Anonymizing Shortest Paths on Social Network Graphs

Shyue-Liang Wang[1], Zheng-Ze Tsai[1], Tzung-Pei Hong[2], and I-Hsien Ting[1]

[1] Department of Information Management
[2] Department of Computer Science and Information Engineering
National University of Kaohsiung
Kaohsiung, Taiwan 81148

Abstract. Social networking is gaining enormous popularity in the past few years. However, the popularity may also bring unexpected consequences for users regarding safety and privacy concerns. To prevent privacy being breached and modeling a social network as a weighted graph, many effective anonymization techniques have been proposed. In this work, we consider the edge weight anonymity problem. In particular, to protect the weight privacy of the shortest path between two vertices on a weighted graph, we present a new concept called *k-anonymous path privacy*. A published social network graph with *k-anonymous path privacy* has at least k indistinguishable shortest paths between the source and destination vertices. Greedy-based modification algorithms and experimental results showing the feasibility and characteristics of the proposed approach are presented.

Keywords: Social networks, privacy preserving, edge weight, shortest path, k-anonymity.

1 Introduction

Privacy preserving data mining, privacy preserving data publishing, and privacy preserving network publishing have attracted considerable attention in recent years because of the concern of breaching privacy from published data. Social network applications, such as MySpace and Facebook and other online communities, collaboration networks, telecommunication networks, have become very popular for sharing information. There are millions of registered users associated with others through friendships, hobbies, professional association, and so on. These user information and relationship can be modeled as vertices, edges, and edge weights in complex graphs and are of significant importance in various application domains such as marketing, psychology, epidemiology and homeland security. As a result, companies and institutions hosting the data are interested and expect to be beneficial in releasing portions of the graphs so that research communities can analyze the data. However, these social network graphs may contain sensitive information. In order to protect the privacy of users against different types of attacks, graphs should be anonymized before they are published.

Some current practices to protect user privacy from published data include removing all identifiable personal information such as names and social security numbers,

N.T. Nguyen, C.-G. Kim, and A. Janiak (Eds.): ACIIDS 2011, LNAI 6591, pp. 129–136, 2011.

limiting access, "fuzzing" the data, eliminating unnecessary groupings, augmenting with additional data, etc. However, it is still easy for an attacker to identify the target by performing different structural and non-structural queries. Let's consider the following examples of re-identification attack on relational data, transaction data, and graph data.

For published relational data, given a public voter registration data and a private microdata such as the de-identified (name and social security number removed) patient data of Massachusetts's state employees, a simple "linking" attack by joining the two datasets can re-identify the identity and medical history of the state's governor. According to one study, approximately 87% of the population of the United States can be uniquely identified on the basis of their 5-digit zip code, sex, and date of birth [1, 13, 14].

For published transaction data, America Online (AOL) released a large portion of its search engine query logs for research purposes in August 2006. The dataset contained 20 million queries posed by 650,000 AOL users over a 3 month period. Before releasing the data, AOL replaces each user's name by a random identifier. However, by examining unique query terms, the New York Times [2] demonstrated that the searcher No. 4417749 was traced back to Thelma Arnold, a 62-year-old widow who lives in Lilburn, Georgia. Despite a query does not contain address or name, a searcher may still be re-identified from combination of query terms that are unique enough about the searcher.

For published graph data, even when a network is published without any identity information, it is still possible to locate the target with high probability based on some structural information around the target [5, 17]. Similar to the quasi-identifiers in relational or set-valued data that can be used as background knowledge for re-identification; any topological structure of the network can be utilized to identify the target in a released network. There have been four types of structural attacks in this environment [5, 7, 16]: degree-attack, subgraph attack, 1-neighborhood attack, and hub-fingerprint-attack. It is also possible that an attacker can also launch a query based on non-structural information (such as vertex label) to identify the target.

