Privacy Preservation in Social Network Using K-Nearest Perturbation Technique

Prof. Sharath Kumar
Computer Science and Engineering
VIT University
Chennai, India

Dr. N Maheswari
Computer Science and Engineering
VIT University
Chennai, India

Aditya Khobragade
Computer Science and Engineering
VIT University
Chennai, India

Prabakaran A
Computer Science and Engineering
VIT University
Chennai, India

***Abstract*—Social networks has a vast growth in today's world and due to this vast growth of social networks, privacy preservation for social network data has become a major issue in networking society. Now-a-days in social network, data is growing so rapidly day by day that it is collected and stored in gigabyte, terabyte or more than that. This huge data generated in social network contains information which is both sensitive and non-sensitive. So the sensitive data in network contain personal information of individuals or confidential files of any organization/company. Leakage of this information is threat to an individual privacy. In social network, a privacy framework play an important role that the data of an individual can carefully process in order to preserve his or her personal information and guarantee information functionality within an acceptable boundary. The researcher proposed techniques to preserve privacy in social network. In this paper, a technique called k-nearest perturbation approach have been applied to social network data to preserve not only privacy but also secure link of social network data in a precise way.**

***Keywords—Anonymization, Social Network, Privacy, Attacks, Graph Technique*.**

I. Introduction

Due to rapidly increasing popularity of social network, large number of people prefer to social networks. Social network are used for building social relationship among people that share common interest, thought, or real life connections. *S*ocial network provide services that facilitate the development of online social networks by connecting an individual profile with those of other individuals and group or organization. Social network interaction of individuals happen over an Internet such mails, messages, transaction, etc. This produced large amount of data that is collected and stored by the social network providers. Produced information is represented has social network data. These data includes information about social media network, friendship, kinship, disease information, transaction between organization/company and relationships among people.

Social networks, is a huge network that contain many social relationships. This social network can be represented as special graph structure connected through vertices and edges. Where vertices/nodes act as individual social actors in a network which is represented as Vertices model in the graph. And relationships/link between each social actors/entities is represented as Edges model.

The nodes are abstract demonstrations of both individuals and organization that are connected by one or more features. The edges or link, used to represent relationships or connections between these nodes. Social network analysis is used in various fields such as organization, biology, communication, economics, and geography . Scientists, advertisers and researchers may require this data for information and security purpose. To analyze data for research, the data should be shared and made public. But making such data public may leak his or her personal information. For example, social networking sites like Facebook, LinkedIn, Instagram and Twitter have thousands of users connected to it. These social networking sites contain sensitive and personal information about individuals. Thus, sharing this information leaks personal privacy. The term privacy says" its person right to elect which information about him should be shared or made communal to others individual and under what conditions". So, our major challenge is to preserving privacy in a secure way and maximizing utility of social network analysis

II. Related work

For securing privacy many techniques were proposed by researchers. The technique like k-anonymity and l-diversity was built for privacy preserving on social network data. But this technique cannot be applied directly for privacy preservation. Anonymization in social network data is difficult task. Because, attacks directly come from classifying individuals from sensitive attributes such as name and SSN. To identify individual's target, information such as neighborhood graphs can be used in social network. To perform such technique on original data, it has to anonymize the records without affecting other records. Thus in social networks, inserting edges or nodes are difficult task that affects other nodes in the graph [2], [7], [14].

To achieve privacy on original data, alteration of sensitive information has to done on data using perturbation techniques. Additive and Matrix Multiplicative is a perturbation approach that randomly adds noisy values on selected attributes to achieve privacy preservation for those attributes. But these techniques fails, if adversary already guess the selected attribute on which perturbation approach is performing [5], [7].

In social network, entities and edges are determined as important functionality. This approach was applied to social network analysis. Using graph, social network data can easily represented in hierarchical structure. Where nodes are connected with edges having weight assign to it. Here weight can be perturbed to preserve privacy in social graph [8], [10], [16].

