Indian Journal of Science and Technology, *Vol 9(37), DOI: 10.17485/ijst/2016/v9i37/93810, October 2016*
ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645

Privacy Preservation in Social Network Analysis using Edge Weight Perturbation

Nayan Mattani, J. Sharath Kumar, A. Prabakaran and N. Maheswari

School of computing science and Engineering, VIT University, Chennai - 600127, Tamil Nadu, India;
nayan.mattani2015@vit.ac.in, sharathkumar.j@vit.ac.in, prabakaran.a2011@vit.ac.in, maheswari.n@vit.ac.in

Abstract

Objectives: This paper focuses on a privacy preservation technique which is applied on graphs to preserve sensitive information present as shortest paths. **Methods/Statistical Analysis**: Provide privacy in Social Network Analysis by protecting the sensitive edge weights with the help of preserving the nearest shortest path lengths as well as shortest paths so that individual confidential information can be protected from multiple type of attacks. This research work provides more privacy then greedy perturbation technique in social network analysis. **Findings**: Privacy preservation has a tradeoff between the utility of data and preservation of sensitive information. This is achieved by modification of shortest path length in graphs and also maintains the structure of the graph. This procedure enhances the privacy of sensitive information with minimal concerns to utility. **Application/Improvements:** The privacy preservation technique of edge weight perturbation is applied to social graphs in a small user group to preserve sensitive information when data is shared within the group members. The edge weight perturbation algorithm can be improved by combing the algorithm with the preservation techniques for the user nodes.

Keywords: Anonymization, Graphs, Perturbation, Privacy, Social Networks

1. Introduction

Social Network data consists of entities represented as either individuals, companies, groups, organizations called as nodes that are joined by one or more properties and connections or links between these entities are called edges which indicates some kind of relationship (flows) between these nodes. This flow may carry all sorts of data, so in a social network all the nodes and edges are interconnected. Social Network Analysis is a technique used for investigating the mapping and measuring of social structures consisting of either nodes, peoples, groups, organizations and other connected knowledge entities and links representing edges between these nodes. Social Network Analysis is used in geography, information science, sociology etc[1,2]. To perform analysis on Social Network data, data is collected from multiple sources and then data is shared online. Data collected from Social Networks may contain very confidential and sensitive information about the individuals or users[3]. A social graph usually is acyclic, undirected graph, weighted, these information usually are used by attackers to reveal the identity of the user[4]. The thing that we are facing today is securing the confidential information and takes advantages form Social Network analysis. Any information released in Social Network such as Face book, Twitter, etc. that include entities and links between these entities may lead to privacy implications for involved users. Privacy breach occurs when individuals or organizations confidential and sensitive information is disclosed to an adversary[5].

Privacy breaches can be classified into three categories[3]

- **Identity Disclosure**: This disclosure occurs when a particular individual or user behind a record is disclosed. This type of breach leads to leakage of information of a user and type of relationship he shares with any other individuals in

**Author for correspondence*

any social network. Naive Anonymization is a method that uproots specifically distinguishing data connected with every node or replaces it with whatever other pseudo-irregular name.

- **Sensitive Link Disclosure:** This disclosure occurs when any edge or connection or relation between two entities is leaked. Social media reveals information when they got something special regarding any superstar or any minister.
- **Sensitive Attribute Disclosure:** This disclosure occurs when an adversary gains someone personal, sensitive and confidential individual attribute.

Privacy preserving is a method or technique that protects individual and any confidential information in social network[6]. So privacy preservation of individuals while sharing individual's collected information in social network is an important research area.

Initially the degree of node was considered indicating the number of edges connected to that node with different ones. The enhancement was also there considering the isomorphism, clustering, group formation, changing the structure of graph considered as privacy[7,8]. Increasing the use of social network applications and continuous development, there is a need to protect the individual, community or company's personal, private and solid information should be taken into account for preserving privacy.

Different researches have tried enhancing the privacy in social networks by taking or observing different sequences as privacy and they tried experiments on the small networks which are mostly undirected graphs, connections mainly represent some kind of relationships which does not consider any other weights of that relationship[9,10]. Edge deletion with random choice is considered for privacy preservation in social networks[11]. The structural properties of the graph should be maintained in order to increase the utility of the social graph.

In many cases, weight on the edges define some meaning like consider the network of researchers in which weight on edges represent relations as 1 indicates friends, 2 indicates close friends, 3 indicates don't know each other. And consider the disease transmission network in which the weights represent the kind of disease particular individual is suffering.

