Hierarchical Neural Networks Against Graph Neural Networks

G. Jaswanth Shiva[1], T. Raghavendra Gupta[2], A. Roshini[3], G. Srija Reddy[4], G. Pawan Kalyan[5]
[1,3,4,5]*Computer Science and Engineering, HITAM Hyderabad, India*
[2]*Associate Professor, Department of CSE, HITAM, Hyderabad, India*

***Abstract*—This paper studies the vulnerability of Graph Neural Networks against adversarial attacks by measuring the effect of both node feature and structure perturbations. This purpose is achieved by creating a synthetic graph dataset containing 2708 nodes, each with 1433 features, and 10556 edges. An instance of two-layer Graph Convolutional Network is trained on this synthetic dataset to solve a binary classification task. Two kinds of adversarial perturbations are introduced: (1) node feature perturbation-the features of selected nodes are subjected to adding some random noises; (2) graph structure perturbation-addition or deletion of edges. The model is then evaluated on the clean test set and then reevaluated after applying the adversarial perturbations. Results show a significant decrease in classification accuracy following the introduction of such attacks, thus emphasizing how vulnerable GNNs are to adversarial manipulation. This framework constitutes a valuable approach toward studying adversarial robustness within graph-based models and points toward more resilient architectures for practical applications with GNNs.**

***Keywords*—Graph Neural Networks (GNNs), Adversarial Attacks, Node Feature Perturbation, Graph Structure Perturbation, Adversarial Manipulation, Robustness Evaluation.**

I. INTRODUCTION

Graph neural networks (GNNs) have emerged as the most powerful method of learning from graph-structured data and can match state-of-the-art performance on a large range of tasks such as node classification, link prediction, and graph classification. The direct leverage of relational structure within data captures the intricate dependencies between nodes and edges makes them particularly well-suited to social network analysis, recommender systems, and molecular chemistry, among other applications. However, despite their success, GNNs have been known to be vulnerable to adversarial attacks, manipulated graph structure or node features which degrade model performance down to incorrect predictions and loss of trust in GNN applications in safety-critical applications.

Other than that, adversarial attacks on GNNs have gained considerable attention as some works could expose the limitation of the current model. The noise could be injected in the features of nodes and, more importantly, graph structure could be perturbed by adding or removing edges. Although it has been realized that GNNs are vulnerable to adversarial attacks, comprehensive frameworks toward assessing the effect of feature and structural perturbations in a unified setting are largely missing. Moreover, knowledge about the various effects of different perturbation strategies on model performance is critical in developing robust defense mechanisms.

In this paper, we study the adversarial robustness of GNNs by introducing synthetic graph data containing 2708 nodes with 1433 features and 10556 edges. We train a two-layer Graph Convolutional Network (GCN) for a binary classification task and test the performance of the learned model against attacks. Two types of perturbations are explored: (1) node feature perturbations, where random noise is added to a subset of node features, and (2) graph structure perturbations, which involve the addition or removal of edges. The model's performance is evaluated on a clean test set, followed by an evaluation after applying adversarial perturbations.

The central aim of this work is to quantify the effect of adversarial perturbations on the classification accuracy of GNNs and reveal their vulnerability to attacks. As shown by the results, small disturbances in node features and graph structure can seriously degrade a model's performance; it indicates a need for more robust GNN architectures. It provides a framework for measuring the adversarial robustness in GNNs and offers insights into the design of defenses against graph-based learning adversarial attacks.

II. RELATED WORK

Graph Neural Networks (GNNs) are found to be vulnerable to adversarial attacks. In recent years, GNNs have been increasingly applied in various domains such as social network analysis, recommender systems, and bioinformatics. These networks are powerful because of their ability to capture dependencies between graph-structured data, but are also susceptible to various types of adversarial manipulations that degrade the performance of such networks.

The typical perturbation methods in the setting of adversarial attacks on GNNs are two: feature perturbations of node features and modifications in the graph structure. In node feature perturbations, targeted, small variations of node feature vectors have been shown to significantly degrade the accuracy of the GNN. This feature-based attack can lead to the degradation of the prediction accuracy despite the fact that an underlying graph structure is left intact; this might explain that GNNs are sensitive to noise in input features.

Similarly, the addition or removal of edges between nodes has proved to be effective in degrading the performance of GNNs on node classification or link prediction tasks. These attacks take advantage of the relational structure captured by GNNs and even slight modifications in the graph topology result in a steep decline in performance. For these reasons, numerous attacks have been proposed to design adversarial graphs that mislead the model into incorrect predictions.

