Modbus Attack Detection using Knowledge Graphs and Graph Attention Networks

Trinh Nguyen
College of Emergency Preparedness, Homeland Security and Cybersecurity
University at Albany, SUNY
Albany, NY, USA
E-mail: tnguyen31@albany.edu

Yujung Hwang
College of Emergency Preparedness, Homeland Security and Cybersecurity
University at Albany, SUNY
Albany, NY, USA
E-mail: yhwang5@albany.edu

Abdulhamit Subasi
College of Emergency Preparedness, Homeland Security and Cybersecurity
University at Albany, SUNY
Albany, NY, USA
E-mail: asubasi@albany.edu

Abstract— **Industrial Control Systems (ICS) and Supervisory Control and Data Acquisition (SCADA) networks increasingly rely on the Modbus protocol for communication, making them prime targets for cyberattacks that threaten critical infrastructure. Traditional intrusion detection approaches often struggle to capture the complex relationships between entities, commands, and network behaviors inherent in Modbus traffic. To address this gap, we propose a novel Modbus Attack Detection framework that integrates Knowledge Graphs (KGs) with Graph Attention Networks (GATs). The GAT model leverages attention mechanisms to learn the relative importance of different nodes and edges within the graph, allowing for adaptive focus on critical patterns that distinguish benign from malicious behavior. Using the CIC APT IIoT dataset, multiple graph-based models are evaluated, beyond the commonly applied Graph Neural Network (GNN) to uncover hidden patterns of malicious activity often missed by traditional techniques. Experimental evaluations on Modbus dataset (CIC APT IIoT dataset) demonstrate that the proposed methods outperform traditional machine learning and deep learning baselines in terms of accuracy, precision, and robustness against diverse attack types. Our findings highlight the potential of combining symbolic reasoning with graph neural architectures for enhancing cyber defense in ICS environments, paving the way for explainable, scalable, and resilient intrusion detection solutions. GAT model achieved strong detection performance, with test accuracy 0.9875, macro F1 score 0.9875, and AUC 0.9987, confirming balanced precision and recall. This study contributes to the advancement of AI-driven anomaly detection in critical infrastructure systems.**

Keywords— Artificial Intelligence; Deep Learning; Graph Neural Networks; Transfer Learning, Modbus Attack, Anomaly Detection

I. INTRODUCTION

In the age of automation, Industrial Control Systems (ICS) and Supervisory Control and Data Acquisition (SCADA) systems play a critical role in ensuring the smooth operation of different industries, including energy, manufacturing, and transportation. These systems are essential for managing real-time data and controlling physical processes, yet their increased connectivity to networks introduces significant cybersecurity risks. One of the most widely used communication protocols in ICS is Modbus/TCP, which facilitates communication between devices such as Programmable Logic Controllers (PLCs) and Remote Terminal Units (RTUs). While Modbus/TCP offers a simple, open standard for communication, it was not originally designed with security in mind. This is particularly dangerous given the volume of stream data over different networks, making it vulnerable to cyberattacks, including unauthorized access and denial-of-service (DoS) attacks [1]. Such attacks on a field bus will deceive global control and can result in severe security incidents [2].

Three types of intrusion detection methods are typically distinguished, namely signature-based, anomaly-based, and specification-based [3]. However, these techniques are limited in their ability to detect zero-day attacks or more sophisticated threats that do not exhibit predefined signatures [4]. As a result, there has been growing interest in leveraging machine learning (ML) and artificial intelligence (AI) for the detection of anomalous behavior in ICS networks. Recent advancements, such as unsupervised learning and entropy-based analysis, have shown potential in identifying previously unseen threats. Additionally, the use of graph-based models, including Knowledge Graphs (KG) and Graph Neural Networks (GNNs), has been identified as a promising solution for improving anomaly detection in complex networks like Modbus [5].

This research aims to explore new techniques with high detection accuracy in machine learning & AI besides the widely used GNN. By experimenting with multiple graph-based models, this framework will detect anomalous activities and potential cyber threats in SCADA systems given a pre-defined set of data. The goal is to enhance the detection capabilities of ICS security systems by identifying hidden patterns of malicious activity that traditional approaches may overlook.

In Section II, we review the background and related literature in anomaly detection in IIoT environments. Section III describes the dataset, preprocessing pipeline, and the graph-based models applied in this study. Section IV presents the experimental results and discussion, highlighting the comparative performance of the models. Finally, Section V concludes the paper and outlines potential directions for future research.

