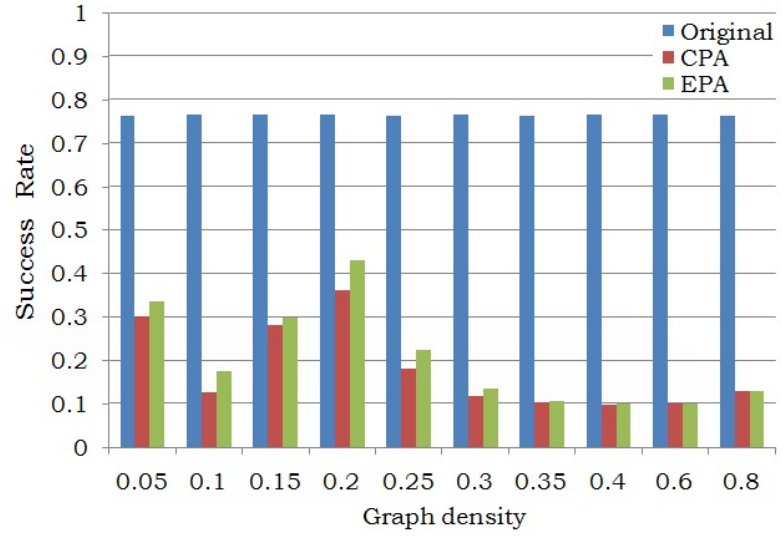


(b) m=5

Figure 6.4: The adversary's success rate for attacker nodes $m \in \{4, 5, 6, 7\}$.