Consider the real life example of business network in nodes represents the companies or agents and link or connection between these agents and companies represents some transactions happened between them, these transaction expenses are there in million per month[12] . Business network have company A, company D and different intermediate agents are Agent 1, Agent 2, Agent 3. Suppose company wants to do some contract with company D later and there doesn't exist any path there directly from A to D, so contact can be done with the help of agents. Company A will choose that agent who has contact with both these A and D and whose is having the most shortest path containing less transaction price among A and D. If the transaction on the edges changes after applying some operation but the same shortest path of transaction is there after changing of transactions then company A will take some proper decision without having the details of company D and agents.

After considering the shortest path as a utility, preservation of the shortest path in Social Network is very important because already work exist on the un-weighted graphs and the researchers focused on privacy considering the re-identification of nodes and edges. The shortest path plays a vital role as data utility between nodes pairs having applications in different fields like location searching in Geographic Information Systems, minimum delay will be there in communications path. The precise and depth of un-weighted graph is weighted graphs. The enhanced algorithms are expanded in weighted graphs problems.

In this paper, privacy has been preserved by taking into account the weighted edges by maintaining the exact shortest path .The paper is organized by describing contents: An introduction to the related work and different existing perturbation techniques are in Part II. Comparison of enhanced method and existing greedy perturbation method are described in Part III. Results of experiments of both enhanced method and greedy method are explained in Part IV. Detailed conclusion is finally given in Part V after describing above all contents.

Social Network in today's age gains remarkable attention of users. Analysis of Social Networks is helpful in order to obtain any information about individuals, organizations etc. In order to extract any information regarding individuals, groups, organizations and connection between individuals different data mining techniques have been proposed in privacy preserving data mining which gives the guarantee that mining results are very nearer to original data and these methods also maintain the path data utility without revealing any information[12]. From the existing mining and analysis methods, two sec-

tions are stated. First section include the techniques in which modification of algorithms is done in order to perform mining operations on datasets even after not having the exact or correct values of data.

Second section includes the techniques which changes the values of the dataset in order to provide the privacy to data values. These techniques are made in such a way that it changes the entire data values or only some private or confidential parts of data using some randomization or perturbation[13–15].

PPDM techniques can be classified into two types, 1. Perturbation methods consist of generalization, suppression, additive or multiplicative factor, fuzzy based or geometric projections and random number projections, 2. Cryptography based method used for hiding the data[16].

Social Network data is not presented like that of relational data in a matrix format. Many people don't use any relational algorithm to protect social network data[17]. So the traditional methods are not useful for graph data due to the complexity in the structure of the data. Privacy can be provided by means of different de-identification to provide the protection of entity in a network. Different method has been provided in order to protect the nodes form the attackers who is already having information about some neighbors by the addition and the removal of the un-weighted edges[18]. A new technique was used to consider only the sensitive labels of edges and then deleted these edges with sensitive information, using clustering techniques.

Different methods described above considers either node or edge privacy. The paper only considers the edge weight privacy. The existing techniques include greedy perturbation and Gaussian randomization which provides the edge weight privacy by maintaining the shortest paths.

2. Weight Perturbation Shortest Path Technique

The most important concept in both techniques is the perturbation. Perturbation is one method of modification. Perturbation is the alteration of the attribute value by new value. In real World Perturbation hides the original cost and showing something different to the world with some perturbed cost[19]. Now consider the simple graph consisting of six vertices and nine edges as shown in Figure 1.

The cost of edge between the vertices is stored in W set. As we see from figure that the weights between the vertices are 6, 7, 6, 9, 13, 10, 5, 10, 25 and vertices are V1, V2, V3, V4, V5 and V6.

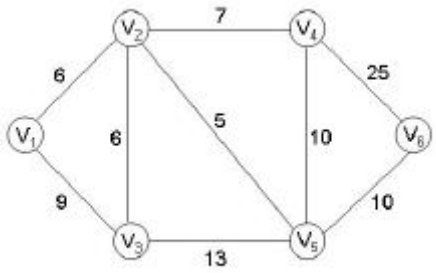

Figure 1. Simple Social Network G.

After applying the concept of perturbation the weight function of the edges changes. The edges, vertices and graph in the original graph are denoted by the E,V and G (V,E) while in the perturbed graph they are represented by E*,V*,G*. When applying the perturbation strategy the vertices always remains the same but only weight function changes and shortest path changes[19]. The perturbed Social Network graph is shown in Figure 2. The 90% of the edge weight changes and only 10% remains same. From the perturbed graph we clearly see that the edge between the vertex V2 and V4, V1 and V2, V1 and V3, V3 and V5, V4 and V5, V4 and V6, and V5 and V6 changes.