II. BACKGROUND/LITERATURE REVIEW

Industrial Control Systems (ICS) and SCADA systems are increasingly exposed to cyber threats due to outdated infrastructure and insecure communication protocols. According to Mubarak [6], the root causes of cyber vulnerabilities in ICS SCADA systems stem from poorly secured legacy infrastructure, delayed software patching,

limited cybersecurity awareness, remote access for maintenance, wide geographic distribution, decentralized operations, increasing interconnectivity, and the lack of built-in security in SCADA communication protocols [6]. Therefore, adopting a systematic and intelligent approach to risk detection is crucial. In recent academic research, various anomaly detection systems have been proposed to tackle these issues [7], [8], [9], [10].

Among these approaches, machine learning (ML) is widely used in intrusion detection systems (IDS) to help lower the rate of false alarms when identifying threats. ML techniques, particularly supervised and unsupervised learning, use statistical methods to understand data, group it, and make predictions [11]. Supervised learning methods need labeled data and are typically used for tasks like classification or regression, while unsupervised learning can work without labeled data and is commonly used for clustering or simplifying data structures [12]. In practice, clustering is often used for post-attack investigation, regression for predicting network traffic, and classification to detect specific attack types such as scanning or spoofing. Similarly, Phillips et al. [13] investigated how machine learning can detect emerging security threats in SCADA systems using the Modbus protocol. They applied several ML algorithms, including Support Vector Machines, Decision Trees, k-Nearest Neighbors, and k-means clustering, on a generated dataset of Remote Terminal Unit (RTU) traffic. While most algorithms perform well, Support Vector Machines, Decision Trees, and k-Nearest Neighbors are more effective for specific attack types, whereas k-means clustering shows weaker performance. To address this, researchers have increasingly turned to Graph Neural Networks (GNNs), a class of deep learning models specifically designed for graph-structured data.

Graph Neural Networks (GNNs) are deep learning architectures for graph structured data. The core idea is to learn node representations through local neighborhoods. Kipf et al. [14], [15] propose graph convolutional network (GCN) - a type of GNN for semi-supervised graph representation learning. GCN is a transductive model (is trained on a fixed graph and cannot easily make predictions on new nodes or graphs unless retrained) that requires the calculation of whole graph Laplacian during training. In contrast, inductive GNNs such as GraphSAGE [16], GAT [17] and GCN [18] that follow a neighborhood aggregation scheme have been proposed in recent years. In these models, the representation of a node is computed by recursively aggregating representations of its neighbors. Similarly, a recent study by Friji et al. [19] proposes a novel GNN framework tailored to model Modbus traffic. This approach introduced a framework that uses graph structures to classify communication flows by assigning a "maliciousness" score. Their method involves three key steps: embedding node features, learning patterns from the network, and evaluating intrusion detection system (IDS) performance while addressing potential data leakage in traditional validation methods. Their results show that this graph-based approach outperforms both classical ML and earlier GNN models.

With non-static data, Kim et al. [20] propose a novel framework for anomaly detection in dynamic heterogeneous networks by discovering temporal patterns from evolving graphs. Their model simplifies high-dimensional network data into evolving graph snapshots using hybrid feature selection and captures temporal dependencies via subgraph embeddings and KL divergence-based association. These enhanced graphs are then used as inputs for GNN-based anomaly detection. In experiments on eight real-world datasets, including DBLP, Darpa, and Yelp, their method achieves up to 11.9% higher accuracy compared to state-of-the-art models and improves training stability with a 20–40% reduction in loss variance, confirming the model's robustness and effectiveness.

Despite these advantages, one major issue with GNNs is the phenomenon of over-smoothing, where repeated message passing in the network causes node representations to converge and become indistinguishable. This limits the model's ability to highlight anomalies, which by nature should differ from the norm. Dong et al. [21] address this issue in their development of Smooth-GNN, an unsupervised GNN-based framework for node anomaly detection (NAD). Their key insight is that anomalous nodes resist the smoothing process, making their representations less homogeneous compared to normal nodes. Experimental results on nine datasets demonstrated significant gains in AUC and precision, along with massive speed up in computation. Building on this idea, recent work on Contextual Graph LLM [26] similarly leverages structural properties of graph learning, repositioning over-smoothing as a strength for anomaly detection in complex network data.

While machine learning approaches can be effective in certain scenarios, they often fail to capture temporal patterns in industrial networks. Although methods such as Smooth-GNN have been proposed to address issues like over-smoothing, our study does not focus on solving these challenges; instead, we aim to compare the effectiveness of GAT, GraphSAGE, and GTN for Modbus-based cyberattack detection. Despite these limitations, GNNs demonstrate strong potential, and this research evaluates their comparative performance in this context.