A. Privacy attacks on Social Network.

In Social Network, if an adversary has good background knowledge about shared social network data, then it may pose some threat for individual privacy. Data such as mails, messages or telephone communication comes from setting of many of the sources of social network. According to privacy concerned on such type of data, user's expectations are very high. When such social network data is made open or shared, then it require more privacy to protect individual data by simply replacing the sensitive attributes such as name and SSN of people by some extraordinary identifiers. The privacy attacks in social networks can be categorized into three types:

- Identity Leak Exposed **-** When an individual behind a record is exposed we say identity is leaked. This type of threat leads to uncover the information of a user and relationship or connections he/she have with other individuals in the network.
- Sensitive Link Exposed- When the connections/relationship between two individuals are revealed we say sensitive link disclosure occurred. Social activities generate this type of information when social media services are utilized by users.
- Sensitive Attribute Exposed– When an attacker obtains the information of a sensitive and confidential user attribute such threat refers to sensitive attribute disclosure. An entity and link relationship is linked with sensitive attributes.

III. K-NEAREST PERTURBATION TECHNIQUE

A. Edge Perturbation Technique

In Edge perturbation, edges are considered as sensitive attribute which represent connection or relationship between two entities. In graph, edges are associated with weight that imitates connection between two entities which carries the data. Due to the rapid growth of social networks, safety and privacy concern arising more in social network such as Facebook, LinkedIn and Instagram. When this network data made open or being shared which bring a high risk of leakage personal information from social network analysis. In addition to the techniques used for social network, a business transaction warehouse is essentially a social network were edges are assigned with particular weights that are considered to be stable. In perturbation, noisy data is added to original record, so that the original data values must be preserved and it cannot be known from the distorted data. Thus, we can recover distribution of the original data but not original data.

For example, in real world suppose there are two companies want to do business deal with each other. Where nodes are represented as company & agents and edges are represented as connection/relationship between nodes. Company X wants to buy some goods or services, from Company Y so deal cannot be made directly by accessing each other due to trade off. Company X needs to choose some trade intermediate agents/suppliers that is Agent1, Agent2, and Agent3. Who has the shortest path of transaction cost between themselves and Company Y [7], [16]. If the weights of edges are perturbed which carrying transaction cost between Company X and Y. Then, the shortest paths well preserved by perturbed the corresponding length, Company X may be able to make good decision based on this privacy-preserving social network without having confidential details about the relationship between agents and Company Y [7].

Perturbation is most important concept for social network. Perturbation is used to modify the attribute value by new value. Consider the graph consisting of six vertices and nine edges as shown in figure 1[7]. Where graph G in the original graph having the edges and vertices denoted as G (V, E) and while in the perturbed graph G* having the edges and vertices denoted as G* (V*, E*). In figure 1[16], vertices V are given as V1, V2, V3, V4, V5, and V6. And the edges E between vertices are assigned with the weight.

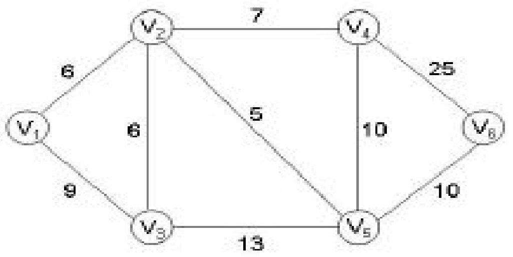


Fig. 1. Social Network Graph G

After applying the concept of perturbation on graph G weight of the edges are modified. Figure 2[16] shows the perturbed graph G*.In perturbed graph, 90% of the edge weight changes and only 10% remains as it is. From the perturbed graph G* we clearly see that the edge between the vertex pair (V1, V2), (V1, V3), (V2, V4), (V3, V5), (V4, V5), (V4, V6) and (V5, V6) are modified and vertex pair (V2, V3), (V2, V5) remains unchanged.

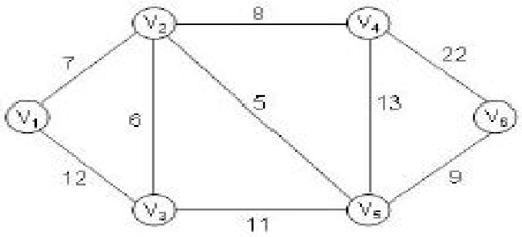


Fig. 2. Perturbed Graph G*

When the network is distributed, information security is preserved by annoying the weights of edges on graph. And thus it recollects the shortest path and length of the path between pairs of nodes is required in the original network. To preserve privacy in social network Greedy Perturbation strategy proposed, which maintain the shortest path and estimate the length of the path between pair of nodes.

