A Graph Neural Network Approach for Lateral Movement Attack Detection on IoT Network

Anfal Tariq, Nida Fatima, Aoun Muhammad, Umer Fayyaz, Sehrish Raza
Faculty of Engineering and Technology
The Islamia University of Bahawalpur
Bahawalpur, Pakistan
anfal.ch212@gmail.com, nida60271@gmail.com, aoun.muhammad@iub.edu.pk, umar.fayyaz@iub.edu.pk, sehrishraza6@gmail.com

***Abstract*—IoT networks have facilitated cognitive cities, industrial control systems, and healthcare systems, they have also posed very serious security risks due to device weaknesses, variety, and deficient security. Among these dangers, lateral movement attacks are very covert and rely on hacked IoT devices with legitimate user authentication for movement within networks. To fill these voids, this paper a suggested a Graph Neural Network (GAT method) based edge classification for lateral movement detection on IoT networks. By portraying the network as a graph with IoT devices as nodes and communications as edges, the proposed approach uses self-attention to assign weights and highlight communications based on harmless and hostile traffic. The GAT approach on the TON-IoT dataset surpasses LSTM and CNN benchmarks with an accuracy and F1-score of 0.9970. Attention visualization assists analysis and highlights likely paths for attacks. Word Count: 125**

***Index Terms*—Graph Neural Networks (GNN), Graph Attention Network (GAT), Lateral Movement Detection, IoT Security, Edge Classification**

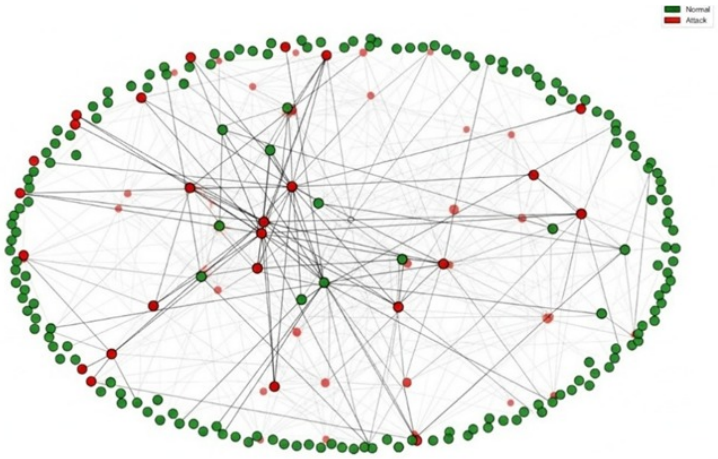

Fig. 1. IoT network graph illustrating lateral movement

I. Introduction

A. Background

An intrinsic dilemma exists between the functional value of IoT devices and their security posture: restricted computational resources,insufficient power budgets,and economical design often result in the implementation of poor or hard-coded authentication mechanisms,rare firmware updates, and the use of unprotected communication protocols [1].Among the array of cyber threats aimed at IoT ecosystems,lateral movement attacks have emerged as a particularly concealed,impactful,and operationally destructive pathway [2], [3].This disconnected view hinders them from simulating the multi-device;multi-hop relational connections and the chronological pattern that comprise the core of a lateral movement attack [4], [5].

B. Motivation

Lacking a model of this layout, deviation detection becomes a crude tool [5].Also,GNNs are by design equipped of dealing heterogeneous graphs with varied node and edge types,making them a suitable choice for modeling heterogenous IoT ecosystems [6].The Graph Neural Networks (GNNs),and more particularly Graph Attention Networks (GATs),stand out as a conceptually based and functional resolution to these challenges [7].The GAT forms integrate a self-attention mechanism into this pooling process, allowing the model to flexibly derive a weight for each neighbor [7].

C. Contributions

To ascertain and evaluate the usefulness of graph-structural learning,build and extensively appraise our GAT model against two robust,non-graph deep learning benchmarks.We conduct thorough trials using the TON-IoT(test/train) dataset [1].We devise methods to produce attention heatmaps over the network graph,pictorially showcasing the edges and nodes that were most influential in the model's labelling decision.