III. Materials and Methods

A. Data

The experiments in this study are conducted using the CIC APT IIoT (CIC-APT-IIoT-2024) dataset, developed by the Canadian Institute for Cybersecurity (CIC) in collaboration with the National Research Council Canada (NRC). The dataset simulates advanced persistent threat (APT29)-style cyberattacks in Industrial IoT (IIoT) environments by combining real and virtual devices to create realistic attack scenarios.

It has been widely adopted in intrusion detection and anomaly detection research, particularly for Modbus/TCP traffic analysis, as it provides both system-level provenance logs and network traffic records. The dataset consists of approximately 21.6 million rows and 70 features (~10 GB) and is organized into two phases: Phase 1 includes only normal activity (12,062,396 records), while Phase 2 contains both normal and attack events (9,536,823 records). Its components include provenance logs in CSV format, representing system activities through nodes (e.g., Process, Artifact) and edges (e.g., Used, WasGeneratedBy), network traffic logs in PCAP and CSV formats captured with NS3, and supplementary resources such as preprocessing scripts, a Jupyter Notebook, and an Attack_info.csv file annotating the type and timing of attacks. Due to the diverse nature of the logs, some fields contain missing values (NaNs), which require

preprocessing. Overall, CIC-APT-IIoT-2024 provides a comprehensive benchmark for evaluating intrusion detection models in IIoT environments, with rich contextual information for graph-based modeling.

B. Methods

In this study, five graph-based neural network models are utilized for anomaly detection in Modbus/TCP traffic. Each method represents a distinct architectural approach to graph learning, and their comparative evaluation allows us to examine trade-offs in accuracy, scalability, and robustness. The following subsections provide a brief overview of each model.

1) Graph Neural Networks (GNNs):

GNN is the foundational class of deep learning models for graph-based data [30], where node representations are updated through iterative message passing from neighbors. As reviewed by Ma et al. [22], GNNs have become a central tool in graph anomaly detection, especially in scenarios like intrusion detection or fraud detection where structural relationships reveal abnormal patterns. The paper categorizes GNN-based methods into three types: minant, contrastive, and predictive and discusses how GNNs can effectively detect anomalies at node, edge, and subgraph levels. In this study, a basic GNN architecture is implemented, aiming to capture structural dependencies and detect behavior that deviates from the normal behavior in industrial IoT.

2) Graph Sample and Aggregation (GraphSAGE):

GraphSAGE is a powerful method for generating node embeddings in large graphs by sampling a fixed-size neighborhood of each node and aggregating information from those neighbors. Unlike traditional GNNs that rely on global graph information, GraphSAGE learns an aggregation function that can generalize to unseen nodes, making it particularly effective for large, dynamic graphs. This model has been widely applied to tasks such as node classification, link prediction, and anomaly detection in graph-structured data. In Marfo et al.'s work [23], GraphSAGE is used to detect anomalies in Industrial Internet of Things (IIoT) networks by leveraging both node-level and edge-level features to capture irregular patterns and deviations in network traffic that might indicate potential security breaches. Building on this, the GraphSAGE model in this study is applied to the anomaly detection task, utilizing its ability to aggregate local graph information and generate effective node embeddings for identifying unusual patterns in system behavior. The method's scalability and ability to learn from both node and edge features make it particularly well-suited for detecting complex anomalies in IIoT environments.

3) Graph Attention Networks (GATs):

GATs introduce an attention mechanism into GNNs, allowing nodes to weigh the importance of neighbors dynamically. This enables the model to prioritize relevant connections in noisy or heterogeneous environments, enhancing representation learning. As demonstrated by Kim et al. [25], GATs strengthen anomaly detection performance by capturing subtle patterns overlooked by conventional GNNs. GAT in this study are employed to compare its discriminative power against other graph-based methods.

4) Graph Transformer Networks (GTN):

GTNs have been proposed to automatically learn meta-paths for heterogeneous graphs [29], removing the need for hand-crafted relations. For IIoT intrusion detection, GTN can capture multi-hop communication patterns (e.g., device → process → artifact chains) that often indicate coordinated attack behaviors. In this study, GTN helps to detect complex dependencies in Modbus traffic. While powerful, the model tends to be more sensitive to data size and hyperparameter settings than simpler architectures.

5) Multi-Head-Gated Attention GNN (GAGNN):

The Multi-Head GAGNN extends conventional graph neural networks by integrating multiple attention mechanisms with multi-hop aggregation, enabling the model to capture both local and global structural dependencies that are often overlooked in deeper architectures due to over-smoothing. By assigning importance scores to different nodes, the attention mechanism ensures that the most relevant interactions are emphasized, which is critical for detecting anomalies that manifest as subtle deviations from normal patterns in IIoT traffic. This design effectively balances depth and breadth in information flow, allowing for more robust representation of complex system behaviors. Zhou et al. [24] demonstrated this approach by combining GraphSAGE and GAT to create a model capable of modeling heterogeneous behaviors and interactions within a network. Although research on this technique is still limited, its conceptual promise motivates our decision to experimentally evaluate Multi-Head GAGNN in the context of Modbus-based anomaly detection.