B. Greedy Perturbation Technique

The greedy perturbation algorithm focuses on keeping the same shortest path before and after the perturbation. Let starts first describing the greedy perturbation technique. In real world all the data cannot be preserved. Some shortest path between the vertex pairs can be preserved and L denotes that set of pairs that keep preserving [1], [7].

Social network graph can be represented as G consists of E representing the edge list and V containing the number of vertices in a social network and shortest path P and path length is generated from n*n matrix where n is the number of vertices in the graph. In graph G, Wi,j represent the edge weight between vertex pair. P(v1, v2) indicates shortest path between v1 and v2. While D(v1, v2) represents the shortest path length between v1 and v2. After applying the greedy perturbation, W*i,j represent perturbed weight, P*(v1,v2) represent shortest, and D*(v1,v2) represents perturbed path length.

To perturb a graph it must fulfill the following conditions:

- In both original and perturbed graph number of vertices should be equal.
- In both original and perturbed graph number of edges should be equal.
- Maximize the number of perturbed weight such that perturbed weight is not equal to original weight.
- After the perturbation, shortest path should be same in both perturbed graph and original graph.

Edges in graph are classified into different categories:

- Nonbetweenness edge: The edge that does not pass through even single shortest path, then edge is known as nonbetweenness edge.
- All betweenness edge: When all the shortest path pass through that edge, then the edge is known as all betweenness edge.
- Partial between-ness edge: An edge is called partial betweenness edge when it passes through even single shortest path.

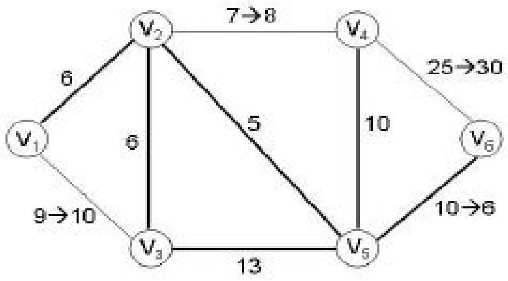


Fig. 3. Three different types of edges

Figure 3[16] shows all the types of edges. Consider three shortest path vertex pairs stored in H={V(1,6), V(4,6), V(3,6)}.Edges {E(1,3), E(2,4), E(4,6), E(3,5)} signify non between-ness edge because no shortest path lengths pass through that edge. Edges E(1,2), E(2,5) , E(2,3) and E(4,5) represents partial between-ness edge because only some shortest path lengths pass through that edges and edge E(5,6) signify all between-ness edge because all shortest path lengths P(1,6), P(4,6) and P(3,6) pass through that edge. Perturbation is performed on all these edges. For non-between-ness edges if weight of an edge is increased by randomly generated number r (W*i,j = Wi,j + r) all shortest path lengths and shortest paths in H will not be changed. For all-between-ness edges if weight of an edge is decreased by random generated 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*(v1,v2) = D(v1,v2) – r). In the social networks most of the edges are partial between-ness edges. So greedy perturbation algorithm mostly focus on partial between-ness edges. Greedy perturbation algorithm increase the partial between-ness 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 between-ness 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 between-ness 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. But greedy perturbation fails to preserves privacy since there are more chances for adversary to know the same shortest path after perturbation because the perturbed shortest path length will less than other shortest path in the graph which make it easier to guess if he/she has good background knowledge about graph.

C. K-Nearest Perturbation Technique

The purpose of this algorithm is to enhance the privacy and to preserve same shortest path after perturbation done on graph. So, the objective of k-nearest perturbation algorithm is to provide privacy by perturbing the graph so that k-nearest shortest paths can be achieved. The proposed approach is to perturbed the edge weights of non-overlapping edges closed to shortest path length so that they all path possess the same path length. The proposed algorithm first finds all the shortest path and also minimize or maximize the edge weights of non-overlapping edges between the other shortest path and the shortest paths. In given graph *G*, aim is to provide k level privacy between a source vertex vi and a destination vertex vj, without adding or deleting any vertices or edges such that there exists *k* shortest paths between the given pair of vertices by perturbing the graph and to maintain the same cost for all k-pairs.