There are basically three types of sensitive information that one may want to keep private and may be under attack in a social network environment: node information, link information and edge weight information [3, 8, 9]. The node information is the information attached to a vertex. For example, the emails sent by an individual, the personal information such as age, sex, zip code, and transaction data such as purchased items [6, 10-12]. The link information is about the relationships among the individuals which may be considered sensitive. Links can be used to represent financial exchanges, friend relationships, conflict likelihood, sexual relations, disease transmission [9]. Depending on the application, the edge weight information can semantically represent "degree of friendship", "trustworthiness", and "behavior" etc. If considering routing problem, (for information spread and marketing), edge weights may correspond to the cost of information propagation [4]. To protect node information, many generalization and suppression-based k-anonymity techniques for relational and set-valued data have been proposed. To protect link information, there are some studies such as k-degree, k-automorphism, k-isomorphism privacy models addressing various types of structural attacks. To protect edge weight privacy, perturbation-based approaches to preserve linear property such as shortest paths by anonymizing the edge weights have been proposed recently [4, 8, 9]. In this work, we consider the problem of anonymizing the shortest path by minimally modifying edge weights such that the published social network graph reveals at least k shortest paths between source and destination vertices. We define a new concept called *k-anonymous path privacy*. Greedy-based modification algorithms and experimental results showing the feasibility and characteristics of the proposed approach are presented.

The rest of the paper is organized as follows. Section 2 gives the problem description. Section 3 describes the proposed algorithms. Section 4 reports the numerical experiments. Section 5 concludes the paper.

2 Problem Description

Recent studies in privacy preserving social networks have proposed many novel models and anonymization approaches. Most of them model the social networks by un-weighted graphs. It then perturbed the graphs before the publication in order to conceal the identities of vertices or link relationships among group of vertices.

However, weighted graphs can be used for analyzing the formation of communities within the network, business transaction networks, viral and targeted marketing and advertising, modeling the structure and dynamics such as opinion formation, and for analysis of the network for maximizing the spread of information through the social links [4]. Depending on the applications, the edge weights could be used to represent "degree of friendship", "trustworthiness", and "business transaction", etc.

In order to protect the privacy of these sensitive information (sensitive edges), current works concentrate on preserving the shortest paths characteristic between pairs of vertices [4, 9] and k-anonymous weight privacy [8]. To preserve the shortest paths between pairs of vertices, Gaussian randomization perturbation and greedy perturbation techniques that minimally modify the edge weights without adding or deleting any vertices and edges have been proposed. A linear programming abstract model that can preserve linear properties of edge weights (including shortest paths) after anonymization is presented in [4]. To eliminate the distinguishability between edge weights, the k-anonymous weight privacy is defined as[8]: the edge $(i \rightarrow j)$ is k-anonymous if and only if there exist at least k edges in $\Phi(i)$ whose weights $w_{i,tl}$, $l = 1, ..., c$, and $c \geq k$, satisfy $\| w_{i,j} - w_{i,tl} \| \leqq \mu$, l=1,..., c. Here, μ is a predefined positive parameter to control the degree of privacy and $\Phi(i)$ is the adjacent edge set in which all edges come from the i-th node.

In this work, we consider a different type of privacy, *k-anonymous path privacy*, that hiders adversary to infer the sensitive relationship between two entities (vertices) in a social network. The basic idea is to hide the true sensitive information, e.g. the shortest path, by obfuscating it with at least *k-1* other paths so that the true path will not be revealed. Figure 1 shows an undirected weighted graph with six vertices. Assuming the relationship represented by the shortest path between vertices v_1 and v_6 is sensitive, $\{(v_1, v_5), (v_5, v_4), (v_4, v_6)\}$, and expected to be hidden. One possible technique is to perturb minimal number of edge weights so that there will be k shortest paths between the two vertices. Figure 2 shows the anonymized graph with two shortest paths $\{\{(v_1, v_5), (v_5, v_4), (v_4, v_6)\}, \{(v_1, v_5), (v_5, v_6)\}\}$, where the weight of edge $e_{5,6}$ is modified to two. Therefore, given a graph G, a set of source and destination nodes H, and privacy level k, the objective of *k-anonymous path privacy* is to minimally modify the graph such that there exists *k* shortest paths between each given pair of nodes specified in *H*, without adding or deleting any vertices or edges. The privacy level *k* is treated as the number of shortest paths between the specified source and destination vertices in this work.