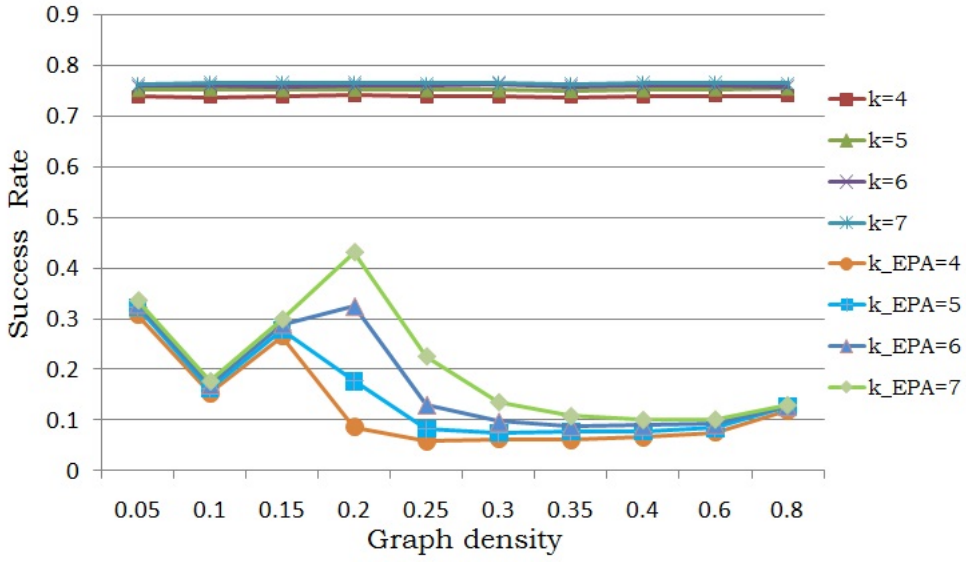


(c) m=6

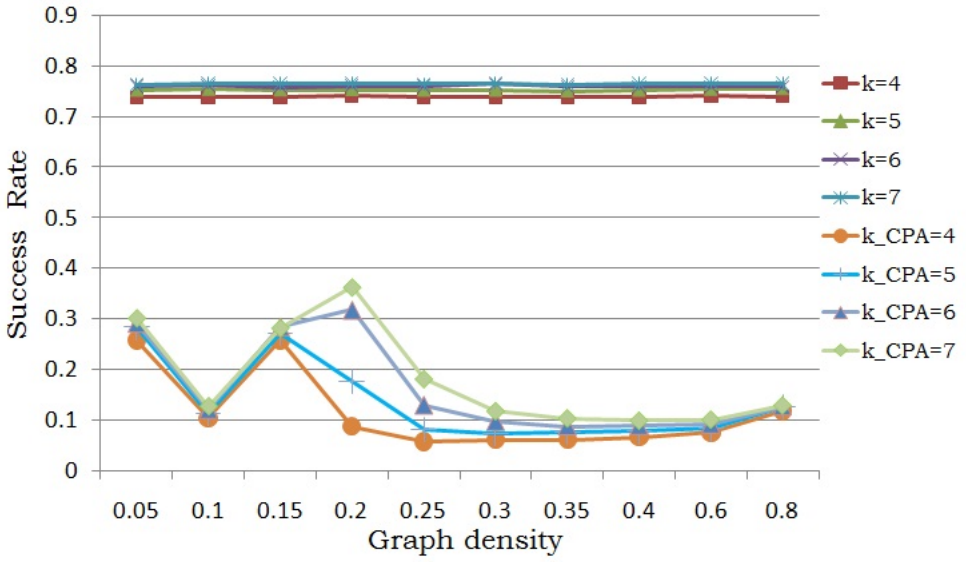


(d) m=7

Figure 6.4: The adversary's success rate for attacker nodes $m \in \{4, 5, 6, 7\}$ (cont.)



(a) EPA



(b) CPA

Figure 6.5: The attacker's success rate before and after the graph is transformed by EPA and CPA, respectively

6.2.1 Anonymization quality and cost

The social network data originated from an online community for students at University of California which is referred as *Panzarasa graph* in [44] and satisfies $(1, 1)$-anonymity only. It recorded 59,835 messages sent between 1899 students from 23th, March, 2004 to 26th, Oct, 2004. As it is a directed and weighted graph, we converted each directed edge to an undirected edge and deleted all the weights of the edges and self-loops. It happens that one student sends to another more than one message at different time, so we only kept one record if the sender and the receiver are the same in the data sets. Furthermore, we deleted six isolated nodes, then we got a simple undirected graph without weights and multiple loops, where the number of vertices and edges are 1893 and 20296, respectively.

The density of the input graph is:

$$density = \frac{2 * 20296}{1893 * (1893 - 1)} = 0.01133362.$$

The statistical property information about this social network graph before and after normalization is shown in Table 6.1

	Original graph	After normalization
Number of vertices	1899	1893
Number of edges	59,835	20,296
density	0.03320199	0.01133362

Table 6.1: The property information about *Panzarasa graph*

We deal with end-vertices in the graph first as mentioned in Section 5.4.1 and transform the graph 100 times with CPA and EPA respectively considered that the transformation starts from the different vertex every time. The results is shown in Table 6.2. As shown in the Table 6.2, both methods improve the privacy protection where the

	Original graph	Transformed by CPA	Transformed by EPA
$(k, 1)$-anonymity	(1,1)-anonymity	(2,1)-anonymity	(2,1)-anonymity
Connectivity	1	2	2
Average added edges	0	34.65	34.53

Table 6.2: The anonymization quality and cost of transforming the social network graph with both methods.

social network graph satisfies (2,1)-anonymity after transformed by both methods while meets only (1,1)-anonymity previously. Both of them increase the connectivity of the graph from 1 to 2. The transformation cost of CPA and EPA are 34.65 and 34.53, in terms of average added edges, respectively, which is not very big compared to the number of edges in the graph. As the density of this social network graph is close to 0.01, the result also validates that in order to anonymize sparse graphs, i.e., the graph density is 0.01, EPA adds less edges than CPA.

6.2.2 Evaluation of the anonymization against the walk-based active attack

As is known from [3], to achieve the high success rate only a small number of attacker nodes are needed although the attacker only knows the connection with its neighbors and the connection between itself.

We create $(2 + 0.1) \log 1893 \approx 7$ vertices for the adversary graph H where $\delta = 0.1$ according to $(2 + \delta) \log n$ and select $(\log 1893)^2 \approx 11$ targeted vertices randomly from the social network graph. Construct the subgraph H and plant it into *Panzarasa graph*, G_p for short. We call the new graph $G_p + H$. After $G_p + H$ is released, we retrieve H and identify targeted nodes with the help of N_j for the jth targeted node for $j \in \{1, ..., 11\}$. If for each N_j there is only one vertex in the published graph G_p connecting the vertices in N_j, then we treat it successful otherwise failure.

To validate the performance of our proposed approaches, we still create an attacker graph H with 7 attacker vertices and plant it into G_p. We transform $G_p + H$ to

$(G_p + H)_{CPA}$ with CPA and transform $G_p + H$ to $(G_p + H)_{EPA}$ with EPA, respectively. After the anonymized graph is released, recover H and identify targeted vertices according to N_j. The results are shown in Table 6.3.

