Model Extraction Attacks & Defenses

Malachi Davey, Bolin Shen, & Dr. Yushun Dong

FSU | FLORIDA STATE UNIVERSITY

FSU UNDERGRADUATE RESEARCH OPPORTUNITY PROGRAM
CENTER FOR UNDERGRADUATE RESEARCH & ACADEMIC ENGAGEMENT

Abstract:

GNNs (Graph Neural Networks) are deep learning models that are designed to analyze data in the form of graphs, made up of nodes (entities) and edges (relationships). This research tests the security of these models. Are they capable of holding integrity when under attack, and able to defend their sensitive information? These models are attacked when an "attacker" queries them, with the goal of extracting sensitive information on the behavior of the model. This is critical in modern times with the rise of artificial intelligence, large-scale companies have made it a priority to implement AI in their products and want to ensure that their trained models are secure and safe to use by users.

To start, we had to train the baseline GNNs on benchmark graph datasets. We simulated an adversary using GraphMI and PyGIP attack methods and applied defense mechanisms like DropEdge and Feature Masking to prevent privacy leakage. Success of the simultaneous attacks and defenses was based on the AUC and model performance. Results showed that GraphMI successfully reconstructs node features and partial graph structure from trained GNNs at an accuracy significantly higher than random baselines. DropEdge and Feature Masking reduce inversion accuracy and make GraphMI weaker when attacking the GNNs. The results show that baseline GNN models are vulnerable to GraphMI model extraction attacks, as sensitive graph information can be reconstructed above random levels. However, implementing defense mechanisms such as DropEdge, feature masking, and differential privacy noise significantly reduced reconstruction accuracy, thereby improving model integrity under attack.

Introduction:

Graph Neural Networks (GNNs) are widely used for learning on structured data, such as, biological networks, social graphs, and molecules. GNNs achieve strong predictive performance, based on nodes and edges, but at the same time they still are vulnerable to leaking sensitive information about their training data through model inversion attacks.

Graph Model Inversion (GraphMI) attacks breach a trained GNN to reconstruct certain properties of the original model, for example structural patterns and node features. In the artificial intelligence field, this raises heavy concerns about privacy, especially chemistry, healthcare, and social platforms.

Although prior work has demonstrated the effectiveness of GraphMI attacks, there has been little attention to evaluating defenses that reduce data leakage without modifying the attacker or model architecture. In this project, we study whether lightweight training-time defenses can reduce the success of GraphMI attacks.

The ultimate goal is to reproduce GraphMI attacks using the PyGIP framework and evaluate how defenses such as DropEdge, feature masking, and feature noise affect reconstruction performance across benchmark graph datasets.

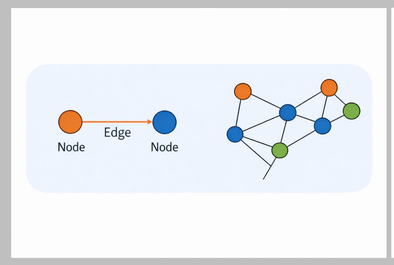

Figure 1.
Nodes and edges in a graph structure.
Nodes represent entities, and edges represent the relationships connecting them within a graph.

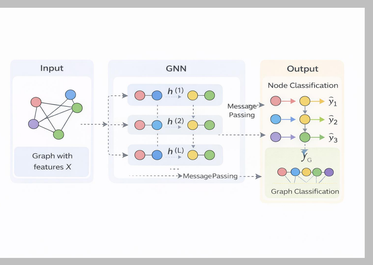

Figure 2.
A simple overview of a Graph Neural Network (GNN). The model takes a graph as input, lets nodes share information with their neighbors through multiple layers, and then uses the updated node representations to make predictions.

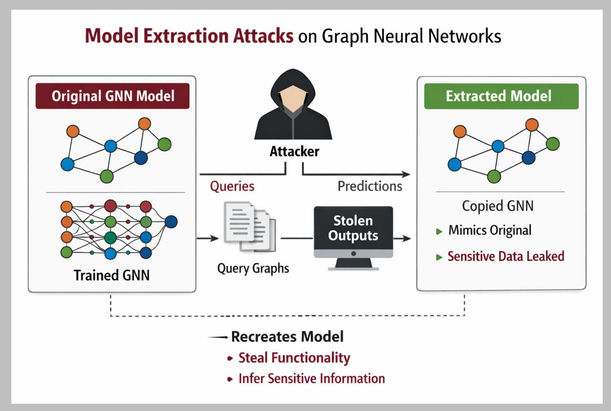

Figure 3.
An attacker queries a trained GNN, collects its outputs, and uses them to recreate a copy of the model, potentially exposing sensitive information.