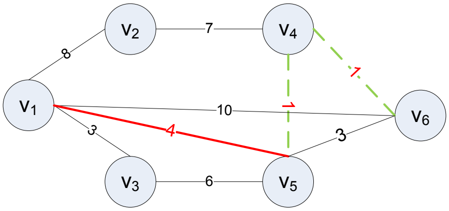

Fig. 1. The Original Network

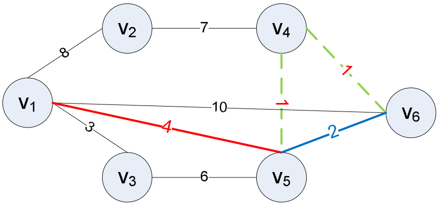

Fig. 2. The Anonymized Network with k=2

3 Proposed Algorithms

For one given pair of source and destination vertices, the objective of *k-anonymous path privacy* is to minimally modify the graph so that *k* shortest paths can be achieved. We propose a greedy-based approach and modify the edge weights in the top-k shortest paths so that they all possess the same path length. The proposed algorithm first finds the second shortest path and reduces proportionally the edge weights of non-overlapping edges between the second shortest and the shortest paths. For example, in Figure 1, the second shortest path between v_1 and v_6 is $p_{1,6} = \{(v_1, v_5), (v_5, v_6)\}$. The non-overlapping edge is $e_{5,6}$ which has weight three. The non-overlapping edges in the shortest path are $e_{5,4}$ and $e_{4,6}$ which has total weight of two. Therefore the edge weight of $e_{5,6}$ is reduced to two so that both paths will have the same path length. The process repeats itself after all top-k shortest paths are modified. The proposed algorithms for anonymizing single pair of source and destination vertices (K-Single Path anonymization algorithm; KSP) and multiple pairs of source and destination vertices (K-Multiple Path anonymization algorithm; MSP) are given in the following. For simplicity, we assume that edge weights can be modified only once in this work.

The following notations will be used:
v_i: vertex *i;*
$e_{i,j}$: edge between vertices v_i and v_j;

$w_{i,j}$: weight of edge $e_{i,j}$;
$p_{i,j}$: path between vertices v_i and v_j;
$d_{i,j}$: length of path $p_{i,j}$;
SPL: Shortest Path List;
TSPL: Temporary Shortest Path List;

Given a graph *G*, a source vertex and a destination vertex (v_i, v_j), and privacy level *k*, the objective is to minimally modify the graph such that there exists *k* shortest paths between the given pair of vertices, without adding or deleting any vertices or edges.

K-Single Path Anonymization Algorithm (KSP)
Input: W, weighted adjacency matrix of a given graph G,
The source and destination vertices for which the shortest path is to be anonymized,
K, number of shortest path between each pair of source and destination vertices,
Output: anonymized weighted adjacency matrix W*,

```
Find the shortest path $p_{i,j}$ & its length $d_{i,j}$;
For ($j$ = 2 to $k$)
  {Find the $j$-th shortest path $p'_{i,j}$ & its length $d'_{i,j}$;
    For (each edge $e'_{p,q}$ on $p'_{i,j}$ that is non-overlap with top $(j-1)$ shortest paths)
       {$w'_{pq} = w'_{pq} + \frac{w'_{pq}}{\sum w'_{pq}} \times (d'_{ij} - d_{ij})$; //summation over non-overlapping edges
        Update the adjacency matrix;
       }; // end of for each edge
  }; // end of for j = 2 to k
```

For a set of source and destination vertices *H*, a given privacy level *k*, the *k-anonymous path privacy* problem is to minimally modify the graph *G* such that there exists *k* shortest paths between each given pair of vertices specified in *H*, without adding or deleting any vertices or edges. The following algorithm further assumed that anonymized paths cannot be modified again when anonymizing other sets of paths (for different pairs of source and destination vertices).

