International Journal of Computational Intelligence Research
ISSN 0973-1873 Volume 13, Number 5 (2017), pp. 965-979

http://www.ripublication.com

Challenges in Social Network Data Privacy

Jyothi Vadisala[1*] and Valli Kumari Vatsavayi[2]

[1,2]Department of CS & SE, College of Engineering(A), Andhra University, Visakhapatnam, India.

Abstract

Due to the advent of computer technology, social networks have become an explosive increasing to a wide variety of applications. Social networks have become an online platform where the people connect and interact with each other as well as share their similar career or personal interests and also for better understanding of interesting phenomena such as sociological and behavioral aspects of individuals or groups. So the social network data are published for various third party consumers such as researchers and advertisers. As a result, there will be a privacy breach which has to be considered. There are different privacy risks and attacks which can breach the privacy in various ways. There are different privacy-preserving techniques have been developed for preserving the privacy of social network data. In this paper, we discuss a various risks and how the adversaries can exploit information to perpetrate privacy on published data. This paper is a survey which helps readers to understand the threats, various privacy preserving mechanisms and their vulnerabilities to privacy breach attacks in social network data publishing.

Keywords: Privacy, Social Networks, Anonymization, Graphs, k-degree.

1. INTRODUCTION

The social network is a web-based application which provides various users to connect, communicate, interact and share the information on the web. There are different social network sites such as Facebook, Twitter and LinkedIn etc. are used for connecting the people and interacting with each other. People create personal profile information for different social network sites to share their ideas, photos, videos, e-mails, instant messaging and also used for finding old friends or finding people who have similar interests or problems across different areas. A study reveals that

India has recorded world's largest growth in terms of social media users in 2013 [1]. A 2013 survey found that 73% of U.S. adults use social networking sites.

Nowadays, social networks are continually growing in number and size, the owners collect an unprecedented amount of information about online social network users. For Example, Facebook statistics say that it has over 1.5 billion monthly and 800 million daily active users. There are wide varieties of datasets which are publicly available to experiment with. The SNAP [2] (Stanford Large Network Dataset Collection) contain different datasets from different sources of varying size. The Flickr dataset hosted on the Amazon web services platform is publicly available for different uses.

Generally, the operators of online social networks collect data from social network service users and publish the anonymized release of data to the third parties like researchers, medical diagnosis, marketing, banking and criminologists. The collected data are rich in content and also contain sensitive data of the users. So the published data should not breach the privacy of the users. As a result, the operators publish the naïve anonymized data i.e. replacing the uniquely identifiable information (like SSN, Aadhra Card) with some random value but it cannot ensure the privacy of users. In [3] shown that, with some knowledge the users are re-identified from the anonymized graph. This shows that basic anonymization techniques are not sufficient for providing the privacy of the users, so there is a serious challenge for network operators to provide the privacy for the network users data which is published to third parties for their needs. Nonetheless, Privacy preserving social network data publishing is a main research area which is concerned with publishing of social network data while preserving the user's privacy.

Privacy preservation of social network data is much more complex than preservation of relational data due to the structural properties of the network data.. This paper presents a survey of recent approaches and challenges to ensure privacy for social network data, privacy attacks and privacy preserving techniques. In this paper, we present a framework for social network threat analysis, categorize various adversarial background knowledge used by adversaries to mount privacy breach attacks on published social network data and we also present a different graph anonymization techniques and metrics that are used regularly to assess the level of anonymity.

The framework of online social network environment is shown in Fig. 1. The users create a profile to connect, interact and exchange information using social media applications. The users can have a membership for more than one social media services. The users create profiles by providing their personal and private information like contact no, address, date of birth and other information. The user interactions also include some sensitive data like their likes, habits, etc. The operators of the social network will maintain the data of the service users and share it to the third party consumers which is used for analytics and researchers for their research purpose. The data collected by the operators often contains sensitive information so the operators release anonymized versions of the complete network or a partial network to the third

party users such as data analytics, medical diagnosis, education, marketings and researchers. The published data also have access to the adversaries where the intention of an adversary to re-identify the particular users from the network.