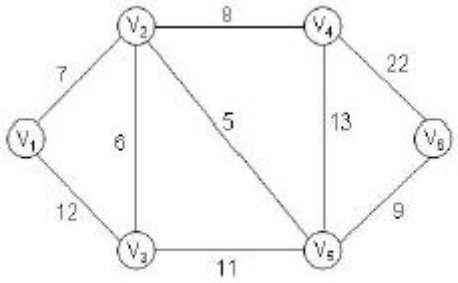

Figure 2. Perturbed Social Network G* of G.

3. Greedy Perturbation Technique

The greedy perturbation algorithm keeps the same shortest path before and after perturbation as well as tries to keep the length of shortest path as nearer to original

shortest path[19,20]. Let's start first describing the greedy perturbation technique. In real world, all the data can't be preserved protected. Some shortest path between the node pairs can be protected and H denotes that set of pairs that we keep protecting.

Social Network graph can be represented as G consists of E representing the edge list and V containing the number of vertices in an network and shortest path P and path length is generated from n*n matrix. wi, j, ps1, ps2 and ds1, s2 indicates the original weight of an edge, shortest path between s1 and s2 while ds1, s2 represents the shortest path length between s1 and s2. After applying the perturbation, w*i, j , p*s1, s2 , d*s1, s2 represents the perturbed weight, shortest path and perturbed path length.

Develop a perturbed graph which fulfills the below conditions:

- Number of vertices should be equal in both original and perturbed graph.
- Number of edges should be equal in both original and perturbed graph.
- Maximize the number of perturbed weight such that perturbed weight is not equal to original weight.
- The perturbed weight try to keep the same as that of original weight of the edge.
- The path should be same in both perturbed graph and original graph.

Depending upon combining the above conditions, edges in graph are classified into different categories.

- Non betweenness edge: An edge is called non-between edges when it does not pass through even single shortest path.
- All betweenness edge: An edge is called all-betweenness edge when the entire shortest path passes through that edge.
- Partial betweenness edge: An edge is called partial-between edge when it pass through even single shortest p.

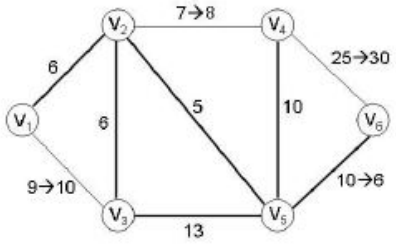

Figure 3. Edge weight modification for non visited edges.

Figure 3 shows all the types of edges. Consider three shortest path pairs stored in H = {(1,6), (4,6), (3,6)}. Edges e1, 3, e2, 4, e4, 6 and e3, 5 represent non betweeness edge (light edge) because no shortest path lengths pass through that edge. Edges e1, 2, e2, 5, e2, 3 and e4, 5 represents partial betweenness edge (bold edge) because only some shortest path lengths pass through that edges and edge e5, 6 represent all betweenness edge because all shortest path lengths p1, 6, p4, 6 and p3, 6 pass through that edge. Perturbation is done on all these edges based on different definitions. For non betweenness edges if weight of an edge is increased by random number r (w*i, j = wi, j+r) all shortest path lengths and shortest paths in H will not be changed. For all betweenness edges if weight of an edge is decreased by random number r (w*i, j = wi, j-r) all shortest paths in H will not be changed but shortest path lengths in H will be decreased (d*s1, s2 =ds1, s2-r). There are special criteria for the partial betweenness edges whether to increase the weight or to decrease the weight of that edge. In the social networks most of the edges are partial betweenness edges. So this algorithm focuses mostly on partial betweenness edges. Greedy algorithm increase the partial betweenness weight if t less than difference between the length of conditional shortest path and the original shortest path and we increase the weight of partial betweenness edge by t. The value of t comes by taking the minimum of difference between node pair among all node pairs which are there in set. The weight of the partial betweenness edge is decreased based on some criteria. The greedy algorithm not only keeps the same shortest before and after perturbation and also tries to keep shortest path length close to that of original one.

4. Weight Perturbation Shortest Path Technique

This technique not only concentrates on the partial betweenness edge but also take advantages of the non betweenness edges. Advantage of the non betweenness can be taken by swapping the partial betweenness edge weight by minimum from all of the non betweenness edges. If the weight of partial betweenness is more than minimum among all non betweenness edge weights than that particular partial betweenness edge weight is replaced by the minimum non betweenness edge weight.

This perturbation technique gives good results than greedy perturbation in some metrics[20]. In this technique

two concepts are playing vital role and that concepts are perturbation and the swapping. Initially the original graph is perturbed and after perturbation we are applying the concept of swapping. Initially greedy algorithm is followed and when there is need to replace the edge weight of partial betweenness swapping operation is performed by comparing weights of partial betweenness and non betweenness. The shortest path will be the more private because partial betweenness edge weight is replaced by minimum non betweenness edge weight.

