Partial Data, Potential Exposure: Evaluating Privacy Leakage via GNNExplainer on Social Networks

Liang-Jen Huang
National Cheng Kung University
liangjenh@gmail.com

Cheng-Te Li
National Cheng Kung University
chengte@ncku.edu.tw

***Abstract*—This paper investigates potential privacy breaches arising from access to only portions of user data. We propose a novel approach that utilizes explanatory subgraphs generated by GNNExplainer to depict the partial user data accessible through queries. By employing the link prediction technique of Graph Neural Networks, we attempt to deduce further undisclosed user information. Our experimental results show that even when only a portion of user information is exposed, there remains a significant risk of privacy leakage.**

I. Introduction

Although a vast amount of information is shared publicly on today's social networks, sensitive information containing personal privacy remains difficult to access. Under such circumstances, malicious actors may adopt indirect methods to obtain or infer these hard-to-reach messages. For instance, they can exploit publicly available information to conduct queries, which in turn reveal social relationships between certain individuals. Such actions not only infringe on individual privacy rights but also pose the risk of being used for illegitimate purposes.

This research investigates a critical question: Can we reveal more extensive relationships among users when only partial user information is available, specifically the results of certain queries? For this purpose, we employ subgraphs generated by GNNExplainer [1] as a substitute for traditional query results. We consider that if relationships between users can be predicted merely through subgraphs, it indicates a potential risk to privacy protection. Conversely, if these key explanatory subgraphs fail to reveal more friendship relationships, it proves the effectiveness of privacy protection measures.

In this study, we utilize GNNExplainer to generate several explanatory subgraphs that serve as partial user information obtainable through queries. Subsequently, we conduct link prediction to understand the relationships among users in these subgraphs, determining how many of them could be accurately identified by the GNN model, as illustrated in Figure 1. Our experimental results reveal that our model achieved AUC values of 91% and 85% on the Facebook and GitHub datasets, respectively. These findings indicate that even when only partial user data are available, the risk of privacy leakage persists.

II. The Proposed Method

This study is conducted in three stages. First, we train a base GNN model to enable GNNExplainer to generate subgraphs that explain the model's predictive outcomes. Next, we select several nodes and employ GNNExplainer to generate a series of explanatory subgraphs, which are then compiled into a new dataset. Finally, we utilize this newly synthesized dataset for link prediction, aiming to reconstruct the original structure of the social network.

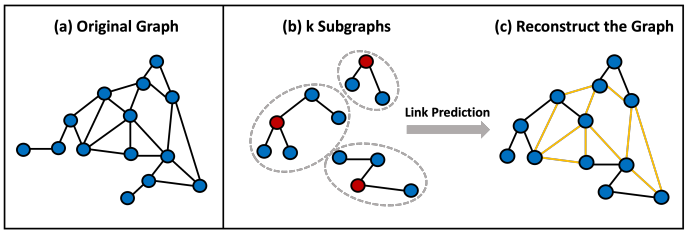

Fig. 1. The main objective of this study. Panel (a) presents the initial network structure. In panel (b), the red dots represent the nodes selected for GNNExplainer to generate explanations (with k = 3), and the dashed gray circles highlight the generated explanatory subgraphs, representing partial user information. Panel (c) displays link predictions to uncover further relationships among users based on these subgraphs, with yellow lines indicating the predicted edges.

Preliminary. GNNExplainer [1] is an interpretable method for GNN-based models that uses a mask-out operation to identify key nodes and edges for model predictions, assembling them into an explanatory subgraph. This method maximizes the mutual information between the model's output and the subgraph, revealing a significant and compact subgraph structure critical for predictions.

Generating Explanatory Subgraphs and New Dataset Synthesis. This phase aims to use GNNExplainer to derive crucial explanatory subgraphs from previous GNN model, representing the partial user information that could be obtained through queries. We randomly select k specific nodes (with $k = 3$ set for this study) and generate k corresponding explanatory subgraphs for these nodes with GNNExplainer. Further, we examine the connectivity between these k explanatory subgraphs. A reselection process is initiated if they lack path connections, as disconnected subgraphs do not align with our objectives. Next, we synthesize a larger graph by combining k subgraphs and preserving all node features. By repeatedly applying this process, we create multiple new synthetic graphs that serve as a new dataset for subsequent link prediction tasks.

Link Prediction Mechanism. In this phase, we aim to

TABLE I
DATA STATISTICS.

Dataset	Facebook	GitHub
#graph	1	1
#nodes	22470	37700
#edges	342004	578006
#features	128	128
#classes	4	2

reconstruct the original graph structure using explanatory subgraphs generated by GNNExplainer. To achieve this, we employ a fundamental link prediction technique. Initially, we learn embeddings for each node in the graph using a GNN model. Then, we predict the presence of edges between pairs of nodes by calculating the dot product of their node embeddings. A higher dot product value indicates a stronger similarity between the nodes, suggesting a higher probability of an edge between them.