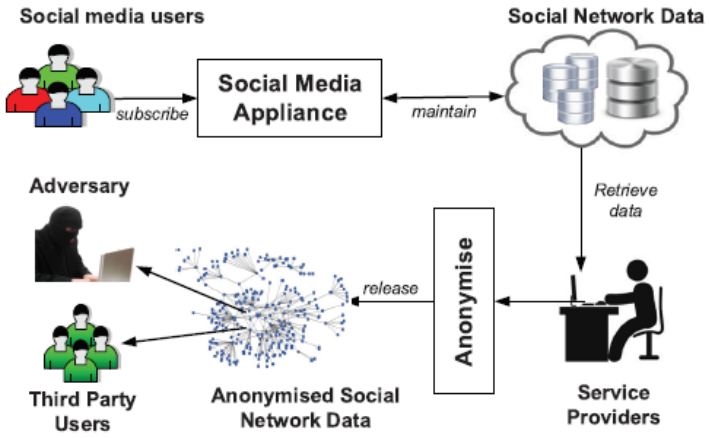

Figure 1: High Level Threat Analysis Framework

2. SOCIAL NETWORK DATA MODEL:

In general, the social network is modeled as a graph $G = (V, E)$ where V represents a set of vertices and E represents set of edges $E \subseteq V \times V$. A Graph G may be directed and undirected. The Facebook social network is represented as a simple undirected graph and whereas Twitter is represented as a directed graph. In a directed graph, the direction of an edge is associated with them. A graph is called multi-graph if it has multiple edges between the vertices. A graph is called a weighted graph where each edge can have weights which can represent some information like degree of friendship. A graph $G' = (V', E')$ be a subgraph of $G = (V, E)$ where $V' \subset V \text{ and } E' \subset E$. Two vertices are adjacent if they share a common edge. A graph is called a simple graph, where the loops and multiple edges are disallowed.

Fig. 2a represents the example of a social network representation with an undirected graph. Fig. 2b is an unlabeled graph of Fig. 2a where the graph has vertex identity only. The graphs which do not have vertex and edge attributes are called as unlabeled graphs. So for unlabeled graphs, the adversary uses the information of graph structure to breach the privacy of an individual. In Fig. 2b, the vertices represent information about an individual or an organization and the edges represent relationships between individuals or relationship between among the organizations in the network. Each vertex has some information like security number, name, income, etc. in addition vertices also has some additional information like age attribute which is shown in Fig. 2c are called vertex labelled graph. Like vertices, edges also have attributes that define the relationship between users, which is shown in Fig. 2d. The graph edges can

have some sensitive information and are public by default on most online social networks and the users can change their default settings rarely.

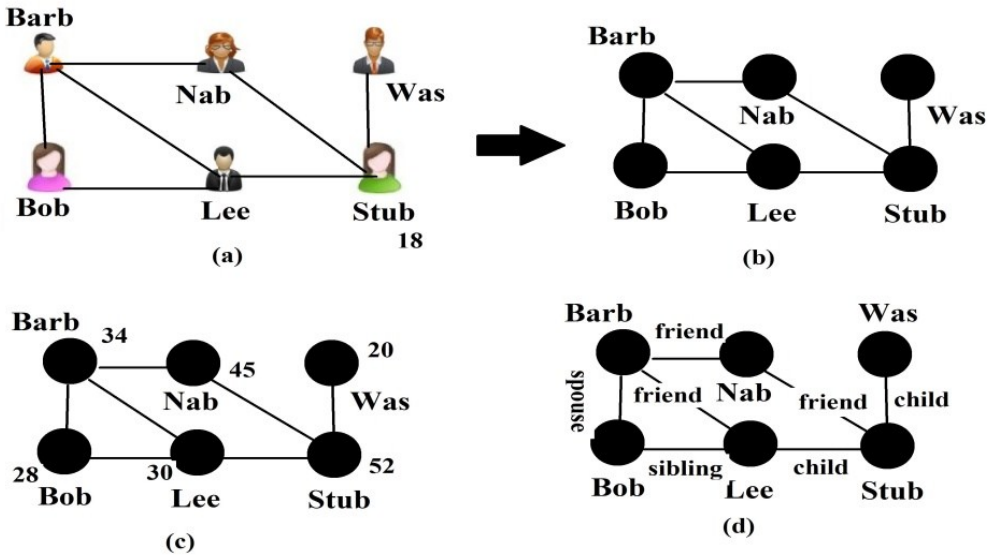

Figure 2: Graphical Representation of Social Network data

2.1. Privacy Breach Risks

The private and sensitive information of an individual is disclosed to unauthorized individuals is called a privacy breach. Generally, network users have a strong perception that the network operators keep their private information secures. The operators commonly anonymize the data before publishing it for use by the third party consumers to ensure the privacy of the social network users. Social network operators are facing an important challenge to maintain online social network user's privacy while publishing the social network data.