5.1 Weight Perturbation Shortest Path Algorithm

1. Generate P and D based on W, and assign D to D*
2. for all non betweenness edges e(i,j)
3. w*(i,j) = w (i,j)+r (r is random positive number)
4. end for each
5. update d*
6. for all all betweenness edges e(i,j)
7. w*(i,j)=w(i,j)-r (r is random positive number)
8. end for
9. update D*
10. sort (all partially betweenness edges , descending order) with respect to the number of the shortest paths which pass through this
11. for all partial betweenness edges
12. for all Target Pairs
13. if ((d*(s1,s2)<= original one)>(d*(s1,s2)>original one))
14. w*(i,j)=w(i,j)+t
15. else
16. w*(i,j)=w(i,j)-t
17. end if
18. end for
19. end for
20. for all partial betweenness edges as partial betweenness edge
21. partial betweenness edge weight = graph.get Edge Weight (partial betweenness edge)
22. for all non betweenness edges as nonedge
23. non betweennes edge weight = graph. Get EdgeWeight (nonebetweenness edge)
24. If (non betweenness edge weight less than partial betweenness edge weight)
25. graph.set EdgeWeight (partial betweenness edge, non betweenness edge weight)
26. graph.set EdgeWeight (non betweenness edge, partial-betweenness edge weight)
27. break
28. end for
29. end for
30. update D*

Weight perturbation algorithm mostly focuses on the partial betweenness edges and the weight of the partial betweenness edge is increased or decreased by t.

A partial betweenness edge can be increased by t if t satisfies the following condition:

t < d^s1, s2 - ds1, s2

For all ps1, s2 such that ei, j belongs to ps1, s2[21].

where d^s1, s2 represents the conditional shortest path length in graph G^ = {V,E-{(i,j), (j,i)},W-{wi, j, wj, i}}. Figure 4 shows the increment of an partial betweenness edge. The length of the conditional path length of course is greater than path length.

If t satisfies above condition than perturbed paths will not be changed only perturbed path lengths will be greater than original path length.

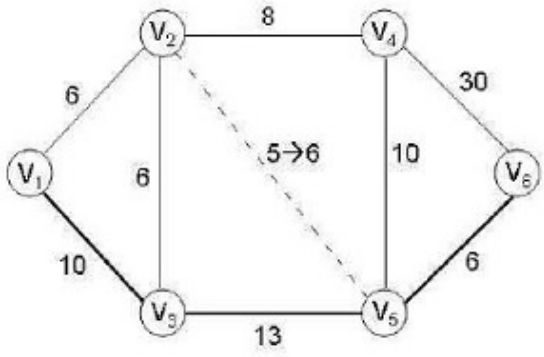

Figure 4. Increasing weight of partial betweenness edge e (2, 5).

Among the three targeted pairs only two shortest paths pass through e2, 5 so in this only two constraints are taken to increase weight w*2,5 by t while p4, 6 does not pass through it.

The two constraints are as follows:

t < d^1,6 - d1,6

t < d^3,6 – d3,6

where d^1, 6 is 29 and d^1, 6 is 17, d^3, 6 is 19 and d3, 6 is 17. Now after solving above inequalities we saw that t should be less than 2 so we select largest integer

as 1. So partial edge weight w2, 5 is increased by 1 and it becomes 6.

A partial betweenness edge can be decreased if it satisfies the following condition:

t < ds1, i + wi, j + ds2, j – ds1, s2

For all paths s1 to s2 such that edge not belongs to paths between s1 and s2. If t satisfies the above criteria then perturbed paths will be same but path length will be decreased.

The path that joins ps1, i , wi, j, pj, s2 represents the conditional shortest path vs1 and vs2.

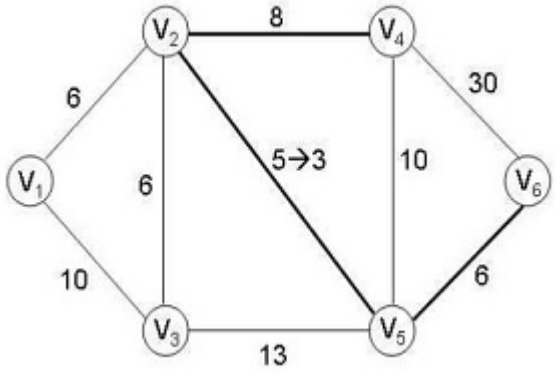

Figure 5. Decreasing weight of partial betweenness edge e (2, 5).