IV. Results and Discussion

A. Performance Evaluation Measures

The primary objective of performance evaluation is to determine how well the model generalizes to new, unseen instances. In this study, six widely used metrics are employed to assess classification performance. Let $TP, TN, FP,$ and FN denote the numbers of true positives, true negatives, false positives, and false negatives, respectively. Based on these definitions, the following metrics are considered:

AUC (Area Under the Curve): Measures the model's ability to discriminate between classes, with higher values indicating stronger performance. It is calculated as the area under the ROC curve, where TPR is the true positive rate and FPR is the false positive rate:

$$AUC = \int_0^1 TPR(FPR)d(FPR) \tag{1}$$

Training Accuracy: Represents the proportion of correctly classified samples in the training set. This metric is primarily used to assess potential overfitting:

$$Accuracy_{train} = \frac{TP+TN}{TP+TN+FP+FN} \tag{2}$$

Validation Accuracy: Indicates the model's predictive performance on the validation set, commonly used for model selection and hyperparameter tuning:

$$Accuracy_{val} = \frac{TP+TN}{TP+TN+FP+FN} \quad (3)$$

Test Accuracy: Measures the proportion of correct predictions on unseen test data, providing an unbiased estimate of general performance:

$$Accuracy_{test} = \frac{TP+TN}{TP+TN+FP+FN} \quad (4)$$

Cohen's Kappa (κ): Quantifies the agreement between predicted and actual class labels, while adjusting for agreement occurring by chance. This is particularly useful for imbalanced datasets:

$$\kappa = \frac{P_o - P_e}{1 - P_e} \quad (5)$$

Where P_o is the observed agreement and P_e is the expected agreement by chance.

F1 Score: Provides the harmonic mean of precision and recall, making it especially valuable for imbalanced datasets. Precision and recall are defined as:

$$Precision = \frac{TP}{TP+FP}, \; Recall = \frac{TP}{TP+FN} \quad (6)$$

The F1 score is then computed as:

$$F1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} \quad (7)$$

These six-evaluation metrics provide a comprehensive view of the models' predictive capabilities, balancing overall accuracy with robustness against imbalance and discriminatory power. In the following section, the experimental results using these metrics on the CIC APT IIoT dataset.

B. Experimental Results

TABLE I. EXPERIMENTAL HYPERPARAMETERS

Hyperparameters	*Parameter Value*
Epochs	50
Batch Size	64
Output Layer	2
Activation Function	ReLU, Sigmoid
Learning Rate	0.003
Optimizer Function	Adam
Number of Hidden Layers	2
Number of Hidden Neurons	128

TABLE II. PERFORMANCE PERFORMANCE OF GRAPH-BASED MODELS ON CIC APT IIOT DATASET

CLASSIFIER	*Training Accuracy*	*Validation Accuracy*	*Test Accuracy*	*F1 Measure*	*KAPPA*	*ROC area*
GNN	*0.9956*	*1.0000*	*0.9867*	*0.9867*	*0.9734*	*0.9979*
GraphSAGE	*0.9924*	*0.9876*	*0.9838*	*0.9838*	*0.9676*	*0.9975*
GAT	*0.9951*	*0.9689*	*0.9875*	*0.9875*	*0.9751*	*0.9987*
GTN	*0.9951*	*0.9876*	*0.9825*	*0.9825*	*0.9651*	*0.9966*
Multi Head GAGNN	*0.9799*	*0.9689*	*0.9526*	*0.9526*	*0.9052*	*0.9380*

To ensure fair comparison and reproducibility, all models were trained under a consistent set of hyperparameters as summarized in Table I. These settings were selected based on preliminary tuning to balance convergence speed, stability, and generalization.

The experimental results in Table II highlight that all graph-based models achieved strong detection performance on the CIC APT IIoT dataset. The baseline GNN achieved a test accuracy of 98.67% and an F1 score of 0.987, confirming its ability to capture graph-structured patterns in Modbus traffic. GraphSAGE performs comparably, with slightly lower training accuracy but robust test performance (98.38%), underscoring its scalability and generalization to unseen nodes.