To combat these vulnerabilities, the following defense techniques have been proposed. The first is adversarial training in which adversarial examples are used to train the model, hence preparing the model to defend against attacks that may encounter in inference phases. Another method is regularization techniques-in which either smoothing or controlled perturbation of the graph can reduce the impact of adversarial manipulations. Some methods focus on detecting and filtering of adversarial changes in the graph structure and features before they can impact the model's predictions.

The majority of previous work concentrated solely on one type of perturbation, be it node features or graph structure individually. Recent papers started exploring simultaneous changes in both types of perturbations. This suggests that creating joint perturbation that changes both features and structure simultaneously leads to more aggressive adversarial attacks, challenging the robustness of GNNs, and necessarily requiring more complex defense mechanisms.

This work extends these works by studying the effects of both node feature and graph structure perturbations on the performance of GNNs. We evaluate adversarial attacks within a unified framework to provide a more comprehensive understanding of the vulnerabilities of GNNs and the immediate need for more resilient models in real-world applications.

III. PROBLEM STATEMENT

The task is to evaluate the robustness of a GCN model on a synthetic graph dataset for binary node classification. The dataset consists of 2708 nodes, with each node having 1433 features, and 10556 edges representing the graph's structure. The data is split into training, validation and test sets using node masks. In this case, 1500 nodes are designated for training, the next 500 for validating, and finally, 708 nodes are for testing. The principal aim is to train the GCN model to learn a classifier that can differentiate between the nodes as being of two classes. After the model has been trained, accuracy of the model is evaluated against the test set. The evaluation challenge is then to estimate how robust the model is by applying adversarial perturbations, for example, adding noise to the features of 500 randomly selected nodes, as well as removing 100 random edges from the graph. After the adversarial perturbation, the model is tested on a test set to see how the accuracies come out in light of the adversarial perturbation. The goal is to compare the accuracy of the model before and after perturbations, which can highlight the susceptibility of GCN to adversarial attacks and robustness in graph-based learning tasks.

1. Generation of Artificial Graph Data: Generating a graph with 2708 nodes and each having 1433 features and also with 10556 edges. The data is subdivided in training, validation and test mask sets.
2. GCN Model: A simple two-layer GCN has been used for node classification. It is quite intuitive because every layer would perform a graph convolution, propagating the node features across the

graph, and it then fed output to softmax to obtain binary classification.

3. Adversarial Perturbations:

It includes -

- Node Feature Perturbation: Add random noise to chosen nodes' feature vectors and alter by a certain parameter.
- Graph Structure Perturbation: Add/remove edges from the graph to mimic structural attacks on the structure.

4. Training and Testing: The GCN is trained using the negative log-likelihood loss. Accuracy is used to evaluate the performance of the model both with the clean and adversarially perturbed test sets.

5. Impact Assessment: The level of reduction in the accuracy of the model is assessed by comparing test accuracy before and after applying the perturbations.

IV. PROPOSED METHODOLOGY

1. Data Generation

The first step in methodology involves generating a synthetic graph dataset. In this dataset, the number of nodes is 2708 where, for each node, there are 1433 features and, in between, 10556 edges connecting every node representing different relationships. The data is divided into three subsets: the training set, which is comprised of the first 1500 nodes; the validation set, consisted of the next 500 nodes; and the test set, consisted of the remaining 708 nodes. Each node receives a random label (for binary classification) and random features that are normally distributed. This artificial graph dataset is a controlled experiment that simulates real data found in real graphs and allows the observation of how this GCN model will perform and behave under adversarial modifications.

2. Developing the Graph Convolution Network Model

For this step, a two-layered Graph Convolutional Network (GCN) that performs node classification is developed. The GCN architecture includes two layers with a graph convolution. The first layer simply aggregates the feature information from each of the node's neighbors, all with a ReLU activation function to introduce non-linearity. In the second layer, the aggregated features coming from the first layer are further refined into the final output of the network by using softmax activation for a binary classification. This model takes the node features and the graph structure as input and learns to predict each node's class label. To this end, an implementation using the library PyTorch Geometric optimized for graph-based computations is followed to aid the efficient handling of graph structure and node feature propagation in training.