In the Figure 5 the pairs p1, 6 and p3, 6 does not pass through edge going from V1 to V6 and V3 to V6 but shortest path length pass through that edge and decreasing the weight of e2, 5 will not affect the paths p1, 6 and p3, 6. We have to do something with p4, 6 and there is only one way to decrease weight by t defined below:

d4, 2 + (w2, 5-t)+d5, 6> d4, 6 =>t < d4, 2 +w2, 5+d5, 6 – d4, 6

in which d4, 2 is 8, d5, 6 is 6 and d4, 6 is 16. Once the inequality is solved we observe that t <3 and we select as 2 (integer).

So w2, 5 will be reduced by 2 and w*2, 5 will be 3.

After applying perturbation in Graph, all edges are classified into partial, non and all betweenness edges. In greedy algorithm all and non betweenness edge weight is decreased and increased based on random criteria which same follows on Weight Perturbation Shortest Path algorithm but the main difference between is that in greedy perturbation they decrease or increase the partial non between edge by parameter t while in Weight Perturbation Shortest Path algorithm after increment or decrementing the partial edge weight, replacement of partial betweenness edge weight is done by comparing weight of partial and non betweenness, if weight (partial betweenness) is less than weight (non betweenness) it will not be swapped and if weight (partial betweenness) is greater than weight (non betweenness) it will be swapped.

6. Results and Discussions

R-Mat is a synthetic data generator and it can generate graphs with power law vertex degree and also having the small world characteristic, both these are most important properties for many real world social networks. In this paper, experiments are performed on the synthetic datasets. So synthetic dataset generated by R-MAT has been used. So synthetic data is generated consisting of 1000 nodes, 1000 edges and maximum weight of the edge is 30 and minimum weight of the edge is 10.

6.1 Target Pairs

In graph G = {V, E, W} let H be the number of selected targeted pairs Figure 6 need to preserved, the target pairs are selected randomly, but the pairs should have path associate with it.

Start Vertex	End Vertex
17	914
1	991
136	722
578	81
126	73
636	273
703	2
800	19
142	54
137	720
594	105
766	32
258	27
823	938
297	464
800	48
195	453
86	540

Figure 6. Targeted pairs of graph G.

6.2 Performance Metrics

The performance of greedy perturbation, weight perturbation algorithm and its percentage of privacy preservation are measured by plotting the edge weight and shortest

path length. In each result below, the x-axis is the difference between the original ones and the corresponding perturbed ones and the y-axis denotes the percentage of either perturbed weights or perturbed lengths which fall within the x-axis difference to original ones. In each figure, there are two lines, a dashed line and a solid line. The dashed line represents the perturbed shortest path lengths and the solid line denotes the perturbed edge weights.

6.2.1 Edge Weight

For all partially visited edges in target pairs, the difference between original edges and perturbed edges are identified and the percentage is plotted.

6.2.2 Shortest Path Length

For all shortest path length in target pairs, the difference between original shortest path cost and perturbed shortest path cost are identified and the percentage is plotted.

6.2.3 Correlation Coefficient

Correlation is a measure that tells how two variables are related together. A positive value indicates the extent of two variables either increases or decreases together and negative value indicated that if one variable value increases the other variable value decreases. So correlation is metric used here to compare the greedy and enhanced perturbation technique. The correlation value always lies between 1 and -1. It is a normalized measurement of how the percentage of edge weight and shortest path length are linearly related.

Target Pairs	Original Cost	Perturbed Cost	Perturbed Cost	Perturbed Cost	Perturbed Cost
{17:914}	73	67	52	70	61
{1:991}	59	51	61	69	66
{136:722}	77	90	122	146	107
{578:81}	54	55	33	58	56
{126:73}	45	45	61	49	58
{636:273}	69	72	69	57	65
{703:2}	68	59	72	72	70
{800:19}	48	19	68	41	51
{142:54}	49	42	43	67	73
{137:720}	63	69	63	56	61
{594:105}	23	29	55	28	37
{766:32}	55	37	51	55	52
{258:27}	91	73	88	86	94
{823:938}	73	88	75	86	86
{297:464}	56	67	53	46	57
{800:48}	59	57	68	73	60
{195:453}	57	69	58	47	61
{86:540}	69	81	71	83	70

Figure 7. Comparison of original and perturbed shortest path costs in various iterations using greedy perturbation of targeted pairs on synthetic dataset.