1) Algorithm

```
• Input: W is the adjacency weight matrix of an original graph G; the set of selected shortest paths to be preserved.
• Output: Perturbed weighted adjacency matrix W*.
• Foreach AllTargetPairs as TargetPair
• K-NearShortestPaths=KShortestPaths(TargetPair,K) //K=3
• PathCost=BellmanFord.ShortestPathCost (Graph,TargetPair)
• Paths=BellmanFord.ShortestPaths (Graph,TargetPair)
• Foreach K-NearShortestPaths as KNearShortestPath
• KthPathCost=BellmanFord
• ShortestPathCost(Graph,KNearShortestPath)
• KthPaths=BellmanFord.ShortestPaths(Graph,KNearShortestPath)
• List PathsToModify. Foreach KthPaths as KthPath
• Int count=0
• Foreach Paths as Path
• If(KthPath==Path)
• count = 1
• End If
• End For
• If count == 0
• PathsToModify.add(KthPath)
• End If
• End For
• Double PathsToModify_Weight_Sum
• Foreach PathsToModify as PathToModify
• PathsToModify_Weight_Sum+=Graph.getEdgeWeight(PathToModify)
• End For
• DoubleCost=PathCost-(KthPathCost PathsToModify_Weight_Sum)
• If(Cost>0.0)
• Double Increase = ( Cost / PathToModify.Size)
• Foreach PathsToModify as PathToModify
• Graph.setEdgeWeight(PathToModify)=Graph.getEdgeWeight(PathToModify)+Increase
• End For
• End If
• Else
• Double Decrease=Cost/PathToModify.Size
• Foreach PathsToModify as PathToModify
• Graph.setEdgeWeight(PathToModify)=Graph.getEdgeWeight(PathToModify)-Decrease.
• Cost = Cost – Decrease
• End For
• End Else
• End For
• End For
```

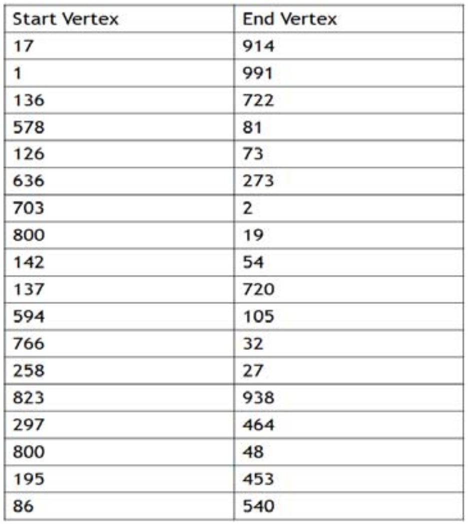
After applying this algorithm on graph, obtain k-nearest perturbation, the shortest path length is closed to the targeted shortest path length. That is, the cost of k-nearest shortest path will be same as the original shortest path. After publishing the perturbed graph on network, the k-nearest shortest path in the graph provide more uncertainty for adversary to guess the original shortest path. So, the privacy of graph is enhanced and thus preserve the same shortest path in the graph.

V. RESULT AND DICUSSION

Here the experiment analysis is done on synthetic dataset. To generate synthetic dataset for experiment R-MAT tool used. It generates the undirected graph with edges and vertex degree, and also having characteristic, which are important properties for social network. In this paper, the experiment is done on synthetic dataset generated by R-MAT. 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.

A. Target Pairs

In graph G = V,E,W let H be the number of selected target pairs 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

Fig. 4. Target Pairs

B. Performance Metrics

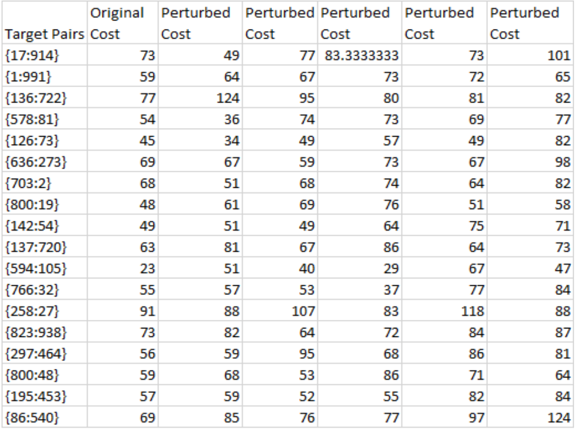
The Performance is measured by plotting the Edge Weight and Shortest Path Length for greedy perturbation, enhanced K-Nearest Perturbation algorithm and its percentage of privacy preservation. In each result below, the x-axis is the difference between the original ones and the corresponding perturbed 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.