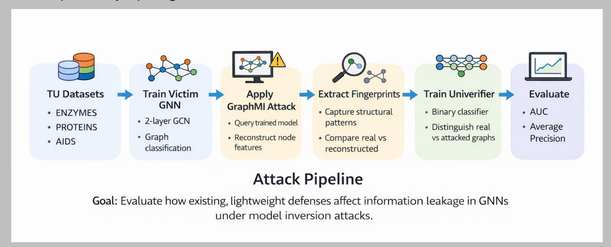

Figure 4.
Overview of the GraphMI attack pipeline used in this study

Methodology:

Goal

The goal of these experiments is to evaluate how existing, lightweight defenses affect information leakage in GNNs under model inversion attacks.

Attack Pipeline

- Reproduced the GraphMI attack using the PyGIP framework
- Trained Graph Neural Networks (GNNs) for graph classification on standard TU benchmark datasets
- Extracted model "fingerprints" from attacked GNNs
- Used extracted fingerprints to train a Univerifier
- Univerifier distinguishes real graphs from reconstructed graphs generated by the attack
- Evaluated attack performance using:
  - AUC (Area Under the ROC Curve)
  - AP (Average Precision)

Datasets

We ran experiments on three standard TU benchmark datasets. Which includes:
- ENZYMES
- PROTEINS
- AIDS

These datasets each have a different structural complexity and feature dominance, which makes them useful for understanding how effective a defense is or not.

Defenses

We implemented and tested three training-time defenses applied to the victim GNN:
- DropEdge: randomly removes edges during training to weaken structural information that is relied on by GraphMI.
- Feature Masking: randomly masks node features to reduce feature leakage.
- Gaussian Feature Noise: adds noise to node features during training as a light "disruption"

AUC (Lower = Better Privacy)

Dataset	Baseline	DropEdge	Feature Mask	DP Noise
ENZYMES	0.5418	0.4805	0.5418	0.4669
PROTEINS	0.5300	0.4823	0.5028	0.4823
AIDS	0.5027	0.5027	0.4829	0.5126

Figure 5.
Area Under the ROC Curve (AUC) measures how well the attacker distinguishes reconstructed graphs from real graphs, with lower values indicating better privacy protection.

AP (Lower = Better Privacy)

Dataset	Baseline	DropEdge	Feature Mask	DP Noise
ENZYMES	0.1397	0.1228	0.1397	0.1216
PROTEINS	0.1015	0.0919	0.0946	0.0919
AIDS	0.0493	0.0493	0.0482	0.0506

Figure 6.
Average Precision (AP) measures how accurately the attacker reconstructs hidden graph information, with lower values indicating better privacy protection.

Results:

We measure attack success using AUC (Area Under the ROC Curve) and AP (Average Precision). Both metrics evaluate how well the attacker reconstructs hidden graph information. Values close to 0.5 indicate random guessing, while higher values indicate stronger privacy leakage. Therefore, lower AUC and AP mean stronger defense. The major takeaways from the data shown above the following:
- DropEdge consistently reduces leakage on structure-sensitive datasets.
- DP Gaussian Noise can be highly effective but is dataset-dependent.
- Feature Masking alone is not consistently strong.
- No single defense works best across all graph types.

Overall, the results show that privacy defense performance depends on the underlying graph structure and feature distribution.

Conclusion:

This study demonstrates that baseline GNNs are vulnerable to GraphMI model extraction attacks, with sensitive graph information reconstructed above random levels. While structural and feature-level defenses reduce leakage, their effectiveness is highly dataset-dependent.
These findings highlight a key challenge in GNN security: privacy robustness is not universal. Defense strategies must be selected based on graph structure and feature characteristics rather than applied uniformly.
As graph-based AI systems are increasingly deployed in domains such as healthcare, finance, and cybersecurity, understanding model extraction risks is critical to protecting sensitive relational data.

Resources:

[1] Z. Zhang *et al.*, "Model Inversion Attacks Against Graph Neural Networks," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 9, pp. 8729–8743, 2023.
[2] X. You *et al.*, "GNNFingers: A Fingerprinting Framework for Verifying Ownerships of Graph Neural Networks," in *Proc. ACM Web Conf. (WWW '24)*, 2024.