Target Pairs	Perturbed Cost (0)	Enhanced Cost (0)	Perturbed Cost (1)	Enhanced Cost (1)	Perturbed Cost (2)	Enhanced Cost (2)	Perturbed Cost (3)	Enhanced Cost (3)
{17:914}	67	51	52	44	70	65	61	58
{1:991}	51	26	61	57	69	65	66	62
{136:722}	90	82	122	113	146	131	107	102
{578:81}	55	46	33	31	58	55	56	52
{126:73}	45	41	61	56	49	47	58	55
{636:273}	72	67	69	66	57	53	65	61
{703:2}	59	57	72	61	72	67	70	65
{800:19}	19	18	68	65	41	39	51	43
{142:54}	42	35	43	40	67	63	73	71
{137:720}	69	65	63	61	56	45	61	56
{594:105}	29	22	55	41	28	27	37	36
{766:32}	37	34	51	49	55	52	52	48
{258:27}	73	67	88	83	86	83	94	91
{823:938}	88	78	75	71	86	68	86	81
{297:464}	67	62	53	48	46	40	57	47
{800:48}	57	51	68	63	73	70	60	56
{195:453}	69	56	58	54	47	44	61	58
{86:540}	81	74	71	67	83	76	70	65

Figure 8. Comparison of greedy perturbation and Weight Perturbation Shortest Path algorithm path costs of targeted pairs on synthetic datasets.

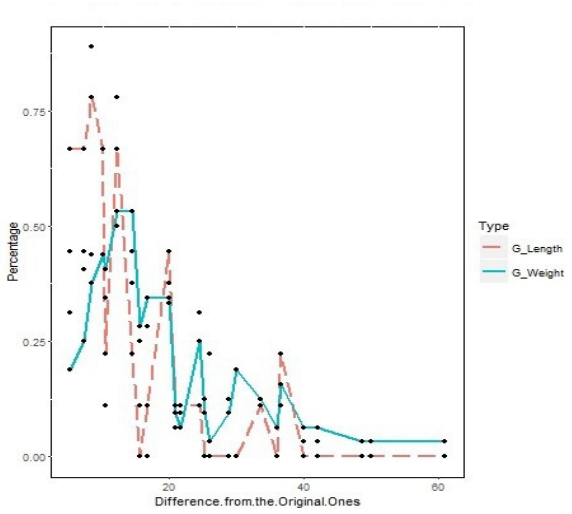

Figure 9. Greedy perturbation 50% targeted pairs being preserved.

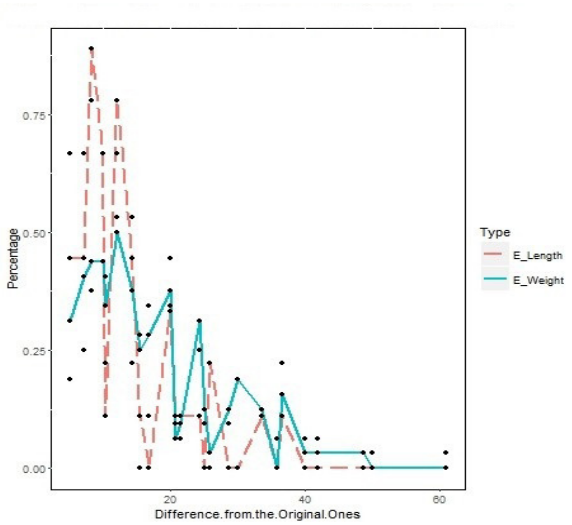

Figure 10. Weight Perturbation Shortest Path 50% targeted pairs preserved.

In greedy perturbation experiment, weights of partially betweenness edges are only taken in synthetic dataset.

The Figure 7 is the comparison of original and perturbation of the shortest paths. The graph is plotted for various iterations showing the difference of values. The Figure 8 is a comparison of shortest path of targeted pairs, so a part of the graph is calculated for the path cost. The Figure 9 indicates results that are obtained after applying greedy perturbation algorithm in which 50% targeted pairs being preserved. G-Length represents the difference between the original and perturbed shortest paths and G-Weight represents the difference between the original and the perturbed partial edge weight. The Figure 10 indicates the results obtained after applying weight perturbation, shortest path algorithm in which 50% targeted pairs being preserved. E-Length represents the difference between the original and perturbed shortest paths and E-Weight represents the difference between the original and the perturbed partial edge weight.

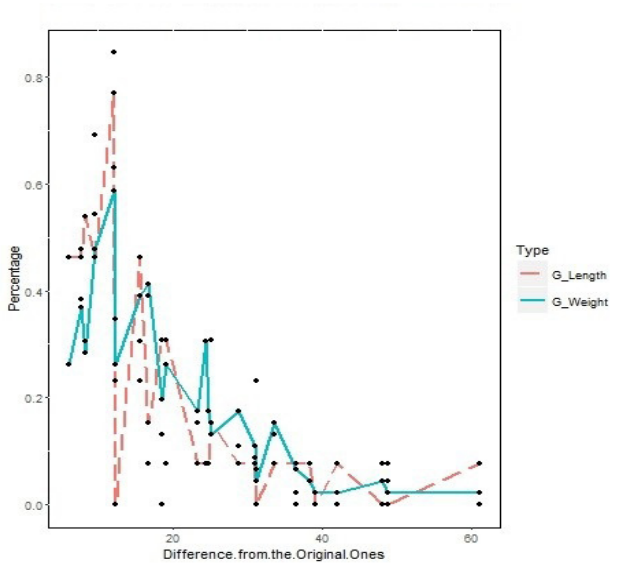

Figure 11. Greedy perturbation 75% targeted pairs being preserved.

