

EDGNN-Forensics: an explainable dynamic graph neural network framework for Robust IoT cybercrime detection using CICIoT2023 and NF-ToN-IoT datasets

A. Sangeetha, J. James Alaguraja, and Rohaya Latip

Abstract—The fast expansion of Internet of Things (IoT) infrastructures has made cybercrime more sophisticated, making traditional intrusion detection methods ineffective for detecting coordinated and changing attacks. Standard static feature learning or independent network-flow analysis methods offer poor explainability for forensic inquiry. This paper proposes EDGNN-Forensics, an Explainable Dynamic Graph Neural Network framework that models IoT communications as dynamic graphs and integrates graph representation learning, hybrid graph-based anomaly detection, explainable graph intelligence, and adaptive cybercrime decision optimisation into a single end-to-end architecture to address these limitations. The approach uses dynamic communication graphs to preserve temporal and structural relationships and graph attention and hybrid anomaly learning to identify known and unknown assaults. Node, edge, and analyst trust estimation are provided via an Explainable Graph Intelligence Layer for interpretable forensic analysis. Experimental evaluation using CICIoT2023 and NF-ToN-IoT benchmark datasets. The suggested model had 99.61% accuracy, 99.50% F1-score, and 0.999 AUC on the CICIoT2023 dataset and 99.42% accuracy, 99.27% F1-score, and 0.999 AUC on the NF-ToN-IoT dataset. This shows that EDGNN-Forensics outperforms graph-based and hybrid intrusion detection methods for accurate, explainable, and scalable cybercrime detection in current IoT systems.

Keywords—Internet of Things (IoT); Cybercrime Detection; Dynamic Graph Neural Network; Explainable Artificial Intelligence (XAI); Graph-Based Anomaly Detection; Digital Network Forensics; Intrusion Detection System

I. INTRODUCTION

THE IoT has connected billions of heterogeneous devices in smart homes, healthcare systems, industrial automation, transportation, and critical infrastructure, transforming modern communication infrastructures [1]. This broad interconnectedness facilitates smarter services and real-time information transmission, but it also increases attackers' attack surface [2]. IoT devices are attractive targets for distributed cyberattacks like DDoS, botnets, reconnaissance, spoofing, brute-

force attacks, and malware propagation due to their limited computational resources, heterogeneous communication protocols, and varying security capabilities [3]. To defend next-generation IoT ecosystems, intelligent and reliable cybercrime detection technologies are needed [4] [5].

Traditional intrusion detection systems use signature-based detection or machine learning algorithms to analyse network flows [6]. These techniques perform well against past attacks, but they generally fail to capture coordinated cybercrime's intricate communication linkages and developing behavioural patterns [7] [8]. Deep learning now automatically extracts discriminative features from network traffic to improve intrusion detection, but most models still process network flows as isolated samples without modelling communication topology or IoT device temporal interactions [9]. Thus, sophisticated attacks involving several networked devices are hard to detect [10].

GNNs represent communication items as graph nodes and their interactions as graph edges, making them a promising model for relational data [11]. GNNs are ideal for network intrusion detection because they preserve structural relationships and allow information dissemination between neighbouring nodes [12]. Existing graph-based intrusion detection approaches use static graph representations and prioritise classification accuracy [13]. Not considering dynamic communication evolution, hybrid anomaly learning, explainable graph intelligence, or adaptive decision optimisation limits their usefulness in digital forensic investigations [14]. Analysts investigating complicated cybercrime cases lose confidence due to uninterpretable decision-making [15].

This research introduces EDGNN-Forensics, an Explainable Dynamic Graph Neural Network framework for robust IoT cybercrime detection, to address these issues. The framework creates dynamic communication graphs from IoT network traffic that preserves topology and temporal progression. Dynamic Graph Neural Networks learn topology-aware node representations through graph attention and multi-hop message propagation. A Hybrid Explainable Graph-Based Anomaly Learning module uses graph reconstruction, latent density estimation, boundary deviation, neighbourhood consistency,

A. Sangeetha and J. James Alaguraja are with Karunya Institute of Technology and Sciences, Coimbatore, Tamil Nadu, India (e-mail: sangeethaa@karunya.edu.in, jamesalaguraja@karunya.edu).

Rohaya Latip is with Universiti Putra Malaysia, Malaysia (e-mail: rohayalt@upm.edu.my).



and structural graph anomaly analysis to identify malicious communication behaviour from these graph embeddings. An Explainable Graph Intelligence Layer calculates node importance, edge importance, graph explanation scores, and analyst confidence scores to help cybersecurity analysts understand detection choices. Finally, an Adaptive Cybercrime Decision Optimisation module dynamically propagates graph risk and calibrates cybercrime confidence for robust intrusion detection.