	$G_p + H$	$(G_p + H)_{CPA}$	$(G_p + H)_{EPA}$
Attacker nodes	7	7	7
Average Success rate	64.83%	43.01%	43.59%
Standard Deviation of Success rate	0.0647	0.0982	0.0938

Table 6.3: The average success rate and the standard deviation of the success rate, before and after the social network graph is transformed by EPA and CPA, respectively.

After the social network graph is transformed by CPA the metric representation of nodes with respect to the set of 7 attacker nodes is different from that previously because of nearly 35 added edges to the graph, the same as nearly 35 edges for EPA. Owing to this changes it is acceptable that the vertex connecting to the nodes in N_j is not unique, which prevents the adversary from identifying the targeted nodes and the success rate goes down. As is shown in Table 6.3, both methods decreases the success rate from 64.83% to 43.01% and 43.59%, respectively. It can be inferred that graphs satisfying $(k, 1)$-anonymity for $k > 1$ can also resist against active attacks where the number of attacker nodes is bigger than 1.

Chapter 7

Conclusions and future work

In this thesis, we have addressed issues related to preserving both privacy and utility of published social networks.

We performed a thorough study of the state of the art, with emphasis on anonymization methods, privacy metrics, passive attacks and active attacks. We studied the theoretical properties of $(k, 1)$-anonymous graphs, in particularly (1,1)-anonymous graphs. Considering the metric representation as the background knowledge, we focused on the anonymization methods against the walk-based attack. We presented two privacy-preserving methods for the publication of social networks considering $(k, 1)$-anonymity as a privacy metric.

According to different emphasis, the Connectivity-Preserving Approach is able to transform graphs to $(k, 1)$-anonymous graphs for $k > 1$ by creating smallest cycle with respect to each one-vertex subset of the graph which arouses less structural breach than Edge-Preserving Approach. While, Edge-Preserving Approach can also anonymize graphs to satisfy $(k, 1)$-anonymity for $k > 1$ by creating the biggest cycle in the eccentricity path of each vertex with fewer edges to be added than Connectivity-Preserving Approach, however, it leads to more information loss.

Furthermore, we demonstrated their resistance against the walk-based attack where there are more than one attacker node in the graph. The results of the decreasing success rate validated that both approaches can preserve individuals' privacy against the active attack.

7.1 Future work

Several lines are still open, and have not been addressed yet, specially due to lack of time. Then, we show those research directions that we believe are interesting:

- We only handle (k, ℓ)-anonymity where $\ell = 1$ in this thesis. It could be desirable and interesting to propose privacy-preserving methods for the publication of social networks regarding (k, ℓ)-anonymity as a privacy metric for $\ell > 1$.

- We study (k, ℓ)-anonymity individually, but there may be relationship between (k, ℓ)-anonymity and other proposed privacy-preserving measures. For example, a graph satisfies k-isomorphism may also satisfies (k, ℓ)-anonymity.

- Transforming a graph to satisfy (k, ℓ)-anonymity induces a large number of edge additions which breaks the privacy of the original graphs. Relaxing the (k, ℓ)-anonymity concept in order to capture the notion that the adversary might not be able to learn the distance to every vertex in the graph, i.e., the adversaries only

knows the metric representation of vertices whose distance to them is smaller than 5, is another desirable direction.

Bibliography

[1] Gagan Aggarwal, Tomás Feder, Krishnaram Kenthapadi, Rajeev Motwani, Rina Panigrahy, Dilys Thomas, and An Zhu. Anonmizing tables. *Proceedings of the 10th international conference on Database Theory*, pages 246–258, May 2005.

[2] A. Arrison. Is Friendster the new TIA? *TechCentralStation*, January 2004.

[3] Lars Backstrom, Cynthia Dwork, and Jon Kleinberg. Where Art Thou R3579X? Anonymized Social Networks, Hidden Patterns, and Structural Steganography. *The 16th International Conference on World Wide Web*, pages 181–190, May 2007.

[4] Albert-László. Barabási* and Réka Albert. Emergence of scaling in random networks. *Science*, 286:509–512, October 1999.

[5] R.J. Bayardo and R. Agrawal. Data privacy through optimal k-anonymization. *ICDE*, pages 217–228, April 2005.

[6] Claudio Bettini, X. Sean Wang, and Sushil Jajodia. The Role of Quasi-identifiers in k-Anonymity Revisited. *arXiv:cs*, Novermber 2006.

[7] J. Black. The perils and promise of online schmoozing. *Business Week Online*, February 2004.