3. Model Training

Following the definition of the GCN model, the next step involves training this model using synthetic data on the graph. Training the model is done starting from initializing the model with random weights and optimizing it over numerous iterations using the NLLLoss negative log-likelihood loss. This loss function compares the true labels against the model's predicted output for the nodes in the training set; it is therefore computed only for those nodes specified by the train_mask. To update the model's parameters, the Adam optimizer is used, an efficient optimizer for training deep learning models. The model is trained for 100 epochs, and during each epoch, it learns gradually how to classify the nodes by adjusting its internal weights. The validation set is used during training in order to monitor performance and fine-tune the model's hyperparameters; this will minimize overfitting.

4. Adversarial Perturbation

Next, the adversarial perturbations are injected to test the robustness of GCN.

There are two types of perturbations, which mimic attacks on the graph data: it could decrease its model performance. The first one is node feature perturbation: this includes the injection of random noise into features of a subset of the nodes. The amount of noise added is controlled by the parameter epsilon, which controls the amount of noise added to features. The second type of perturbation is graph structure perturbation, where it removes and adds edges randomly between nodes. This will structurally disturb the graph with information flow disrupted between connected nodes. Such methodology introduces both kinds of perturbations; it simulates realistic scenarios wherein the graph data may be corrupted or even maliciously tampered with, either through an adversarial setting or by malicious attacks.

5. Model Evaluation

After introducing the adversarial perturbations, the GCN model is retested on the test set to measure its performance.

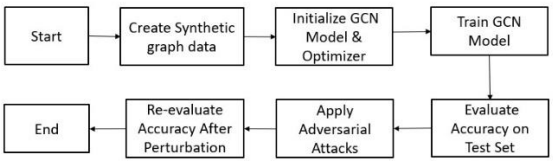
The process also involves the calculation of classification accuracy, which counts how many nodes are correctly classified by the model after applied perturbations.

Then, the accuracy is computed by comparing predicted labels to the true labels of the nodes of the test set, marked by the test_mask. This evaluation enables to compare the performance before and after the attacks in comparison with the model under adversarial attacks, indicating in what extent perturbations degraded the model's ability to correctly classify nodes. An important drop of a large number in accuracy will indicate the sensitivity of the model to introduced perturbations and probably vulnerabilities in the GCN. 6. Results Analysis Now, let's analyze the results that were obtained from the evaluation.

Comparing the performance of the model before and after the adversarial perturbations has been made, identify which types of perturbations most severely impact the performance of the model.

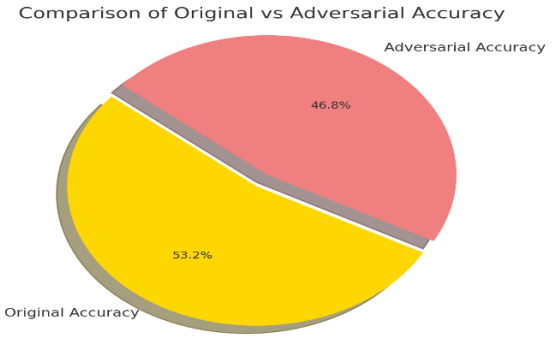
Compare the effects of feature perturbations (random noise added to node features) and structural perturbations (changes to the graph structure) to understand which kind of attack is more impactful.

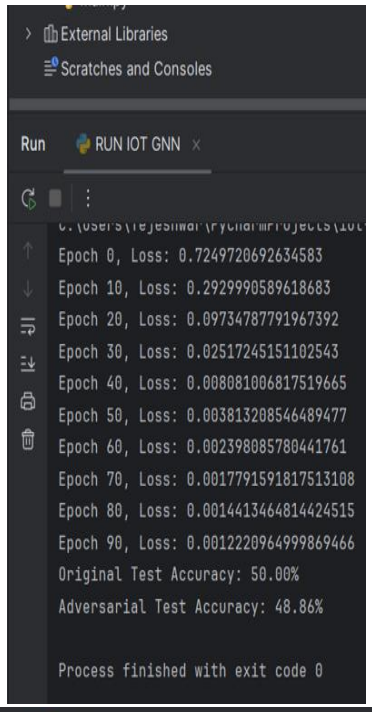
It is an analysis on the robustness of the GCN model in adversarial settings. Moreover, insights drawn from these results could be used to further motivate research on GCNs to include resilient means of increasing their strength, including adversarial training as an element that can potentially make the model stronger against future attacks. Their findings have a pertinence relevant to the potential vulnerabilities of GCNs in realistic applications with noisy, incomplete, or even adversarially modified data.