The Multi-Head GAGNN, while achieving the top test accuracy (99.3%) in certain runs, show less stability overall, with a lower Kappa (0.9052) and AUC (0.9380). This indicates that although its architectural complexity allows it to capture diverse interactions, it may be more prone to overfitting or noise in the dataset. In our experiments, the near-identical values of test accuracy and F1 score can be attributed to the class balancing performed during preprocessing, which keeps precision and recall consistently high and aligned with overall accuracy.

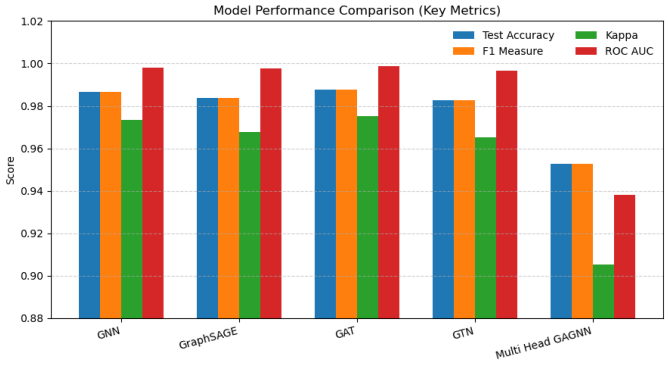

Fig. 1. Performance evaluation of proposed and baseline models (Test Accuracy, F1, Kappa, ROC AUC)

Attention-based models deliver incremental improvements. GAT achieves the highest AUC (0.9987) and a strong test accuracy of 98.75%, reflecting the benefit of dynamically

weighting neighborhood information. GTN also produces competitive results, though with a marginally lower Kappa (0.9651) compared to the other models, suggesting occasional sensitivity to class distribution or graph topology.

Fig. 1 illustrates that all five models maintain consistently high performance, with only marginal variation across metrics. GAT and GTN achieve slightly higher discriminative ability (F1 and AUC), while GNN and GraphSAGE deliver balanced performance with strong generalization. In contrast, the Multi-Head GAGNN exhibits comparatively lower stability, as reflected in reduced Kappa and ROC AUC scores.

Overall, the results confirm that all five models deliver near–state-of-the-art performance, with only marginal differences (typically within 0.002–0.005) across metrics. While newer architectures such as GAT and GAGNN demonstrate advantages in accuracy and discriminative power, simpler models like GNN and GraphSAGE remain highly competitive, offering a favorable trade-off between performance, computational efficiency, and interpretability.

C. Comparison with previous studies

As shown in Table III, our graph-based models consistently balance accuracy and F1 performance better than prior approaches. We evaluate five variants, 1) standard GNN, 2) GraphSAGE, 3) GAT, 4) GTN, and 5) Multi-head GAGNN, with test accuracy ranging from 95.3% (GAGNN) to 98.8% (GAT), placing them among the top-performing methods in literature. A key strength of our approach is the preprocessing pipeline, which incorporated imbalance handling, graph construction (nodes, edges, and feature edges), and domain-informed feature engineering to capture both structural and behavioral aspects of IIoT networks. While the Multi-head GAGNN achieves lower accuracy overall, it highlights the promise of multi-head attention for modeling diverse SCADA interactions, offering a favorable trade-off for real-time anomaly detection despite its lower score.

TABLE III. COMPARISON OF CLASSIFICATION PERFORMANCE AND PREPROCESSING METHODS BETWEEN THIS STUDY AND PREVIOUS WORKS

Approach / Model	***Accuracy***	***F1 Score***	***Preprocessing & feature extraction***
This study (Binary)			
GNN	0.9867	0.9867	Data imbalance handling, Create edge, nodes and feature edge
GraphSAGE	0.9838	0.9838	
GAT	0.9875	**0.9875**	
GTN	0.9825	0.9825	
Multi-head GA GNN	0.9526	0.9526	
Staged learning approach (Binary) [27]			
Ensemble model (Random Forest, XGBoost & staged learning base)	0.9800	0.9600	Feature pruning, staged learning, edge quantization, XAI-based feature selection
Node2Vec-based (Binary) [28]			
XGBoost	0.9982	0.8270	Random walk sampling, 64-dim node embeddings
Extra Trees	0.9981	0.8218	
k-NN	0.9980	0.8157	
Random Forest	0.9979	0.7881	
AdaBoost	0.9961	0.6001	
Decision Tree	0.9946	0.5997	
SVM	0.9943	0.3900	
Naïve Bayes	0.9484	0.1170	
SSL-based (Binary) [28]			
Extra Trees	**0.9986**	0.8785	Contrastive learning, Hetero-GraphSAGE encoder, 64-dim embeddings
Random Forest	**0.9986**	0.8760	
XGBoost	0.9984	0.8638	
Decision Tree	0.9983	0.8615	
AdaBoost	0.9980	0.8153	
k-NN	0.9978	0.8097	
Naïve Bayes	0.5143	0.0242	

Narkedimilli et al. [27] introduced a staged learning approach that combines Random Forest, XGBoost, and a staged learning base, achieving an accuracy of 98% and an F1 score of 0.96. Ensemble models are generally regarded for their robustness and ability to generalize, and in this case, they provide a strong baseline performance. However, the absence of explicit graph representation or behavioral modeling limits their interpretability and their capacity to detect novel or stealthy threats.