C. Shortest Path

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.

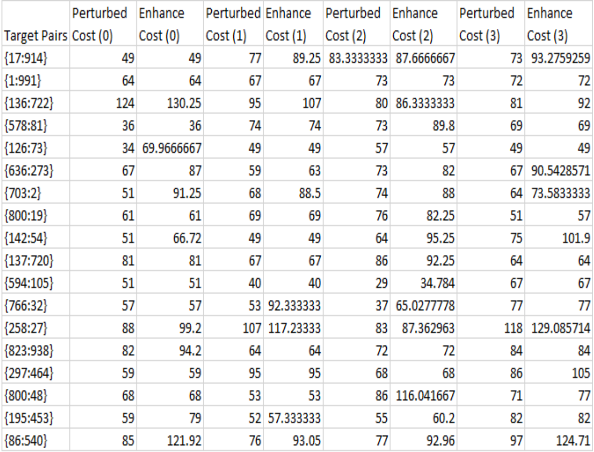
D. Correlation Measure

Correlation is a measure which tells how two variables are closely related. 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. 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. So correlation is metric used here to compare the Greedy Perturbation method and Enhanced K-Nearest Perturbation method.



Target Pairs	Original Cost	Perturbed Cost	Perturbed Cost	Perturbed Cost	Perturbed Cost	Perturbed Cost
{17:914}	73	49	77	83.3333333	73	101
{1:991}	59	64	67	73	72	65
{136:722}	77	124	95	80	81	82
{578:81}	54	36	74	73	69	77
{126:73}	45	34	49	57	49	82
{636:273}	69	67	59	73	67	98
{703:2}	68	51	68	74	64	82
{800:19}	48	61	69	76	51	58
{142:54}	49	51	49	64	75	71
{137:720}	63	81	67	86	64	73
{594:105}	23	51	40	29	67	47
{766:32}	55	57	53	37	77	84
{258:27}	91	88	107	83	118	88
{823:938}	73	82	64	72	84	87
{297:464}	56	59	95	68	86	81
{800:48}	59	68	53	86	71	64
{195:453}	57	59	52	55	82	84
{86:540}	69	85	76	77	97	124

Fig. 5. Comparison of original and perturbed shortest path costs using Greedy perturbation of targeted pairs on synthetic dataset



Target Pairs	Perturbed Cost (0)	Enhance Cost (0)	Perturbed Cost (1)	Enhance Cost (1)	Perturbed Cost (2)	Enhance Cost (2)	Perturbed Cost (3)	Enhance Cost (3)
{17:914}	49	49	77	89.25	83.3333333	87.6666667	73	93.2759259
{1:991}	64	64	67	67	73	73	72	72
{136:722}	124	130.25	95	107	80	86.3333333	81	92
{578:81}	36	36	74	74	73	89.8	69	69
{126:73}	34	69.9666667	49	49	57	57	49	49
{636:273}	67	87	59	63	73	82	67	90.5428571
{703:2}	51	91.25	68	88.5	74	88	64	73.5833333
{800:19}	61	61	69	69	76	82.25	51	57
{142:54}	51	66.72	49	49	64	95.25	75	101.9
{137:720}	81	81	67	67	86	92.25	64	64
{594:105}	51	51	40	40	29	34.784	67	67
{766:32}	57	57	53	92.333333	37	65.0277778	77	77
{258:27}	88	99.2	107	117.23333	83	87.362963	118	129.085714
{823:938}	82	94.2	64	64	72	72	84	84
{297:464}	59	59	95	95	68	68	86	105
{800:48}	68	68	53	53	86	116.041667	71	77
{195:453}	59	79	52	57.333333	55	60.2	82	82
{86:540}	85	121.92	76	93.05	77	92.96	97	124.71

Fig. 6. Comparison of greedy perturbation and K-nearest perturbation algorithm path costs of targeted pairs on synthetic datasets

Figure 7, 9, 11 indicates the results that are obtained after applying Greedy Perturbation algorithm which consists of two edge list. Red edge list indicates Greedy length and green one indicates weight. G-length represents difference ratio between the original and perturbed shortest paths and G-Weight represents the difference between the original and the perturbed edge weight. Figure 8, 10, 12 indicates the results obtained after applying k-nearest shortest path algorithm which consists of two edges list. That is, red edge list indicates the enhanced length and green edge list indicates the enhanced weight. E-Length represents the difference between the original and perturbed shortest paths and E-Weight represents the difference between the original and the perturbed edge weight. Figure 7 and 8 indicates the results obtained after applying greedy and k-nearest perturbation algorithm on 25% percent targeted pairs preserved. Figures 9, 10,11,12 indicates the results obtained after applying greedy and k-nearest perturbation algorithm in which 75%,100% percent pairs preserved.