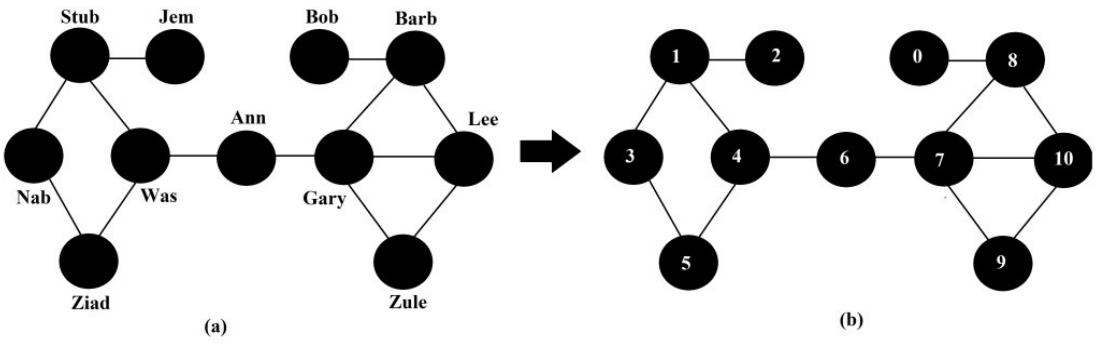

Figure 3: Example of Naïve social network anonymization

Fig. 3 shows the original social network and corresponding naïve social network anonymization. In the naïve anonymization, the identity information associated with

each vertex (i.e. Names) is replaced with a random pseudo-identities (i.e. Numbers). The advantage of the naïve anonymization is that it permits useful analysis of the published social network data. The naïve anonymization can protect the privacy of the individuals with the zero knowledge of an adversary. But, an adversary can exploit a different structural information to re-identify individuals from the anonymized graph [4]. The privacy breach in a network can be grouped into three categories.

- **Identity disclosure:** The identity disclosure occurs when the adversary identifies a particular individual or a user associated with a node is revealed from the anonymized graph. (e.g., User present in a certain disease network).
- **Link disclosure:** Link disclosure occurs when the sensitive relationships between two individuals is revealed. (e.g., A financial transaction have occurred between two nodes).
- **Content disclosure:** When sensitive data associated with the node or a link is compromised (e.g., Political opinion of a node).

In Backstrom et al.[4] these risks on a naïve anonymized graph is considered as active, semi passive and passive attacks. The new vertices and edges are added in active attacks before publishing the social network data. In passive attacks, the graph modifications are not made prior to release the network data. In semi passive attacks, only the edges will be added before publishing the network data. So, the system which is providing privacy preservation over the graphs and networks has to consider all of these issues. There are different approaches for anonymizing the tabular data, but comparing with graph data it is much more challenging for anonymization of graph data rather than tabular data. The reasons are listed below:

- An adversary use variety of background knowledge to breach the privacy. So modelling of background knowledge and the capability of an adversary is difficult. The adversary can use different structural properties to derive the sensitive information. Generally, two nodes that are indistinguishable with respect to any graph structural property need not be indistinguishable on other structural property of the graph. Hence, it is difficult to find what are the most appropriate privacy models for the networks and also how to measure the privacy breach in that structural property.
- The graph information is rich in content, but there are no standardized methods to measure the information loss incurred by graph modifications where the nodes and edges are changed. The importance of the network measures like degree centrality, average shortest path lengths, betweenness, clustering coefficients, etc. to the applications of graph mining (e.g. Identification of communities in the network, clustering, etc.) and also how to preserve these measures. So, it is difficult to measure the information loss for graph data.
- The tabular data contain tuples where each tuple can be viewed as an independent sample from some distribution, but the graph data contain nodes and edges and these are correlated. Therefore, there will be an impact of a

single change of a node or an edge can modify the whole network. So, it is difficult for graph modification algorithms to balance the goals of privacy preservation with the data utility.

Recently, there are different types of privacy models for a variety of background knowledges of the adversaries and graph modification algorithms are developed. But unfortunately there will not be a single model which can solve all the problems. So, protecting against each type of privacy breach may require a different privacy preservation mechanism or a combination of them. In this paper, we summarize the different privacy preservation techniques under different background knowledges. We mainly focus on identity attacks because identity disclosure often leads to both edge and attribute disclosures.

2.2. Adversary Knowledge

The adversary uses a variety of background knowledge to encroach on the privacy of social networks. The adversarial background knowledge plays an important role in understanding the type of the attacks as well as the various protection methods. The background knowledge has referred as an information of network data that an adversary impose a privacy attacks on the published social network data. The adversary can obtain this type of information by crawling or by exploring the overlapping membership of several social networking sites or by stealing the web browsing history, which can be used to re-identify a particular person in the published social network data.