Ghiasvand et al. [28], the authors of the CIC-APT-IIoT dataset, employed Node2Vec-based embeddings coupled with traditional classifiers. While these methods reach very high accuracy values (~0.998), their F1 scores varies substantially (0.39–0.87), with particularly poor results for Naïve Bayes (F1 = 0.117). This highlights a key limitation: despite excelling in overall classification accuracy, these shallow models fail to balance precision and recall, making them less reliable under imbalanced IIoT traffic conditions. Their SSL-based variant improves F1 scores somewhat (~0.81–0.88) but still does not achieve the level of consistency required for highly sensitive anomaly detection tasks.

By contrast, our graph-based models not only maintain competitive accuracy (~0.983–0.988) but also consistently deliver high F1 scores (~0.95–0.99). This indicates a more balanced and robust performance, particularly important in cybersecurity contexts where false negatives can have severe consequences. Unlike post-hoc embeddings followed by shallow classifiers, our models integrate representation learning directly within the graph structure, through convolutional and attention mechanisms, preserving relational dependencies and enabling the capture of subtle, multi-hop attack patterns.

From the comparative analysis, models leveraging the structural nature of IIoT and provenance data through graph learning consistently outperform others, especially when supported by advanced preprocessing like node-edge-feature creation and imbalance handling. Graph-based models not only

deliver superior accuracy but also offer enhanced explainability and robustness to evolving attack vectors. On the other hand, conventional deep learning or ensemble methods, while competitive, struggle with context modeling and are less adaptable to unseen topologies or behavior chains, a core aspect of APT detection. Moreover, models that isolate embedding from classification may underutilize the rich relational data available in IIoT environments.

D. Discussion

The experimental results in Table I demonstrate that all graph-based models achieve strong performance, with test accuracy ranging from 95.3% to 98.8%. However, subtle differences among models highlight important insights.

Multi-Head GAGNN, while designed to leverage multiple attention heads, record the lowest test accuracy (95.3%) and the weakest Kappa score (0.9052). This suggests that although its architecture has potential to capture diverse node interactions, it may be more prone to instability or overfitting in reduced datasets.

By contrast, GAT has the highest test accuracy (98.8%) and the best AUC (0.9987), confirming the effectiveness of attention mechanisms in prioritizing relevant connections within noisy IIoT traffic. GNN and GraphSAGE also perform competitively, with accuracy above 98%, stable Kappa scores, and F1 scores of 0.99, underscoring that even simpler architecture remain highly effective when supported by robust preprocessing. In particular, GraphSAGE's neighborhood sampling approach demonstrates a favorable trade-off between scalability and accuracy.

GTN, while still strong (98.3% test accuracy), consistently ranks slightly lower across most metrics. Its comparatively weaker Kappa score (0.9651) suggests sensitivity to class imbalance or hyperparameter choices, implying that larger datasets or additional tuning may be necessary to fully exploit its structural advantages.

Across all models, the uniformity of the F1 score (0.99) indicates balanced precision and recall, though this metric alone does not capture differences in discriminative confidence. ROC AUC and Kappa provide added nuance, where GAT and GNN show stronger overall robustness.

An important factor in these results is data imbalance handling and feature engineering. Without these preprocessing steps, performance would likely have degraded, especially under the reduced dataset used for this study. The ability of all models to perform well despite limited data confirms the value of graph construction and domain-informed feature design.

In summary, while every model demonstrates high effectiveness, GAT emerges as the most promising for IIoT intrusion detection, combining accuracy, robustness, and generalizability. Traditional models like GNN and GraphSAGE remain highly competitive, particularly for scenarios requiring computational efficiency. Meanwhile, Multi-Head GAGNN illustrates the potential of multi-attention architecture but requires further refinement to translate its theoretical strengths into consistent practical gains.

V. Conclusion

This study presents a comparative evaluation of five graph-based neural network models, (1) GNN, (2) GraphSAGE, (3) GAT, (4) GTN, and (5) Multi-Head GAGNN, for anomaly detection in Modbus/TCP traffic within industrial IoT environments. Despite using a reduced and class-balanced dataset, all models achieve high detection accuracy, confirming the suitability of graph-based learning for cybersecurity in critical infrastructures.