[8] Jeremy Boissevain. Friends of friends:Networks, manipulators, and coalitions. *American Journal of Sociology*, 1974.

[9] A. Campan and T.M. Truta. A clustering approach for data and structural anonymity in social networks. *Proceedings of the 2nd ACM SIGKDD International Workshop on Privacy,Security, and Trust in KDD*, 2008.

[10] Bee-Chung Chen, Kristen LeFevre, and Raghu Ramakrishnan. Privacy Skyline: Privacy with Multidimensional Adversarial Knowledge. *VLDB*, September 2007.

[11] James Cheng, Ada Wai-Chee Fu, and Jia Liu. K-Isomorphism: Privacy Preserving Network Publication against Structural Attacks. *SIGMOD*, June 2010.

[12] Thomas F. Coleman and Jorge J. Moré. Collective dynamics of 'small-world' networks. *SIAM Journal on Numerical Analysis*, 20(1):187–209, 1983.

[13] C. Dwork. Differential privacy. *ICALP*, 2006.

[14] Tomás Feder, Shubha U.Nabar*, and Evimaris Terzi. Anonymizing Graphs. *arXiv*, page 0810.5578, 2008.

[15] B.C.M. Fung, ke Wang, and P.S. Yu. Top-down specialization for information and privacy preservation. *ICDE*, pages 205–216, April 2005.

[16] M. Gladwell. *The Tipping Point: How Little Things Can Make a Big Difference.* Back Bay Books, 2012.

[17] M. Granovetter. The strength of weak ties. *American Journal of Sociology*, 78:1360–1380, 1973.

[18] Michael Hay, Gerome Miklau, David Jensen, Don Towsley, and Philipp Weis. Resisting Structural Reidentification Anonymized Social Networks. *The VLDB journal*, 1(1):102–114, August 2008.

[19] Michael Hay, Gerome Miklau, David Jensen, Philipp Weis, and Siddharth Srivastava. Anonymizing Social Networks. *Computer Science Department Faculty Publication Series*, 2007.

[20] Vijay S. Iyengar. Transforming data to satisfy privacy constraints. *SIGKDD*, pages 279–288, 2002.

[21] Aleksandra Korolova, Rajeev Motwani, Shubha U. Nabar, and Ying Xu. Link privacy in social networks. *CIKM*, pages 246–258, October 2008.

[22] V.E. Krens. Uncloaking terrorist networks. *First Monday*, 7(4), 2002.

[23] Kristen LeFevre, David J. DeWitt, and Raghu Ramakrishnan. Incognito:efficient full-domain k-anonymity. *SIGMOD*, pages 49–60, 2005.

[24] Kristen LeFevre, David J. DeWitt, and Raghu Ramakrishnan. Mondrian multidimensional k-anonymity. *ICDE*, page 25, April 2006.

[25] A. Leonard. You are who you know. *Saalon.com*, June 2004.

[26] Kevin Lewis, Jason Kaufman, Marco Gonzalez, Andreas Wimmer, and Nicholas Christakis. Tastes, ties, and time:A new social network dataset using Facebook.com. *Social Networks*, 30:330–342, October 2008.

[27] Ninghui Li, Tiancheng Li, and Suresh Venkatasubramanian. t-Closeness:Privacy Beyond k-Anonymity and ℓ-Diversity. *IEEE 23rd International Conference on Data Engineering*, pages 106–115, April 2007.

[28] Kun Liu and Evimaria Terzi. Towards Identity Anonymization on Graphs. *SIGMOD*, June 2008.

[29] Ashwin Machanavajjhala, Johannes Gehrke, and Daniel Kifer. ℓ-Diversity:Privacy Beyond k-Anonymity. *Proceedings of the 22nd Internation Conference on Data Engineering*, page 24, 2006.

[30] S. Milgram. The small world problem. *Psychology Today*, 6:62–67, 1967.

[31] S. Milgram. The familiar stranger: An aspect of urban anonymity. *The Individual in a Social World: Essays and Experiments*, 1977.

[32] Rajeev Motwani and Ying Xu. Efficient Algorithms for Masking and Finding Quasi-Identifiers. *VLDB*, September 2007.

[33] Arvind Narayanan and Vitaly Shmatikov. De-anonymizing Social Networks. *IEEE*, March 2009.

[34] A. Newitz. Defenses lacking at social network sites. *SecurityFocus*, December 2003.

[35] Diestel Reinhard. *Graph Theory*. Springer-Verlag Berlin Heidelberg, 2005.