DOI: https://doi.org/10.67535/tsp.000002.016
Published under the ISAISS 2026 Proceedings

II. Related Work

A. Graph Neural Networks for Security

The use of Graph Neural Networks to cybersecurity problems signifies a notable transformation,shifting from established feature-vector based machine learning to models that directly infer over connections [8], [9] .The power of GNNs to propagated influences through in a series layers of neighborhood pooling makes them ideally fitted for uncovering intricate,staged attack spread pathways that cross several devices,which simpler models cannot understand [4], [5].

B. Lateral Movement Detection in IoT Networks

Initial systems like Hopper demonstrated the merit of a graph-based framework by examining enterprise security logs to expose subtle lateral movement,though they were bases on custom built graph indicators and predefined criteria [2].Euler improved this by introducing an expandable GNN-based model tailored to uncover lateral movement from network flow data, showing the benefit of model derived patterns over labor intensive preprocessing [10].

C. Provenance and Subgraph Based Approaches

Noting data privacy issues in collective protection,P3GNN examines a data secure,federated GNN system for APT detection within Software-Defined Networks (SDN),enabling multiple organizations to together train a model without sharing private local graph data [11].The presented network-level graph approach centers on the cross-node communication behaviors that are the indication of lateral movement,while provenance methods can uncover the initial compromise and granular process within a infected host.

D. GNN Based NIDS Comparative Studies

As the domain of GNN-based NIDS matures, contrastive and analytical studies have emerged to review headway and pinpoint issues.Surveys by Zhong [8]and Bilot et al. [9] supply in depth classifications of GNN models, security uses, and accessible datasets,concisely outlining the cutting edge and pending research questions.

III. Methodology

The proposed technique is grounded in recent innovations in graph-based and self-supervised learning for intrusion detection.A Graph Attention Network (GAT) is applied to produce node encoding by combining local information using attention coefficients. This model selection aligns with aligning with recent research endorsing transparent GNN models for practical cybersecurity use.

A. IoT Network Graph Modeling

- Vertices (Nodes) V contains the diversity of an IoT network, including sensors, actuators, cameras, smart appliances, gateways, and backend servers.
- Edges are created between two nodes and if an interaction is logged between them within a specified time period.Edges can be directed or undirected, based on the goals of investigation.
- Edge Features are extracted from traffic logs and such as: packet count, total bytes sent, flow length, protocol type, TCP flags, service port, and timestamps.
- Edge Labels are for supervised learning, where 1 denotes the communication was part of a recorded hostile and 0 indicates normal traffic

B. Graph Neural Networks and the GAT Architecture

Graph Neural Networks are a genre of neural networks intended to work on graph arranged data. A crucial key selection is the training goal. In cybersecurity, both node and edge-level tasks are fitting:

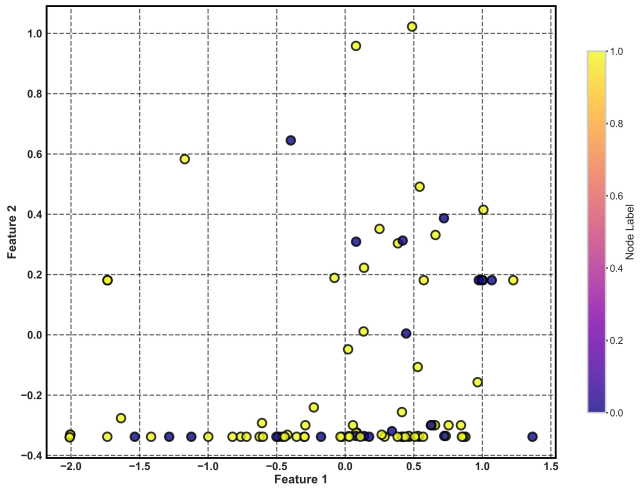

Fig. 2. Node classification illustration

Node Classification forecasts a label or category for each node. In security, this could be classifying devices as ”breached”,”exposed”,or “healthy”.While helpful,this approach can be auxiliary for spotting attacks that happen between devices.