The background knowledge of an adversary can be categorized as personal attributes, edge attributes, structural attributes, auxiliary attributes and subgraph attributes. An adversary can combine these attributes together to breach the privacy of social networks. This information can be easily obtained from the various sources. Backstrom et al. [4] demonstrated that if an adversary knows some information about a graph structure as background knowledge, then the adversary can uniquely identify the vertices of a social network graph from the published social network data.

The personal attributes represent the non-structural information that describes social network users (e.g., name, address, age, salary, marriage status, etc.). These attributes are assigned to the vertex or edge. Some of the personal attributes such as social security number act as a unique identifier. The network user removes these type of attributes before publishing the data. Other personal attributes such as name and address act as quasi-identifiers. Quasi-identifiers may not be sensitive, but an adversary can combine them with other information (e.g., Auxiliary information) to mount sensitive information disclosure attack on the published social network data.

The structural attributes represent the graph information like degree, neighborhood and some other properties which can help an adversary to accomplish privacy attacks on anonymous graphs. The degree of a vertex v is the number of edges incident to that vertex and is represented as $\deg(v) = |\{u|(u,v) \in E\}|$ of a graph $G = (V, E)$. The number of neighbors of a vertex v is the set of vertices adjacent to the vertex v and it

is represented as $N(v) = |\{u|e_{vu} \in E\}|$. These metrics are simple where the adversary can easily obtain and uses as a background knowledge to perpetrate privacy attacks.

The auxiliary information (also referred to as external knowledge) is the information that an adversary is gathered from other sources such as another social network graph which has overlapping users with the published social network graph and group membership of users. An auxiliary social network graph which has overlapping users with the published social network for de-anonymization is also used in [3], [5], [6]. It has been shown that the auxiliary information can be used for a substantial re-identification attack even if it is very noisy.

The adversary also uses a subgraph structure as a background knowledge to breach the privacy from anonymous graphs. For a given Graph $G = (V, E)$, a subgraph is $H = (V', E')\ where\ V' \subseteq V and\ E' \subseteq E$. It contains no vertices or edges that are not in the original network. An embedded subgraph includes subgraphs and special edges within the target social network [4].

In summary, the adversary can use a wide variety of background knowledge to mount an attack on published social network data. It is not possible to model all types of the adversary knowledges and the type of the published graph determines the use of the adversary knowledge.

3. GRAPH ANONYMIZATION TECHNIQUES

From a high level view, the privacy preservation methods can be classified as Graph Modification Methods, Generalization or Clustering Methods and Differential Prvacy Models. In this we will mainly focus on graph modification methods, since they allow us to release the entire network for analysis, providing the widest range of applications for data mining and knowledge extraction.

3.1. Graph Modification Methods

Graph modification approaches anonymize a graph by modifying (adding and/or deleting) edges or vertices in the graph. These methods can be grouped as a Randomization method in which the graph is modified randomly by adding or deleting edges or vertices and another group is a Non Randomization method method in which the graph modification approach consists on edge addition and deletion to meet some desired constraints. The k-anonymity is the most well known models in this group.

- **Randomization Techniques:** In this anonymization, the original graph is modified randomly by adding noise either by adding, deleting, switching edges or vertices and their attributes. Randomization techniques protect against re-identification in a probabilistic manner. Generally, graph randomization techniques can be applied to remove some true edges and/or adding some false edges. One of the strategy is *Rand add/del* method in which randomly adds one edge followed by

deleting another edge which preserves the number of edges in the original graph. Secondly, *Rand Switch* method in which selects a pair of existing edges (v_i, v_j) and (v_m, v_n) randomly and switch the edges to (v_i, v_n) and (v_m, v_j) where (v_i, v_n) and (v_m, v_j) edges do not exist in the original graph. The *Rand Switch* method preserves the number of edges and degree of each vertex.

There are different randomization approaches proposed for privacy preservation in social networks. Hay et al.[7] proposed a *Random perturbation* method in which randomly m edges are removed and then randomly adding n fake edges such that $m = n$ to anonymize unlabeled graphs. This algorithm does not change the set of vertices and the number of edges in the anonymized graph. Ying and Wu [8] proposed *Spctr Add/Del* and *Spctr Switch* randomization methods specifically designed to preserve the spectral characteristics of the original graph. In addition the authors also developed a variation of the Random perturbation method, called *Blockwise Random Add/Delete (Rand Add/Del-B)* method in which the algorithm divides graph into blocks according to the degree sequence and implements modifications by adding or removing edges on the high risk of re-identification, not at random over the entire set of vertices.