III. EXPERIMENTS

Datasets. In the experiment, we used two datasets: Facebook and GitHub [2]. These datasets were employed to conduct preliminary node classification tasks. Detailed statistical information about the datasets is provided in Table I.

Experimental Setup. There are three phases in the experiment: 1) Training the base GNN model for classification; 2) Generating explanatory subgraphs and synthesizing a new dataset, and 3) Performing link prediction to reconstruct the social network structure.

For the base model, we used a three-layer GCN [3] with hidden layer sizes of 64 and 32 for Facebook and a two-layer GCN with a 64-size hidden layer for GitHub. We set the learning rates to 0.001 for Facebook and 0.03 for GitHub, employing ReLU activation and the Adam optimizer to achieve accuracies of 91.10% on Facebook and 86.55% on GitHub after 1000 epochs.

In the subgraph generation phase, GNNExplainer was configured to use a 3-hop neighborhood for Facebook and a 2-hop for GitHub, corresponding to the respective layers of the previous GCN models. Next, we randomly selected three nodes to generate explanatory subgraphs. If these subgraphs were interconnected in the original graph, they were merged into a larger graph. This process was repeated 140 times, resulting in 140 synthetic graphs, with 100 for training, 20 for validation, and 20 for testing. The details of the dataset composition is listed in Table II.

During the link prediction phase, we utilized GCN, GAT [4], and GraphSAGE [5] models on both Facebook and GitHub datasets. These models consist of two layers, with a hidden layer size of 128, to project node representations into a 64-dimensional embedding space for link prediction. The learning rates were set to 0.01 for all models on the Facebook dataset and adjusted to 0.005 for the GitHub dataset. Utilizing the Adam optimizer, we trained the models over 150 epochs on both GitHub and Facebook, with a consistent batch size of 8. Additionally, for each batch during training, we randomly generated several negative samples equal to the actual edges.

TABLE II
SYNTHESIZED DATASET STATISTICS.

	Facebook			GitHub		
	Train	Valid	Test	Train	Valid	Test
#graph	100	20	20	100	20	20
#avg n[1]	5580	4817	5651	11466	8080	13472
#avg e[1]	31216	24839	36241	14161	8795	23144
#avg pe[2]	15608	12421	185555	7081	4398	527030

[1] "#avg n", "#avg e" is the average number of nodes/edges per graph.
[2] "#avg pe" is the average number of edges to predict per graph.

TABLE III
EXPERIMENT RESULT

Dataset	Facebook			Github		
	AUC%	Pr%	Re%	AUC%	Pr%	Re%
GCN	90.48	87.15	35.25	84.97	93.12	8.62
GAT	90.94	89.19	64.16	74.94	90.15	11.88
GraphSAGE	89.52	86.19	34.57	83.51	93.49	8.62

Moreover, we adopted a threshold that yielded the highest F1 score in the validation set, allowing each model to select its optimal threshold for evaluating precision and recall.

Results. In the Facebook dataset, the GCN, GAT, and GraphSAGE models demonstrated outstanding performance, each achieving around 90% AUC. Meanwhile, the GCN model led with an AUC of 84.97% in the GitHub dataset. This suggests that explanatory subgraphs are sufficient to expose further relationships, indicating the risk of privacy leaks from limited queries. Further analysis of precision and recall showed that, in the Facebook dataset, all three models accurately identified over 86% of the predicted friendships, and in the GitHub dataset, this accuracy exceeded 90%. Moreover, the GAT model successfully predicted 64.16% and 11.88% of friendships in the Facebook and GitHub datasets, respectively. The results are listed in Table III.

IV. CONCLUSIONS

This paper introduces a novel approach to evaluating privacy risk leakage using GNN's interpretability methods. By using explanatory subgraphs generated by GNNExplainer to represent traditional query results, we demonstrate that even partial data can lead to significant breaches of relationship privacy. Future work could focus on developing new interpretable techniques to provide sufficient explanations while enhancing user privacy protection.

REFERENCES

[1] Z. Ying, D. Bourgeois, J. You, M. Zitnik, and J. Leskovec, "Gnnexplainer: Generating explanations for graph neural networks," *Advances in neural information processing systems*, vol. 32, 2019.

[2] B. Rozemberczki, C. Allen, and R. Sarkar, "Multi-scale attributed node embedding," *Journal of Complex Networks*, vol. 9, no. 2, p. cnab014, 2021.

[3] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *arXiv preprint arXiv:1609.02907*, 2016.

[4] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," *arXiv preprint arXiv:1710.10903*, 2017.

[5] W. Hamilton, Z. Ying, and J. Leskovec, "Inductive representation learning on large graphs," *Advances in neural information processing systems*, vol. 30, 2017.