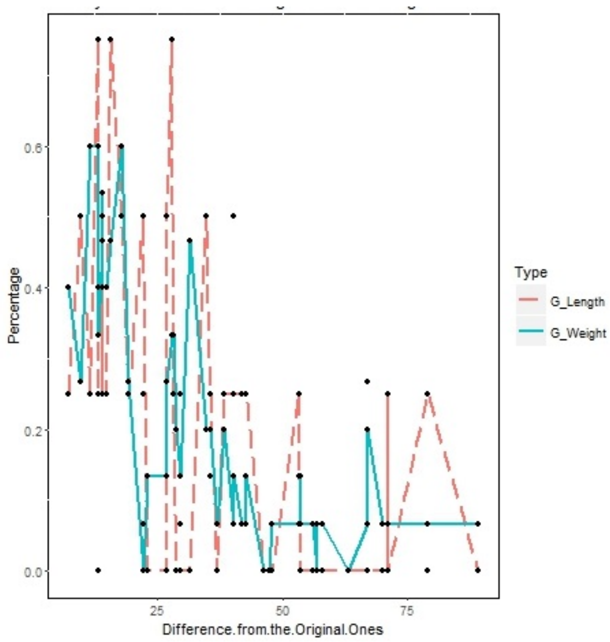


Fig. 7. Greedy Perturbation 25% targeted pairs being preserved

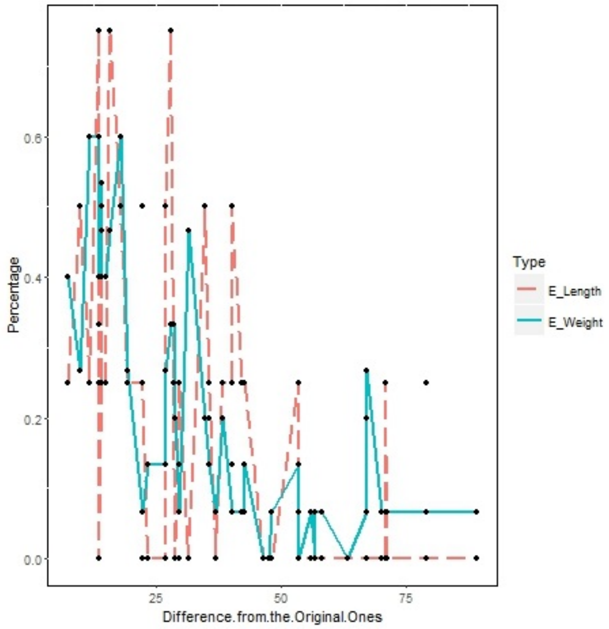


Fig. 8. K-Nearest Perturbation 25% targeted pairs being preserved.