Bonchi et al. [9,10] proposed a new information theoretic perspective on the level of anonymity obtained by randomization methods. They made an essential distinction between image and pre-image anonymity and used entropy quantification to measure the level of anonymity provided by the perturbed graph. They stated that the anonymity level quantified by means of entropy is always greater or equal than the one based on a posterior belief probabilities. They also proposed a *Random Sparsification* method in which randomly remove edges without adding new edges. They compare these methods with three datasets which shows that randomization techniques for identity obfuscation may achieve meaningful levels of anonymity while still preserving features of the original graph. Finally, they showed how the randomization method applied for distributed environments where the network data is distributed among several non-trusting sites, and explain why randomization is far more suitable for such settings than other existing approaches.

The randomization methods are simple and easier to implement than the other anonymization techniques. In the random techniques they do not focus on any adversarial attack while graph anonymization process. The recent study of randomization methods shown that they achieve a meaningful level of anonymization and preserve much of the characteristics of the original graph which is proved theoretically and validated through proper experimental evaluation.

- ***k*-anonymization Techniques:** Most of the graph modification approach uses a k-anonymization methods in which the models provide anonymity by adding or deleting edges or vertices of a graph to meet some certain constant value. There are different k-anonymity based methods that primarily differ in the adversary background knowledge have been developed to mitigate the vertex re-identification. The *k*-degree, *k*-isomorphic, *k*-neighborhood and *k*-nmf are some of the examples of

vertex and edge re-identification privacy preserving methods that adopt the k-anonymity model.

Degree Based Anonymization Techniques: Generally, one of the main graph property is the degree of a vertex. In degree based anonymization approaches the adversary uses the degree of a vertex as a background knowledge to identify the particular vertex in the graph. For example, assume that an adversary knows that a target vertex has 4 adjacent vertices in the network. In the naïve anonymized graph, if there is only one vertex has the degree 4 then the adversary can re-identify the targeted vertex.

k-degree anonymity: A Graph $G = (V, E)$ is said to be a k-degree anonymous if for every vertex $v \in V$ in graph G there are at least $k - 1$ other vertices have the same degree of graph G.

The k-degree anonymization problem can be achieved by transforming the original graph G into k-anonymous graph G' with only adding edges or adding fake vertices or both. In these cases, the main optimization is to minimize the number of newly added edges and vertices to preserve the much of the characteristics of the original graph. Liu and Terzi[11] proposed the first k-degree anonymization problem. They developed a heuristic dynamic programming anonymization method which generates original graph into k-anonymous degree sequence by adding edges such that each degree in the anonymous graph is equal to k. Lu et al. [12] proposed a greedy algorithm, called *Fast k-degree anonymization algorithm* that anonymizes the original graph by interleaving the anonymization of the degree sequence with the construction of anonymized graph. Chester et al.[13] proposed k-degree anonymization by adding only fake vertices rather than edge set. The algorithm creates links between fake vertices and original vertices or between fake vertices in order to achieve the k-anonymity. The fake vertices also must be k-anonymous. Tai and Yu[14] proposed a friendship attack model, where an adversary knows the vertex degree pair of two individuals and their friendship relation. They proposed k^2-degree anonymous if for every vertex with an incident edge of degree pair (d_1, d_2), there exist at least $k - 1$ other vertices each of which also has an incident edge of the same degree pair.

Neighborhood Based Anonymization Techniques: In this case the adversary uses the background knowledge of the immediate neighbors of a vertex to disclose the identity of individuals. There are several approaches have been developed for neighborhoods based attacks of social network data publishing. The neighborhood vertex $v \in V$ of a graph G is a subgraph of the neighbors of vertex v of the original graph.

***k*-neighborhood anonymity:** A graph $G = (V, E)$ is k-anonymous if for every vertex $v \in V$ is k-neighborhood anonymous in G if there are at least $k - 1$ other vertices in the graph such that $N(v_1), N(v_2), \ldots N(v_{k-1})$ are isomorphic where $N(v_i)$ is a neighborhood subgraph of vertex v_i.

Zhou and Pei [15] proposed a greedy algorithm called k-neighborhood anonymity which adds vertices and edges to the original graph until such time that a graph with at least k vertices with their neighborhood subgraphs are isomorphic. The algorithm has two steps. First the algorithm groups the vertices and for each $v \in V$, extracts its neighbors and represent its neighborhood subgraphs based on minimum depth first search technique. In the next step, similar neighborhood subgraph vertices are grouped together, then test for the isomorphism of the neighborhoods of different vertices. Now, the algorithm anonymizes same group neighborhoods of vertices by adding vertices and edges to their neighborhoods until $k-1$ vertices are isomorphic neighborhoods as v. The utility loss metric will be the number of edges changed before and after anonymization process. Sun et al. [16] identified a mutual friend attack problem where the adversary knows the number of common neighbors between two connected vertices. They proposed an edge anonymization k-NMF algorithm in which they ensures for each connected edge $e \in E$ there exist at least $k-1$ other edges that share the same number of common neighbors of e in the graph.