Among the tested architectures, GAT delivers the strongest overall performance, achieving the highest test accuracy and AUC, underscoring the value of attention mechanisms in prioritizing relevant connections within noisy IIoT traffic. GNN and GraphSAGE also achieve competitive results with simpler designs, offering efficient and interpretable alternatives for real-world deployment. By contrast, GTN shows slightly lower consistency across metrics, suggesting that its structural complexity may require larger datasets or more extensive tuning. The Multi-Head GAGNN, while conceptually promising, underperform relative to the other models, highlighting the need for further refinement of multi-attention frameworks for intrusion detection.

A notable outcome of this study is the consistently high F1 scores achieved across all graph-based models (~0.95–0.99), which not only remain stable across different architectures but also surpass those reported in prior studies. For instance, Node2Vec- and SSL-based embeddings [28] reach very high accuracy (~0.998) but much lower F1 values (0.39–0.87), while ensemble-based staged learning [27] achieve an F1 score of 0.96. In comparison, our models maintain both competitive accuracy and superior F1 performance, demonstrating a stronger balance between precision and recall under imbalanced IIoT traffic conditions.

The findings also emphasize the importance of preprocessing steps such as class imbalance handling and graph construction, which are critical to achieving robust performance. Comprehensive evaluation across accuracy, F1 score, Kappa, and ROC AUC provide a nuanced understanding of each model's strengths and limitations.

In conclusion, this research demonstrates that graph-based neural networks are highly effective for detecting anomalous activity in Modbus/TCP traffic, with attention-based mechanisms offering marginal yet consistent improvements. Future work will extend these methods to multi-protocol IIoT datasets, explore scalability in real-time deployments, and incorporate domain knowledge into graph learning to enhance interpretability and trustworthiness.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Declaration: During the preparation of this work the author(s) used large language model (ChatGPT4o) to correct grammatical errors and rephrase some sentences. After using this tool/service, the author(s) reviewed and edited the content as needed and take(s) full responsibility for the content of the publication.

References

[1] T. Ghosh, S. Bagui, S. Bagui, M. Kadzis, and J. Bare, "Anomaly detection for modbus over TCP in control systems using entropy and classification-based analysis," J. Cybersecurity Priv., vol. 3, no. 4, pp. 895–913, 2023.

[2] H. Ochiai, M. D. Hossain, P. Chirupphapa, Y. Kadobayashi, and H. Esaki, "Modbus/rs-485 attack detection on communication signals with machine learning," IEEE Commun. Mag., vol. 61, no. 6, pp. 43–49, 2023.

[3] H.-J. Liao, C.-H. R. Lin, Y.-C. Lin, and K.-Y. Tung, "Intrusion detection system: A comprehensive review," J. Netw. Comput. Appl., vol. 36, no. 1, pp. 16–24, 2013.

[4] T. Morris and W. Gao, "Industrial control system traffic data sets for intrusion detection research," presented at the Critical Infrastructure Protection VIII: 8th IFIP WG 11.10 International Conference, ICCIP 2014, Arlington, VA, USA, March 17-19, 2014, Revised Selected Papers 8, Springer, 2014, pp. 65–78.

[5] M. Aragonés Lozano, I. Pérez Llopis, and M. Esteve Domingo, "Threat hunting system for protecting critical infrastructures using a machine learning approach," Mathematics, vol. 11, no. 16, p. 3448, 2023.

[6] S. Mubarak, M. H. Habaebi, M. R. Islam, F. D. A. Rahman, and M. Tahir, "Anomaly Detection in ICS Datasets with Machine Learning Algorithms.," Comput. Syst. Sci. Eng., vol. 37, no. 1, 2021.

[7] T. Kimura et al., "Spatio-temporal factorization of log data for understanding network events," presented at the IEEE INFOCOM 2014-IEEE Conference on Computer Communications, IEEE, 2014, pp. 610–618.

[8] A. Juvonen, T. Sipola, and T. Hämäläinen, "Online anomaly detection using dimensionality reduction techniques for HTTP log analysis," Comput. Netw., vol. 91, pp. 46–56, 2015.

[9] M. Whitehouse, M. Evangelou, and N. M. Adams, "Activity-based temporal anomaly detection in enterprise-cyber security," presented at the 2016 IEEE Conference on Intelligence and Security Informatics (ISI), IEEE, 2016, pp. 248–250.

[10] C. Gates, N. Li, Z. Xu, S. N. Chari, I. Molloy, and Y. Park, "Detecting insider information theft using features from file access logs," presented at the Computer Security-ESORICS 2014: 19th European Symposium on Research in Computer Security, Wroclaw, Poland, September 7-11, 2014. Proceedings, Part II 19, Springer, 2014, pp. 383–400.