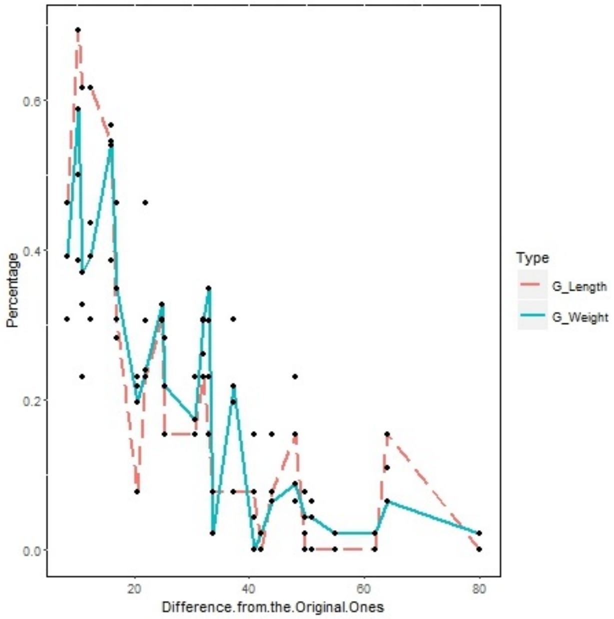


Fig. 9. Greedy Perturbation 75% targeted pairs being preserved

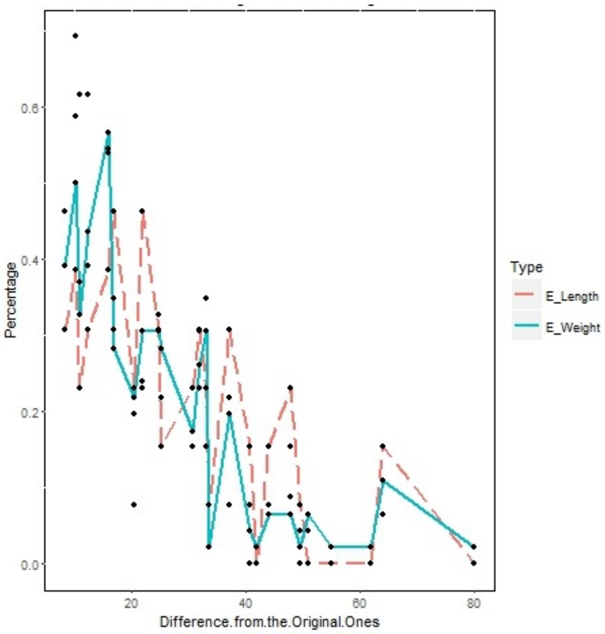


Fig. 10. K-Nearest Perturbation 75% targeted pairs being preserved

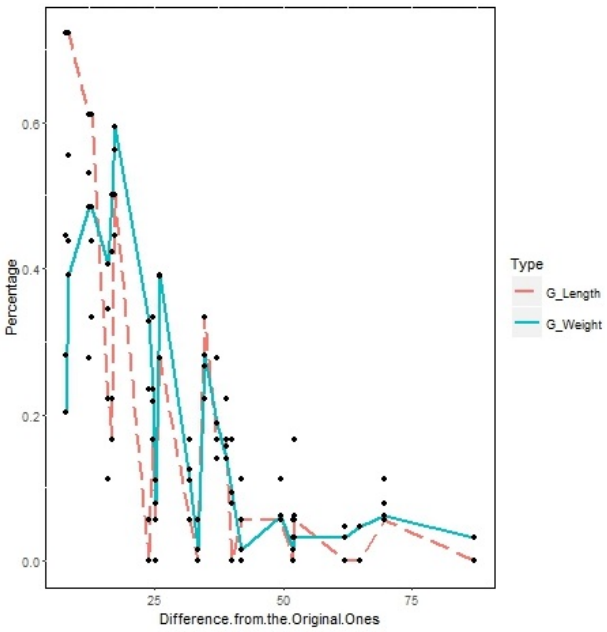


Fig. 11. Greedy Perturbation 100% targeted pairs being preserved

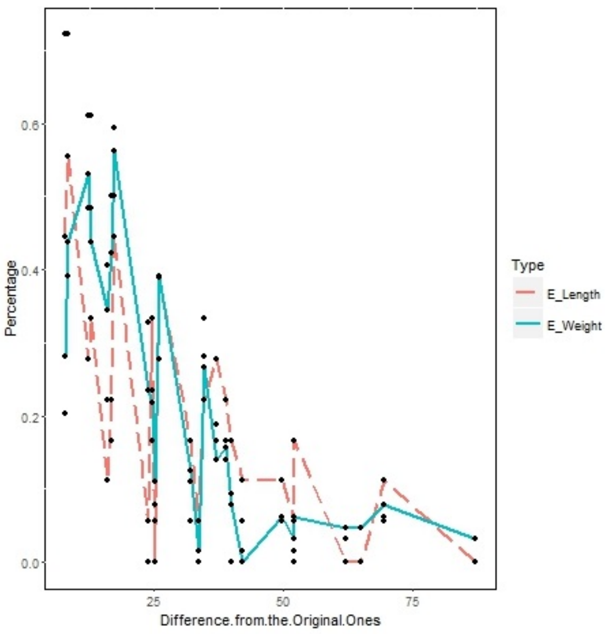


Fig. 12. K-Nearest Perturbation 100% targeted pairs being preserved

For 25% targeted pairs, the correlation measure for greedy perturbation is 0.5717007 i.e. 58% and k-nearest perturbation method is 0.5930333 i.e. 60%. For 75% targeted pairs, the correlation measure for greedy perturbation is 0.8045618 i.e. 80% and k-nearest perturbation method is 0.8916386 i.e. 90%. For 100 % targeted pairs the correlation measure for greedy perturbation is 0.7060303 i.e. 70% and k-nearest perturbation method is 0.743088 i.e. 74%. From above analysis, results conclude k-nearest perturbation algorithm preserved more privacy than the greedy perturbation algorithm.

VI. CONCLUSION

In this paper, the research concludes that edges are the important characteristic which tells connection between two entities/nodes in social network. Such as a connection represents the transaction between two companies and relationship between two people. In this paper, our algorithm focused on perturbing the edges so as to preserve the relationship between nodes and obtain more privacy by preserving same shortest path within the published network. The paper compares the Greedy Perturbation method with the enhanced K-Nearest Perturbation method. The experiments results fulfill expectation of our mathematical analysis. Future research work relate on preserving both the nodes as well as edges privacy so that social network media network can be preserved more privacy.

REFERENCES

[1] LMayank Singh Shishodia, Sumeet Jain, B.K.Tripathy, "GASNA-Greedy Algorithm for Social Network Anonymization," IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, 2013.

[2] Charu C. Aggarwal, IBM T. J. Watson Research Center, "General Survey of Privacy Preserving Data Mining Models and Algorithm," Springer.

[3] M. Hay, G. Miklau, D. Jensen, P. Weis,S. Srivastav, "Anonymizing social networks," MIT, Amherst, MA, Tech. Rep.07-19, 2007.