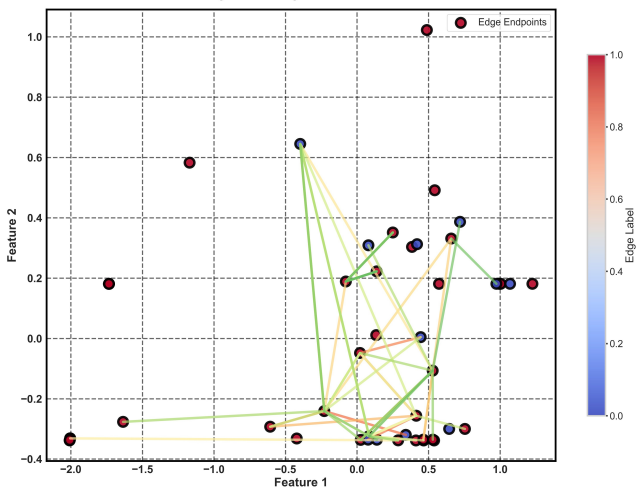

Fig. 3. Edge classification illustration

Edge Classification Predicts a label for each edge in the graph.To carry out edge classification,a common approach is to derive node embeddings using a GNN and then can form edge encoding,often by linking the embeddings of its origin and endpoint nodes.This vector is then passed to a multi-layer perceptron (MLP) model.

IV. Dataset

A. Dataset Description

For practical evidence,we utilize the TON IoT (The Things Network - IoT) dataset,a extensive and lifelike standard for IoT security research [1].

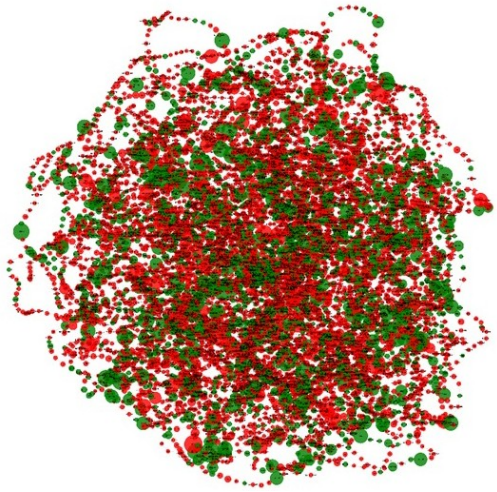

Fig. 4. Sample IoT network graph for training

V. Overview of the Proposed Framework

The core advancement resides in its explicit application of graph-structured learning to encode the indirect connections integral to lateral movement, a ability missing in traditional ordered or grid based models [5], [7].

A. GAT Model Architecture for Edge Classification

The model includes the Input Layer and Stacked GAT Layers with Multi-Head Attention.