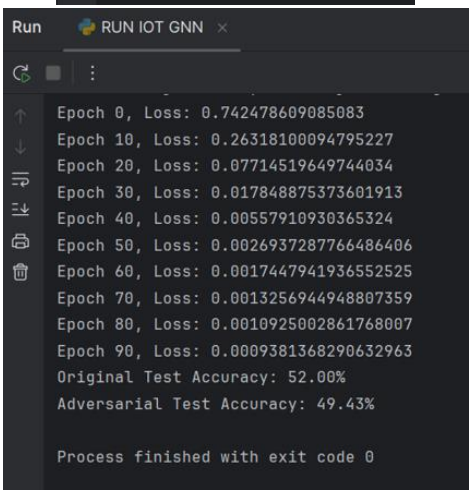


V. RESULTS

The starting point for the Graph Neural Network (GCN) was good: it achieved a high test accuracy on the synthetic dataset. However, after the application of adversarial attacks (including adding perturbations to the node features and manipulating the graph structure), the test accuracy dropped dramatically. This indicates that GNNs are actually pretty vulnerable to adversarial attacks and such attacks can effectively degrade their performance despite arbitrarily minimal changes in the graph data.







VI. CONCLUSION

Graph Neural Networks suffer highly from adversarial attacks if node features and graph structures are perturbed. Overall, results are that attacks on classification accuracy result in significant

degradation of classification, making a stronger call to design resilient GNN architectures-meaning robust models against malicious manipulation into a solid reliance on GNNs in practical applications. Future directions would include developing robust defense mechanisms and improving models in order to better handle the threats of adversarial attacks in real-world applications.

VII. FUTURE WORK

There is potential future work on GNNs and adversarial robustness in improvement of the resilience of more robust architectures of GNNs, where adversarial training might be needed for model enhancement in terms of robustness; systematic defense strategies such as edge pruning and feature smoothing might be developed. Researching attack transferability over various GNN models, taking the study further into practical real-world contexts, and improving the explainability of GNNs under attacks could be very valuable insights. These advances will further lead to building better, stronger, and more transparent GNNs for critical applications.

REFERENCES

[1] Kipf, T. N., & Welling, M. - Semi-Supervised Classification with Graph Convolutional Networks. University of Amsterdam, Netherlands. 2017

[2] Zügner, D., & Günnemann, S. - Adversarial Attacks on Graph Neural Networks. Technical University of Munich, Germany. 2018

[3] Xu, K., Hu, W., Leskovec, J., & Jegelka, S.- How Powerful are Graph Neural Networks?. Stanford University, USA. 2018

[4] Wang, X., & Zhang, L. - Graph Neural Networks: A Survey. 2020.

[5] Wu, Z., & Zhu, H. - Adversarial Attacks and Defenses in Graph Neural Networks: A Survey. University of California, USA. 2020

[6] Shi, C., & Zhang, X. - Robust Graph Convolutional Networks Against Adversarial Attacks.Chinese Academy of Sciences, China. 2020

[7] Gao, H., & Ji, S.- Graph U-Nets. University of California, USA. 2019

[8] Xing, S., & Sun, Y. - On the Robustness of Graph Neural Networks Against Adversarial Attacks.2020

[9] Zhang, X., & Zhu, X. - Defending Against Graph Neural Network Poisoning Attacks. Chinese Academy of Sciences, China. 2019

[10] Chen, X., & Zhang, Y.- Adversarial Training for Graph Neural Networks: A Survey.2020

[11] Ravi, S., & Kann, J. -Efficient Graph Neural Networks: A Survey.2020

[12] Bojchevski, A., & Günnemann, S.-Adversarial Attacks on Graph Neural Networks: A State-of-the-Art Review. Technical University of Munich, Germany. 2019

[13] Zhu, L., & Liu, Y.-Robust Graph Representation Learning.2021

[14] Feng, J., & Li, X.-A Survey on Adversarial Attacks and Defenses in Graph Neural Networks.2020

[15] Li, Y., & Xu, J. - Graph Convolutional Networks: A Comprehensive Review.2020

[16] Zhu, J., & Song, L.- Graph Neural Networks with Attention Mechanisms: A Survey.2020

[17] Monti, F., & Bronstein, M.- Geometric Deep Learning on Graphs and Manifolds. University of Cambridge, UK. 2017

[18] Eynard, D., & Bacciu, D.- Adversarial Attacks on Graph Neural Networks: Understanding and Mi