[4] Bin Zhou ,Jian Pei ,Wo-Shun Luk," A Brief Survey On Anonymization techniques for Privacy Preserving Publishing of Social Network Data," Association for Computing Machinery SIGKDD Explorations Newsletter, Vol. 10, Issue 2,Dec 2008.

[5] Anirban Chakraborty, Annappa B.," A Perturbation Based Approach for Privacy Preserving Publication of Social Network Graph," 2014.

[6] Amardeep Singh ,Divya Bansal , Sanjeev Sofat," Privacy Preserving techniques in Social Networks Data Publishing- A Review ," IJCA , Vol. 87 ,No .15, February 2014.

[7] Lian Liu Lexington, Dr. Jun Zhang," Privacy Preserving Data Mining for Numerical Matrices, Social Network, and Big Data," 2015.

[8] Lokesh Patel, Ravindra Gupta," A Survey of Perturbation Technique For Privacy-Preserving of Data," International Journal of Emerging Technology and Advanced Engineering, Vol.3, Issue 6, June 2013.

[9] E. Zheleva , L. Getoor," 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, pp. 153-171, August 2007.

[10] N.Punitha, R.Amsaveni," Methods and Techniques to protect the privacy information in Privacy Preservation Data Mining," International Journal of Computer Technology Applications, Vol. 2960, 2091-2097.

[11] S. Xu, J. Zhang, D. Han, J. Wang," Data distortion for privacy protection in a terrorist analysis system," IEEE International Conference on Intelligence and Security Informatics, Atlanta, GA, pp. 459-464, 2005.

[12] B. Zhou and J. Pei," Preserving privacy in social networks against neighborhood attacks," Proceedings of the 24th International Conference on Data Engineering (ICDE08), Cancun, Mexico, pp. 506-515, April 2008.

[13] Xintao Wu, Xiaowei Ying, Kun Liu,A," Survey of Algorithms For Privacy Preservation of Graphs and Social Networks, In Managing and Mining Graph Data," Springer Science Business Media, 421-453.

[14] Lijie Zhang, Weining Zhang," Edge Anonymity in Social Network Graphs," International Conference of Computational Science and Engineering, 2009.

[15] X.Wu, X.Ying, K.Liu," A Survey of Privacy Preservation of Graphs and Social Networks, Managing and Mining Graph Data," Springer Science Business Media, 2010.

[16] Lian Liu, Jie Wang , Jinze Liu , Jun Zhang," Privacy Preserving in Social Networks Against Sensitive Edge Disclosure,"2008.