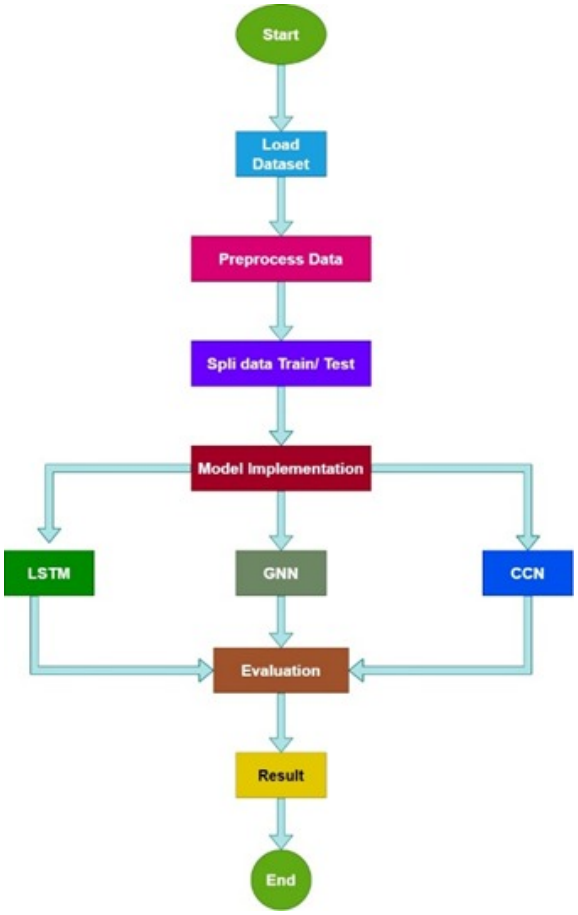

Fig. 5. Training and evaluation workflow

B. Baseline Model Implementations

In LSTM Model each sequence is constructed by lining up a device's communications in order.Each flow is converted by its attribute set.While in CNN Model:The flattened features are fed into a phased MLP for two class prediction.

C. Comparative Statistical Analysis

The GAT model markedly decreases undetected positives, which is essential for lateral movement detection. Enhanced results in relational rather than weight-based,derived from modeling using graphs.Explanations via attention weights provide probabilistic clarity, increasing reliability in predictions. Statistically, the presented GAT-based edge classification model is sturdy, stable, and closely matched with the probabilistic nature of lateral movement attacks, surpassed sequential model and grid structured deep learning models given identical test setups.

TABLE I
Model Strength and Limitations

Model	Strength	Statistical Limitation
CNN	Captures local correlations	Lacks relational reasoning
LSTM	Captures sequential order	Neglects cross-Series correlations
GAT	Learns node relationships with context	Increased runtime

D. Evaluation Metrics

- Accuracy:Measures total correct predictions but can be skewed with imbalanced data.

$$\text{Acurracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}$$

- Precision:The ratio of identified malicious edges that are truly malicious. High precision is essential to reduce false alarms for security operators.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

- Recall:The portion of actual malicious edges that are accurately detected. High recall is basic to ensure the system overlooks as few attacks as possible.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

- F1-Score:The harmonic mean of precision and recall, providing a single score of a model's capability.

$$\text{F1} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

E. Results and Performance Analysis

The performance of the described GAT approach and the standard LSTM and CNN models on the TON-IoT test set is shown:

Model	Accuracy	Precision	Recall	F1-Score
GAT	0.9970	0.9970	0.9970	0.9970
LSTM	0.7608	0.5788	0.7608	0.6574
CNN	0.7671	0.8028	0.7671	0.6731

Fig. 6. Detection performance of GAT,LSTM,CNN

This chart contrasts the efficiency of the suggested GAT model with LSTM and CNN baseline models. The high recall (0.9970) of the GAT model indicates it is highly successful at detecting the most of hostile lateral movement edges. The strong recall (0.9970) of the GAT model suggests it is highly successful at catching the most of hostile lateral movement traversal links.

VI. Discussion

A. Why GAT Outperforms

The attention mechanism [7] is not a basic enhancer;it is anomaly recognizer designed for security.This is a more coherent and streamlined approach than deriving node states and then inferring edge classifications,resulting in more accurate and actionable alarm [5].

VII. Results and Discussion

These results are supported by existing studies showing that focus guided and trained via comparison GNNs attain better detection performance on uneven and intricate intrusion datasets. Overall, the outcomes support that GAT-based edge classification is a robust, explainable, and feasible approach for lateral movement detection in IoT networks.

A. Limitations of the Present Work

A fixed model cannot by nature adapt to these temporal dynamics or detect attacks that occur over very long timescales with non-periodic activities [12], [13].A model's robustness to unseen, new threats that differ significantly from those in the training set is still open [14].In practice, IoT networks may be controlled by different organizations unwilling to share sensitive metrics for centralized training, limiting the model's ability to learn from a broad range of examples [11].