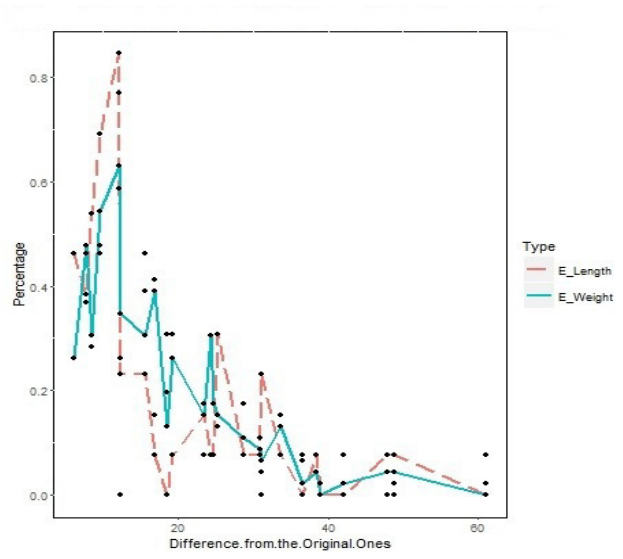

Figure 12. Weight Perturbation Shortest Path 50% targeted pairs preserved.

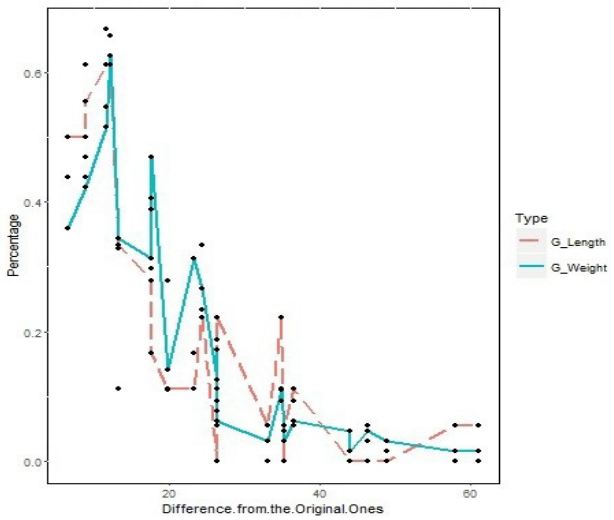

Figure 13. Greedy perturbation 100% targeted pairs being preserved.

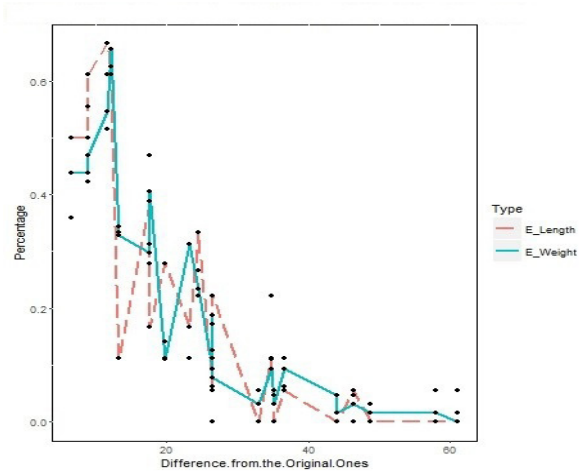

Figure 14. Weight Perturbation Shortest Path 100% targeted pairs preserved.

The Figures 11 to 14 indicates the results obtained after applying greedy and Weight Perturbation Shortest

Path algorithm in which 75%, 100% percent pairs preserved.

For 25% targeted pairs, the correlation coefficient for greedy perturbation is 0.4684875 i.e. 47% and Weight Perturbation Shortest Path algorithm is 0.5696873 i.e. 57%. For 50% targeted pairs, the correlation coefficient for greedy perturbation is 0.6538205 and Weight Perturbation Shortest Path algorithm is 0.7983018. For 75% targeted pairs, the correlation coefficient for greedy perturbation is 0.7969293 and Weight Perturbation Shortest Path algorithm is 0.7973443. For 100% targeted pairs, the correlation coefficient for greedy perturbation is 0.8641661 and Weight Perturbation Shortest Path algorithm is 0.8888129.