Subgraph Based Anonymization Techniques: In this case the adversary uses the subgraph as a background knowledge in which to identify a targeted individual in the original graph. In this the adversary model the knowledge as a query Q that result to a subgraph of the graph G and disclose the vertex identity without the prior structural knowledge of the graph. This can be formalized by the notion of graph automorphism.

***k*-automorphism:** A Graph $G'(V', E')$ is said to be k-automorphic such that for each vertex v there exist at least $k-1$ automorphic functions $\{f_1, f_2, \ldots f_{k-1}\}$ of G' and $f_i(v) \neq f_j(u)$ where $v \neq u$ and $i \neq j$.

Zou et al. [17] proposed the k-automorphism to solve the subgraph based privacy attacks. The anonymization model preserves the privacy by providing at least k-structurally identical subgraphs in the published graph. This approach constructs a graph in which each vertex $v \in V$ is automorphic to at least $v_1, v_2, \ldots v_{k-1}$ other vertices in the graph. This can be achieved by the process of alignment of sub-graphs and addition of edges in the graph. This approach partitions the original graph into a set of unique subgraphs such that each subgraph contains at least k-subgraphs and no subgraphs share a vertex. k-automorphism model able to guarantee privacy under any structural attack. The k-automorphism ensures that the anonymized graph at least $k-1$ automorphism functions such that each function map every vertex to a different other vertex. The information loss is measured as an anonymization cost as defined below:

$$cost\ (G, G') = (E(G) \cup E(G') - E(G) \cap E(G'))$$

Where $E(G)$ indicates number of edges in graph G. The lower cost is an indication of fewer changes to the original graph G. This method also includes statistical measures like clustering coefficients and average shortest path lengths which are used to measure the utility of the released graph. Wu et al. [18] proposed k-symmetric notion in which the adversary has some prior knowledge of any subgraph that contains the individual interest. The k-symmetric is based on the k-automorphism partition which

ensures that every vertex in the published graph has at least $k-1$ automorphically equivalent vertices to it. This approach guarantees that the probability of re-identification is no more than $1/k$. Cheng et al. [19] proposed k-isomorphism by extending the concept of k-symmetric.

***k*-isomorphic graph:** A Graph G is said to be k-isomorphic such that if G is composed of a k disjoint subgraphs where $G = \{s_1, s_2, \dots, s_k\}$ such that $s_i, s_j \in G$ are pairwise isomorphic where $i \neq j$.

In the k-isomorphism approach graph is anonymized by creating k-pair wise isomorphic subgraphs and release them. The algorithm needs more number of vertices to be added in order to make a subgraph isomorphic. So determining the different subgraphs in the graph are isomorphic is an expensive process.

3.2. Generalization Techniques

These anonymization techniques are based on the idea of clustering vertices and edges into groups and then form a super-vertex. The inconvenience of the clustering based methods is that the graph may be shrunk after anonymization and local structures will be difficult to analyze. There are three main classes of clustering-based approaches.

- **Vertex clustering methods:** Vertex clustering methods consist in delivering an anonymized graph which is a generalized graph of the original one, with a super node instead of an original group of nodes. In Hay et al.[20] the nodes of the graph are partitioned into disjoint sets. These nodes are considered as super nodes since they are nodes of a generalized graph. The partitioning of nodes is performed such that the resulting generalized graph maximizes utility and preserves privacy.
- **Edge clustering methods:** Edge clustering methods consist in delivering a representation of the original graph wherein the relational information exists between clusters of vertices. This method consists in leaving the set of edges intact. The edges will only exist between the clusters of vertices.
- **Vertex and Edge Clustering Methods:** Vertex and edge clustering methods consist in partitioning original graph into clusters then combining nodes into a generalized node and edges between clusters into a single edge. In Campan and Truta [21] data of the graph to be anonymized is clustered. For each cluster, the corresponding subgraph is extracted and the nodes of the subgraph are collapsed into a single node. The information about the number of nodes in the cluster is attached to this generalized node as well as the number of edges in the original cluster. Then the inter-cluster edges will be collapsed into a single edge and the structural information released will limit to the total number of edges collapsed into a single edge between the two clusters.