[36] Pierangela Samarati and Latanya Sweeney. Protecting Privacy when Disclosing Information :k-anonymity and Its Enforcement through Generalization and Suppression. *IEEE Security and Privacy*, 1998.

[37] I. Sege. Where everybody knows your name. *Boston.com*, April 2005.

[38] Sophos. Sophos Facebook ID probe shows 41% of users happy to reveal all to protential identity thieves . `https://www.sophos.com/en-us/press-office/press-releases/2007/08/facebook.aspx`. August 14,2007.

[39] M.K. Sparrow. The application of network analysis to criminal intelligence: an assessment of the prospects. *Social networks*, 13:251–274, 2002.

[40] Jaideep Srivastava, Muhammad A. Ahmad, Nishith Pathak, and David Kuo-Wei Hsu. Data Mining Based Social Network Analysis from Online Behavior. *Tutorial at the 8th SIAM International Conference on Data Mining*, 2008.

[41] Z. Stone, T. Zickler, and T. Darrell. Autotagging Facebook: Social network context improves photo annotation. *Computer Vision and Pattern Recognition Workshops*, pages 1–8, June 2008.

[42] Latanya Sweeney. k-anonymity: a model for protecting privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10(5):557–570, May 2002.

[43] Latanya Sweeney. Uniqueness of Simple Demographics in the U.S. Population. *International Journal on Uncertainty, Fuzziness and Knowledge-based Systems*, 10(5):557–570, 2002.

[44] Rolando Trujillo-Rasua and Ismael G. Yero. k-Metric Antidimension: a Privacy Measure for Social Graphs. *arXiv*, December 2014.

[45] S. Varrette, H. Bouvry, P. andCartiaux, and F. Georgatos. Management of an academic HPC cluster:The UL experience. *International Conference on High Performance Computing & Simulation(HPCS)*, pages 959–967, July 2014.

[46] Da-Wei Wang, Churn-Jung Liau, and Tsan-sheng Hsu. Privacy protection in social network data disclosure based on granular computing. *IEEE International Conference on Fuzzy Systems*, pages 997–1003, 2006.

[47] D. Watts. *Six Degrees:The Science of a Connected Age*. W.W.Norton & Company, 2003.

[48] Duncan J. Watts. *Small Worlds:The Dynamics of Networks Between Order and Randomness*. Princeton University Press, 2003.

[49] Peng Wei, Feng Li, Xukai Zou, and Jie Wu. A Two-Stage Deanonymization Attack against Anonymized Social Networks. *IEEE Transactions on Computers*, 63(2):290–303, 2014.

[50] Qiong Wei and Yansheng Lu. Preservation of Privacy in Publishing Social Network Data. *International Symposium on Electronic Commerce and Security*, 2008.

[51] Eric W. Weisstein. Graph Geodesic. `http://mathworld.wolfram.com/GraphGeodesic.html`. [From $MathWorld - A Wolfram Web Resource$].

[52] Douglas B. West. *Introduction to Graph Theory.* Prentice Hall, 2 edition, 2000.

[53] Xiaokui Xiao and Yufei Tao. Anatomy: simple and effective privacy preservation. *Proceedings of the 32nd international conference on Very large data bases*, pages 139–150, 2006.

[54] Xiaokui Xiao and Yufei Tao. M-invariance: towards privacy preserving re-publication of dynamic datasets. *Proceedins of the 2007 ACM SIGMOD international conference on Management of data*, pages 689–700, 2007.

[55] X. Ying and X. Wu. Randomizing social networks: a spectrum preserving approach. *Proceedings of the 8th SIAM Conference on Data Mining*, 2008.

[56] Qing Zhang, N. Koudas, D. Srivastava, and Ting Yu. Aggregate Query Answering on Anonymized Tables. *IEEE 23rd International Conference on Data Engineering*, pages 116–125, April 2007.

[57] E. Zheleva and L. Getoor. K-Isomorphism: Privacy Preserving Network Publication against Structural Attacks. *Proceedings of the 1st ACM SIGKDD International Workshop on Privacy, Security, and Trust in KDD*, pages 153–171, 2007.

[58] Bin Zhou and jian Pei. Preserving Privacy in Social Networks Against Neighborhood Attacks. *IEEE 24th International Conference on Data Engineering*, pages 506–515, 2008.

[59] Lei Zou, Lei Chen, and M.Tamer Özsu. K-Automorphism: A General Framework forPrivacy Preserving Network Publication. *VLDB*, August 2009.