VIII. Future Research Directions

To bypass disjointed dataset and privacy concerns, future work should explore decentralized learning frameworks for GNN [11]. Intruders will assuredly try to craft traffic to avoid the system. Research must study the strength of GNN-based IDS against adversarial attacks [14].This work is restricted to network-level graphs. A strong integration would be to merge these with host-level provenance graphs [15], [16]into a multi model, heterogeneous GNN.

IX. Reference

References

[1] W. W. Lo, S. Layeghy, M. Sarhan, M. Gallagher, and M. Portmann, "E-graphsage: A graph neural network based intrusion detection system for iot," *arXiv preprint arXiv:2103.16329*, 2021.

[2] G. Ho, M. Dhiman, D. Akhawe, V. Paxson, S. Savage, G. M. Voelker, and D. Wagner, "Hopper: Modeling and detecting lateral movement," in *30th USENIX Security Symposium (USENIX Security 21)*, 2021, pp. 3093–3110.

[3] S. Freitas, A. Wicker, D. H. Chau, and J. Neil, "D2m: Dynamic defense and modeling of adversarial movement in networks," in *Proceedings of the 2020 SIAM International Conference on Data Mining*. SIAM, 2020, pp. 541–549.

[4] Y. Wang, Z. Han, J. Li, and X. He, "Bs-gat behavior similarity based graph attention network for network intrusion detection," *arXiv preprint arXiv:2304.07226*, 2023.

[5] X. Li, J. Zhang, Y. Yuan, and C. Zhou, "Network intrusion detection with edge-directed graph multi-head attention networks," *arXiv preprint arXiv:2310.17348*, 2023.

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

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

[8] M. Jin, H. Y. Koh, Q. Wen, D. Zambon, C. Alippi, G. I. Webb, I. King, and S. Pan, "A survey on graph neural networks for time series: Forecasting, classification, imputation, and anomaly detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

[9] T. Bilot, N. El Madhoun, K. Al Agha, and A. Zouaoui, "Graph neural networks for intrusion detection: A survey," *IEEe Access*, vol. 11, pp. 49 114–49 139, 2023.

[10] I. J. King and H. H. Huang, "Euler: Detecting network lateral movement via scalable temporal link prediction," *ACM Transactions on Privacy and Security*, vol. 26, no. 3, pp. 1–36, 2023.

[11] H. Nazari, A. Yazdinejad, A. Dehghantanha, F. Zarrinkalam, and G. Srivastava, "P3gnn: A privacy-preserving provenance graph-based model for autonomous apt detection in software defined networking," in *Proceedings of the Workshop on Autonomous Cybersecurity*, 2023, pp. 34–44.

[12] D. Xu, C. Ruan, E. Korpeoglu, S. Kumar, and K. Achan, "Inductive representation learning on temporal graphs," *arXiv preprint arXiv:2002.07962*, 2020.

[13] E. Rossi, B. Chamberlain, F. Frasca, D. Eynard, F. Monti, and M. Bronstein, "Temporal graph networks for deep learning on dynamic graphs," *arXiv preprint arXiv:2006.10637*, 2020.

[14] Z. Zhan, H. Zhou, and H. Haddadi, "Real-iot: Characterizing gnn intrusion detection robustness under practical adversarial attack," *arXiv preprint arXiv:2507.10836*, 2025.

[15] F. Yang, J. Xu, C. Xiong, Z. Li, and K. Zhang, "{PROGRAPHER}: An anomaly detection system based on provenance graph embedding," in *32nd USENIX Security Symposium (USENIX Security 23)*, 2023, pp. 4355–4372.

[16] Z. Cheng, Q. Lv, J. Liang, Y. Wang, D. Sun, T. Pasquier, and X. Han, "Kairos: Practical intrusion detection and investigation using whole-system provenance," in *2024 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2024, pp. 3533–3551.