3.3. Differential Privacy Models

Differential Privacy is one of the standard privacy model which is different from the previously described models. All the privacy preservation methods discussed so far will be based on the adversary background knowledge, but the differential privacy model does not depend on background knowledge. Differential privacy relies on some query and result perturbation in order to provide privacy guarantee. This can be achieved in differential privacy is by adding some random noise to the query output. This is realized by using the methods such as Laplace distribution and the normal distribution with variance depending on ϵ and the query's sensitivity.

In social network data publishing, the main goal of differential privacy is to guarantee that an adversary in possession of the published results will not be able to determine that a target vertex appears in graph G or a vertex $i \in V$ and $j \in V$ are friends in the original graph $G = (V, E)$. There are various algorithms have been developed to release statistics about social network data. They are categorized into node privacy and edge privacy methods.

- **Node Differential Privacy:** A privatized query Q satisfies node-privacy if it satisfies differential privacy for all pairs of graphs $G_1 = (V_1, E_1), G_2 = (V_2, E_2)$ where $V_2 = V_1 - x$ and $E_2 = E_1 - \{(v_1, v_2) | v_1 = x \vee v_2 = x\}$ for some $x \in V_1$.

In node privacy, If the social network graph G can be obtained from another graph G' or vice versa by adding or deleting a node and all edges corresponding that node then the graphs are said to be node neighbors to each other. This privacy guarantee completely protects all individuals. Node differential privacy provides protection to the nodes as well as to their adjacent edges. There are different approaches have been proposed to achieve the node differential privacy. Hay et al.[22] introduced the notion of differential node privacy and draw attention to some of the difficulties in attaining it. Smith and Raskhodnikova [23] discuss a node differential privacy algorithm for releasing an approximation to the degree distribution of a graph. Kasiviswanathan et al. [24] discussed different node differential privacy algorithms and also discussed the approaches for analyzing the accuracy of proposed algorithms on real networks.

- **Edge Differential Privacy:** A privatized query Q satisfies edge-privacy if it satisfies differential privacy for all pairs of graphs $G_1 = (V_1, E_1), G_2 = (V_2, E_2)$ where $V_1 = V_2$ and $E_2 = E_1 - x$ where $|E_x| = k$.

In edge privacy, G and G' are said to be edge neighbors if G' can be obtained from the G if k arbitrary edges are removed or added from G. Therefore the edge differential privacy ensures that the adversary will not be able to disclose the presence or absence of a particular edge in the graph.

Nissim et al. [25] considers differential edge privacy in the case of estimating the cost of minimum spanning tree and the number of triangles in a graph and they also discussed algorithms for computing the smooth sensitivity of statistics in a variety of domains. Rastogi et al. [26] focused on differential edge privacy for the case of general subgraph counts release against Bayesian adversary.

5. CONCLUSION:

Social network operators are increasingly publishing and sharing social network data with third party consumers. The published social network data contain potentially sensitive information about users and their relationships. Recent works have shown that de-anonymization of the released data is not only possible but also practical. This has prompted privacy concerns and active research in privacy preserving mechanisms. In this paper, we presented a high level framework for social network publishing threat analysis. We also presented the threat model and quantified and classified the background knowledge that is potentially used by adversaries to breach privacy of the published social network data. We also presented a number of methods, approaches, strategies and techniques in privacy-preserving social network data publishing. In conclusion, privacy-preserving publishing of social networks remains a challenging problem, since graph problems are typically difficult and there can be many different ways for an adversary to exploit both internal and external information to mount attacks.

REFERENCES:

[1] "India records highest social networking growth Rate: Study". *news.biharprabha.com. IANS. 26 July 2014*.

[2] J. Leskovec and A. Krevl. (2014). SNAP Datasets [Online]. Available:https://snap.stanford.edu/data/.

[3] A. Narayanan and V. Shmatikov, "De-anonymizing social networks," in Proc. IEEE Symp. Secur. Privacy, 2009, pp. 173–187.

[4] L. Backstrom, C. Dwork, and J. M. Kleinberg, "Wherefore art thou r3579x?: Anonymized social networks, hidden patterns, and structural steganography," Commun. ACM, vol. 54, no. 12, pp. 133–141, 2011.

[5] W. Peng, F. Li, X. Zou, and J. Wu, "A Two-stage de-anonymization attack against anonymized social networks," IEEE Trans. Comput., vol. 63, no. 2, pp. 290–303, Feb. 2014.

[6] S. Ji, W. Li, N. Z. Gong, P. Mittal, and R. Beyah, "On your social network de-anonymizablity: quantification and large scale evaluation with seed knowledge," in Proc. Symp. Netw. Distrib. Syst. Secur. (NDSS'15), 2015, 1–15.