K-Multiple Paths Anonymization Algorithm (KMP)
Input: W, weighted adjacency matrix of a given graph G,
H, the set of source and destination vertices for which the shortest paths are to be anonymized,
K, number of shortest path between each pair of source and destination vertices,
Output: anonymized weighted adjacency matrix W*,

```
Initialize $SPL = \phi$; //shortest path list
while ($H \neq \phi$)
  {for (each pair of vertices $(v_i, v_j)$ in $H$)
```

```
        find its shortest path $p_{i,j}$ and length $d_{i,j}$;
    $d_{r,s} := \min_H d_{i,j}$ ; //minimum of all shortest paths
    $H := H - \{(v_r, v_s)\}$;
    $TSPL := \{p_{r,s}\}$; //the shortest path for $(v_r, v_s)$
    while ($|TSPL| < k$ ) //anonymizing k-1 paths
        {find next shortest path $p'_{r,s}$ and its length $d'_{r,s}$;
        if ($d'_{r,s} = d_{r,s}$) //same length
        {$TSPL := TSPL + p'_{r,s}$ ; // add to anonymized list
        continue;} //find next shortest path
        else // different length
        {let $diff := d'_{r,s} - d_{r,s}$;
        $p''_{r,s} := p'_{r,s} -$ {edges in $SPL$ and $TSPL$};
        If ($p''_{r,s} \neq \phi$ and $d''_{r,s} > diff$) //available edges
            for (each edge $e''_{i,j}$ on the path $p''_{r,s}$ )
              {$w''_{ij} = w''_{ij} + \frac{w''_{ij}}{\sum w''_{ij}} \times (d'_{rs} - diff)$;
                update the adjacency matrix;
                $TSPL := TSPL + p'_{r,s}$ ;
                }; // end of for each edge $e''_{i,j}$
              }; // end of if/else
        }; // end of while ($|TSPL| < k$ )
    $SPL := SPL + TSPL$;
    }; // end of while ($H \neq \phi$)
```

4 Numerical Experiments

To evaluate the characteristics of the proposed algorithms, we run simulations on a real data set, EIES (Electronic Information Exchange System) Acquaintanceship at time 2 collected in [15] and can be downloaded from International Network of Social Network Analysis website. The EIES at time 2 is a network of researchers who participated in an early study on the impact of a computer conference on the formation of interpersonal ties among scientists. The measure of acquaintanceship between users has four levels, ranging from 1 (do not know the other) to 4 (very good friendships). The data set contains 48 users and 830 acquaintanceships and is modeled as a weighted graph.

All experiments reported in this section were performed on an Intel Core 2 Duo P8700 CPU, 2.53 GHz machine with 4 GB main memory, running Microsoft Windows 7 operating system. All the methods were implemented using Java programming language.

Figure 3 shows the preliminary results of ratios of perturbed edges. The ratio is the number of modified edges over the total number of edges on the k shortest paths. It can be observed that the percentage of perturbed edges remain quite stable for different privacy level k and for multiple pairs of source and destination vertices. For anonymizing one pair of source and destination vertices (*H1)*, the average ratio of five randomly selected pairs is about 40%. The average running time is 0.61 seconds. For anonymizing two pairs (*H2*) and three pairs (*H3*) of source and destination vertices, the average ratios of five randomly selected pairs are about 39% and 36% respectively. The average running times are 2.77 and 4.49 seconds respectively. Figure 4 shows the running times of anonymization of *k* shortest paths for different privacy level *k* and for multiple pairs of source and destination vertices. It can be observed that anonymizing multiple pairs of source and destination vertices require relatively more running time when *k* increases to 10. This is due to the fact that it takes longer time to search for extra paths to be anonymized. However, the required level of privacy *k* would depend on applications and is usually not very large.