7. Conclusions

While taking into account the different privacy issues in social networks, the connections among nodes in the social network data play a very important role such as company transactions between company and agent networks or company's relationship and these connections are very sensitive and crucial in nature. This work addresses a balance between protection of sensitive weights of network links (edges) and some global structure utilities such as the shortest paths and the corresponding shortest path lengths. The paper compares the greedy perturbation technique with the Weight Perturbation Shortest Path Technique.

The experiments results fulfill expectation of our mathematical analysis. The experimental results demonstrate that the two perturbation strategies maintain the closeness of the length between the original and Perturbed Social Network. Future research work will be there to preserve both nodes as well as edge privacy so that social network media network can be more preserved.

8. References

1. Vedanayaki M. A study of data mining and Social Network Analysis. Indian Journal of Science and Technology. 2014 Nov; 7(S7):1–3.
2. Mittal P, Garg S, Yadav S. Social Network Analysis using interest mining: A critical review. Indian Journal of Science and Technology. 2016 Apr; 9(16):1–8.
3. Zhou B, Pei J, Luk W. A brief survey on anonymization techniques for privacy preserving publishing of Social Network Data. Association for Computing Machinery SIGKDD Explorations Newsletter. 2008; 10(2):12–22.
4. Rajper S, Shaikh NA, Shaikh ZA, Mallah GA. Automatic detection of learning styles on learning management systems using data mining technique. Indian Journal of Science and Technology. 2016 Apr; 9(15):1–5.
5. Singh A, Bansal D, Sofat S. Privacy preserving techniques in Social Networks Data Publishing - a Review. IJCA. 2014; 87(15):1–6.
6. Hariharan R, Mahesh C, Prasenna P, Kumar RV. Enhancing privacy preservation in data mining using cluster based greedy method in hierarchical approach. Indian Journal of Science and Technology. 2016 Jan; 9(3):1–8.
7. Masoumzaden A, Joshi J. Preserving structural properties in edge-perturbing anonymization techniques for Social Networks. IEEE Transactions on Dependable and Secure Computing. 2012; 9(6):877–89.
8. Nandi G, Das A. A survey on using data mining techniques for Social Network analysis. International Journal of Computer Science Issues. 2013; 10(6):1–25.
9. Hay M, Miklau G, Jensen D, Weis P, Srivastav S. Anonymizing social networks. Tech. Rep; MIT Amherst, MA. 2007.
10. Zheleva E, Getoor L. Preserving the privacy of sensitive relationships in graph data. Proceedings of the First ACM SIGKDD International Workshop on Privacy, Security and Trusting KDD; San Jose, California. 2007. p. 153–71.
11. Zhang L. Privacy preservation in social graphs. [Doctoral Dissertation]. Association for Computing Machinery. 2012. p. 1–118.
12. Adedoyin-Olowe M, Gaber M, Stahl F. A survey of data mining techniques for Social Network analysis. Journal of Data Mining and Digital Humanities, JDMDH-18; 2014. p. 1–25.
13. Liu L, Wang J, Lin Z, Zhang J. Wavelet-based data distortion for privacy-preserving collaborative analysis. Tech. Rep. University of Kentucky, Lexington, KY; 2007. p. 1–12.
14. Mukherjee S, Chen Z, Gangopadhyay A. A privacy preserving technique for Euclidean distance-based mining algorithms using fourier related transforms. The VLDB Journal. 2006; 15(4):293–315.
15. Xu S, Zhang J, Han D, Wang J. Data distortion for privacy protection in a terrorist analysis system. IEEE International Conference on Intelligence and Security Informatics; Atlanta, GA. 2005; 3495:459–64.
16. Rajalakshmi V, Mala GSA. Anonymization by data relocation using sub-clustering for privacy preserving data mining. Indian Journal of Science and Technology. 2014 Jan; 7(7):1–6.
17. Sweeney L. Guaranteeing anonymity when sharing medical data, the Data Fly system. Journal of the American Medical Informatics Association; 1997. p. 51–5.
18. Zhou B, Pei J. Preserving privacy in Social Networks against neighborhood attacks. Proceedings of the 24th

International Conference on Data Engineering (ICDE'08); Cancun, Mexico. 2008. p. 506–15.
19. Liu L, Wang J, Zhang J. Privacy preservation in Social Networks against sensitive edge disclosure. 2009. p. 1–12.
20. Liu L, Lian L. Privacy preserving data mining for numerical matrices, Social Networks and big data. [Theses and dissertations--Computer Science]. Paper. 2015: 31:1–164.
21. Liu K, Das K, Kargupta H. Privacy-preserving data analysis on graphs and Social Networks. Proficiency Labs; 2008. p. 1–22.