[7] M. Hay, G. Miklau, D.Jensen, P. Weis and S. Srivastava. Anonymizing Social Networks. Technical Report No. 07-19, Computer Science Department, University of Massachusetts Amherst, UMass Amherst, 2007.

[8] Xiaowei Ying and Xintao Wu. Randomizing social networks: a spectrum preserving approach. In SIAM Conferenc on Data Mining SDM, pages 739–750, Atlanta, Georgia , USA, 2008.

[9] F. Bonchi, A. Gionis, and T. Tassa. Identity Obfuscation in graphs through the information theoritic lens. In 2011 IEEE 27th International Conference on Data Engineering (ICDE), pages 924-935, Washington DC, USA, 2011.

[10] F. Bonchi, A. Gionis, and T. Tassa Identity Obfuscation in graphs through the information theoritic lens. Information Sciences, pages: 232-256, 2014.

[11] K. Liu and E. Terzi, "Towards identity anonymization on graphs," in Proc. ACM Int. Conf. Managament of Data (SIGMOD'08), 2008, pp. 93–106, 2008.

[12] X. Lu, Y. Song, and S. Bressan. Fast Identity Anonymization on Graphs. In 23rd International Conference on Database and Expert Systems Applications (DEXA), pages 281-295, Vienna, Austria, 2012.

[13] S. Chester, B.M. Kapron. G. Ramesh, G. Srivastava, A. Thomo and S. Venkatesh, "k-nonymization of social networks with pseudo nodes", Social Network Analysis and Mining, Vol. , Issue. 3, pp 381-399, 2013.

[14] Chih-Hua Tai, Philip S Yu, De-Nian Yang, and Ming-Syan Chen. Privacy-preserving social network publication against friendship attacks. In Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining, pages 1262–1270. ACM, 2011c.

[15] B. Zhou and J. Pei, "Preserving privacy in social networks against neighborhood attacks," in Proc. IEEE 24th Int. Conf. Data Eng. (ICDE'08),2008, pp. 506–515.

[16] X. K. Chongjing Sun, Philip S Yu, Y. Fu, Privacy Preserving Social Network publication against Mutual Friend Attacks, Transaction on Data Privacy, 2007

[17] L. Zou, L. Chan, and M. T. Ozsu, "K-automorphism: A general frame-work for privacy preserving network publication," in Proc. VLDB Endowment, 2009, vol. 2, pp. 946–957.

[18] W. Wu, Y. Xiao, W. Wang, Z. He, and Z. Wang, "k-symmetry model for identity anonymization in social networks," in Proc. 13th Int. Conf. Extend. Database Technol. (EDBT'10), 2010, pp. 111–122.

[19] J. Cheng, A. W. Fu and J. Liu,"K-isomorphism: Privcy Preservng Network Publication Againsit Structural Attacks" in Proceedings of the 2010 ACM International Conference on Management of Data (SIGMOD 2010), pp. 459-470.

[20] M. Hay, G. Miklau, D. Jensen, D. F. Towsley, and C. Li, "Resisting structural re-identification in anonymized social networks," Very Large Database J., vol. 19, no. 6, pp. 797–823, 2010.

[21] A. Campan and T. Truta, "A Clustering Approach for Data and Structural Anonymity in Social Networks", in proceedings of the 2nd ACM SIGKDD International Worskhop on Privacy, Security and Trust in KDD (PinKDD'08), 2008.

[22] M. Hay, C. Li, G. Miklau and D. Jensen. Accurate estimation of the degree distribution of private networks. Proceedings of the 2009, 9th IEEE International Conference on Data Mining, pp. 169-178, 2009.

[23] S. Raskhodnikova and A. Smith, Efficient Lipschitz Extensions for High Dimensional Graph Statstics and Node Private Degree Distributions, CoRR, 2015.

[24] S. P. Kasiviswanathan, K. Nissim, S. Raskhodnikova and A. Smith, "Analyzing Graphs with Node Differential Privacy" Proceedings of 10th theory of cryptography conference on Theory of Cryptography, pp. 457-478, 2013.

[25] K. Nisim, S. Raskhodnikova, A. Smith, Smooth Sensitivity and sampling in privte data nalysis. In Symp. Theory of Computing (STOC), pp.75-84, 2007.

[26] Rastogi V, Hay M, Miklau G, Suciu, "Relationship Privacy: Output perturbation for queries with joins", in proceedings of the 28th ACM-SIGMOD-SIGACT-SIGART symposium on principles of databse systems (PODS'09), ACM , New Yor, pp. 107-116, 2009.