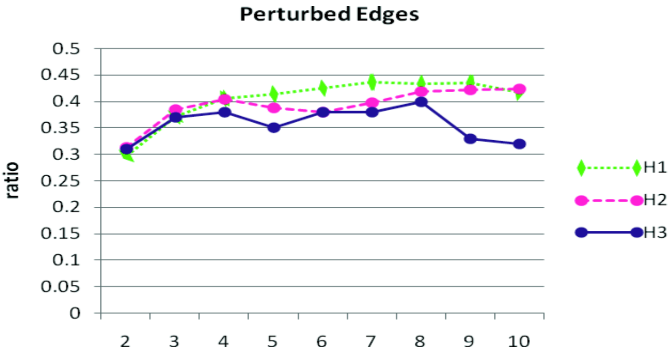

Fig. 3. Ratios of perturbed edges for different *k*

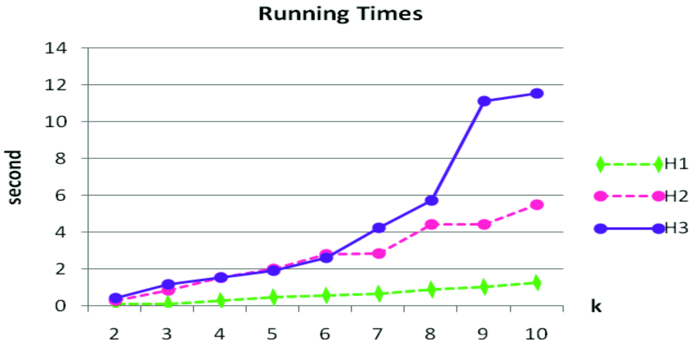

Fig. 4. Running times for different *k*

5 Conclusions

In this work, we have studied the problem of preserving sensitive paths in social networks. We proposed a new concept called *k-anonymous path privacy* and algorithms that minimally perturbed the edge weights to achieve the path anonymity. Examples illustrating the approach and numerical experiments showing the characteristics of the proposed approach were given. It demonstrates that the proposed technique is feasible to achieve the *k-anonymous path privacy*. In the future, we will consider overlapped shortest paths and preserving other types of sensitive characteristics and privacy such as minimal cost spanning trees and others.

Acknowledgments. This work was supported in part by the National Science Council, Taiwan, under grant NSC-99-2221-E-390-033.

References

1. Backstrom, L., Huttenlocher, D.P., Kleinberg, J.M., Lan, X.: Group formation in large social networks: membership, growth, and evolution. In: KDD, pp. 44–54 (2006)
2. Barbaro, M., Zeller Jr., T.: A face is exposed for AOL searcher no. 4417749. New York Times (August 2006)
3. Cheng, J., Fu, A., Liu, J.: K-isomorphism: privacy preserving network publication against structural attacks. In: SIGMOD Conference, pp. 459–470 (2010)
4. Das, S., Egecioglu, O., Abbadi, A.E.: Anonymizing weighted social network graphs. In: ICDE, pp. 904–907 (2010)
5. Hay, M., Miklau, G., Jensen, D., Towsley, D.F., Weis, P.: Resisting structural re-identification in anonymized social networks. PVLDB 1(1), 102–114 (2008)
6. He, Y., Naughton, J.F.: Anonymization of set-valued data via top-down, local generalization. In: VLDB (2009)
7. Liu, K., Terzi, E.: Towards identity anonymization on graphs. In: SIGMOD Conference, pp. 93–106 (2008)
8. Liu, L., Liu, J., Zhang, J.: Privacy preservation of affinities in social networks. In: ICIS (2010)
9. Liu, L., Wang, J., Liu, J., Zhang, J.: Privacy preservation in social networks with sensitive edge weights. In: SDM, pp. 954–965 (2009)
10. Meyerson, A., Williams, R.: On the complexity of optimal k-anonymity. In: Proc. of PODS (2004)
11. Motwani, R., Nabar, S.U.: Anonymizing unstructured data, arXiv: 0810.5582v2, [cs.DB] (2008)
12. Park, H., Shim, K.: Approximate algorithms for k-anonymity. In: Proceedings of the 2007 ACM SIGMOD International Conference on Management of Data, pp. 67–78 (2007)
13. Samarati, P., Sweeny, L.: Generalizing data to provide anonymity when disclosing information. In: Proc. of ACM Symposium on Principles of Database Systems, p. 188 (1998)
14. Sweeny, L.: k-anonymity: a model for protecting privacy. International Journal on Uncertainty, Fuzziness and Knowledge-based Systems 10(5), 557–570 (2002)
15. Wasserman, S., Faust, K.: Social network analysis: methods and applications. Cambridge University Press, New York (1994)
16. Zhou, B., Pei, J.: Preserving privacy in social networks against neighborhood attacks. In: ICDE, pp. 506–515 (2008)
17. Zou, L., Chen, L., Ozsu, M.T.: K-automorphism: A general framework for privacy preserving network publication. In: VLDB (2009)