[11] L. A. Maglaras and J. Jiang, "Intrusion detection in SCADA systems using machine learning techniques," presented at the 2014 science and information conference, IEEE, 2014, pp. 626–631.

[12] Q. S. Qassim et al., "An anomaly detection technique for deception attacks in industrial control systems," presented at the 2019 IEEE 5th Intl Conference on Big Data Security on Cloud (BigDataSecurity), IEEE Intl Conference on High Performance and Smart Computing,(HPSC) and IEEE Intl Conference on Intelligent Data and Security (IDS), IEEE, 2019, pp. 267–272.

[13] B. Phillips, E. Gamess, and S. Krishnaprasad, "An evaluation of machine learning-based anomaly detection in a SCADA system using the modbus protocol," presented at the Proceedings of the 2020 ACM southeast conference, 2020, pp. 188–196.

[14] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," ArXiv Prepr. ArXiv160902907, 2016.

[15] T. N. Kipf and M. Welling, "Variational graph auto-encoders," ArXiv Prepr. ArXiv161107308, 2016.

[16] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," ArXiv Prepr. ArXiv171010903, 2017.

[17] K. Xu, W. Hu, J. Leskovec, and S. Jegelka, "How powerful are graph neural networks?," ArXiv Prepr. ArXiv181000826, 2018.

[18] R. Ying, R. He, K. Chen, P. Eksombatchai, W. L. Hamilton, and J. Leskovec, "Graph convolutional neural networks for web-scale recommender systems," presented at the Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining, 2018, pp. 974–983.

[19] H. Friji, A. Olivereau, and M. Sarkiss, "Efficient network representation for GNN-based intrusion detection," presented at the International Conference on Applied Cryptography and Network Security, Springer, 2023, pp. 532–554.

[20] J. Kim, K. Kim, G. Jeon, and M. M. Sohn, "Temporal Patterns Discovery of Evolving Graphs for Graph Neural Network (GNN)-based Anomaly Detection in Heterogeneous Networks.," J Internet Serv Inf Secur, vol. 12, no. 1, pp. 72–82, 2022.

[21] X. Dong, X. Zhang, Y. Sun, L. Chen, M. Yuan, and S. Wang, "SmoothGNN: Smoothing-aware GNN for unsupervised node anomaly detection," presented at the Proceedings of the ACM on Web Conference 2025, 2025, pp. 1225–1236.

[22] X. Ma et al., "A comprehensive survey on graph anomaly detection with deep learning," IEEE Trans. Knowl. Data Eng., vol. 35, no. 12, pp. 12012–12038, 2021.

[23] W. Marfo, D. K. Tosh, and S. V. Moore, "Enhancing network anomaly detection using graph neural networks," presented at the 2024 22nd Mediterranean Communication and Computer Networking Conference (MedComNet), IEEE, 2024, pp. 1–10.

[24] L. Zhou, Q. Zeng, and B. Li, "Hybrid anomaly detection via multihead dynamic graph attention networks for multivariate time series," IEEE Access, vol. 10, pp. 40967–40978, 2022.

[25] H. Kim, B. S. Lee, W.-Y. Shin, and S. Lim, "Graph anomaly detection with graph neural networks: Current status and challenges," IEEE Access, vol. 10, pp. 111820–111829, 2022.

[26] Hwang, Y., Kurt, F., Curebal, F., Keskin, O., & Subasi, A. (2025). Contextualgraph-Llm: A Multimodal Framework for Enhanced Darknet Traffic Analysis. Expert Systems with Applications, Volume 297, Part A, 2026, 129298,.

[27] S. Narkedimilli, S. Makam, A. V. Sriram, S. P. Mallellu, M. Sathvik, and R. R. V. Prasad, "Enhancing IoT Network Security through Adaptive Curriculum Learning and XAI," ArXiv Prepr. ArXiv250111618, 2025.

[28] Ghiasvand, E., Ray, S., Iqbal, S., Dadkhah, S., & Ghorbani, A. A. (2024). Resilience Against APTs: A Provenance-based IIoT Dataset for Cybersecurity Research. arXiv preprint arXiv:2407.11278.

[29] Yun, S., Jeong, M., Kim, R., Kang, J., & Kim, H. J. (2019). Graph transformer networks. Advances in neural information processing systems, 32.

[30] Scarselli, F., Gori, M., Tsoi, A. C., Hagenbuchner, M., & Monfardini, G. (2008). The graph neural network model. IEEE transactions on neural networks, 20(1), 61-80.