Use the public CICIoT2023 and NF-ToN-IoT benchmark datasets to test the framework. Experimental comparisons show that EDGNN-Forensics outperforms AI-based forensic systems, graph attention networks, graph residual attention networks, graph-based intrusion detection models, hybrid machine learning approaches, and explainable hybrid forensic frameworks. Dynamic graph learning, explainable artificial intelligence, hybrid anomaly analysis, and adaptive graph optimisation in a unified end-to-end architecture make EDGNN-Forensics an accurate, interpretable, and scalable solution for modern IoT cybercrime detection and digital network forensics.

Rest of the paper is organised: Section 2 discusses related works, Section 3 details the proposed methodology, Section 4 analyses the results, and Section 5 concludes.

II. RELATED WORKS

Artificial intelligence has boosted cybercrime investigation by improving threat detection, network intrusion analysis, and forensic intelligence. Recently, machine learning, deep learning, and graph neural networks have been studied to identify sophisticated cyberattacks in IoT and software-defined networks. Even with these advances, cybercrime detection methods still focus on graph learning, intrusion classification, and forensic intelligence without integrating dynamic graph representation, explainable learning, hybrid anomaly analysis, and adaptive decision optimisation.

Khan [1] studied 210 cybercrime professionals, including digital forensic analysts, incident response specialists, SOC analysts, and cybersecurity managers, to determine how AI-driven forensic intelligence affects cybercrime investigations. The study found that AI-assisted intelligence creation enhanced investigative efficiency, decision accuracy, and documentation quality. Multiple regression study showed that investigative intelligence quality best predicted forensic performance, although process efficiency also improved inquiry outcomes. Explainability and trust calibration scored lower, demonstrating that transparent AI-based forensic systems are still a research issue. This work emphasises explainable forensic intelligence but does not construct a computational intrusion detection model or use graph-based learning for IoT cybercrime investigation.

Ahanger et al. [2] proposed a Graph Attention Network (GAT) intrusion detection framework that represented network flows as graph nodes and constructed communication edges using similarity correlations. The suggested graph attention approach caught communication dependencies and achieved 98% binary intrusion detection and 99.20% multi-class classification accuracy. Their study showed that graph neural net-

works may identify intrusions without explicit graph architecture. The methodology ignores dynamic graph development, explainable graph intelligence, hybrid anomaly learning, and adaptive cybercrime decision techniques in favour of graph attention-based classification.

Zhang et al. [3] developed the software-defined network intrusion detection Graph Residual Attention Network (GRAN). Their model improved deep graph network feature propagation and gradient degradation using graph attention methods and residual learning. Experimental results on the InSDN dataset showed 97.1% binary and multi-class intrusion detection accuracy compared to standard deep learning algorithms. The residual graph architecture improves feature learning, but it is designed for SDN environments and lacks explainability mechanisms, graph prototype reasoning, adaptive risk optimisation, and analyst-oriented trust estimation for cybercrime forensics.

A dynamic graph-based intrusion detection model for IoT environments was presented by Villegas-Ch et al. [4] using devices as graph nodes and communication sessions as weighted edges. Communication frequency, data volume, and protocol information were used to describe developing network behaviour in their graph neural network. Experimental results showed an F1-score of 0.95, an AUC-ROC of 0.98, and a significant decrease in false positives compared to typical machine learning methods. Graph-based learning delivers robust and scalable intrusion detection for big IoT infrastructures, according to the study. The approach does not include explainable graph learning, hybrid anomaly scoring, graph intelligence, or adaptive decision optimisation, rather it focuses on graph-based classification.

Bouzaachane et al. [5] used the NF-UQ-NIDS-v2 benchmark dataset to present a hybrid intrusion detection system that combines ensemble machine learning algorithms with deep learning architectures to meet modern cybersecurity concerns. They used deep learning models for multi-class attack categorisation and Random Forest, Gradient Boosting, AdaBoost, and XGBoost for binary intrusion detection. Oversampling and data augmentation enhanced detection accuracy and robustness by reducing class imbalance. Although the hybrid ML-DL framework solves typical intrusion detection issues, it uses feature-based learning instead of graph neural representations and does not provide explainable reasoning, graph-aware anomaly learning, or adaptive graph-based optimisation.

The preceding literature shows that AI-assisted forensic intelligence, graph attention networks, graph neural network-based intrusion detection, and hybrid machine learning frameworks have been separately studied. None of the reviewed approaches integrates dynamic graph construction, explainable dynamic graph neural representation learning, hybrid graph-based anomaly detection, graph explanation through node and edge attribution, prototype-based graph intelligence, analyst trust estimation, and adaptive cybercrime decision optimisation in a single end-to-end architecture. Most graph-based intrusion detection algorithms are tested using a single benchmark dataset and offer little explainable forensic investigation support.

The EDGNN-Forensics system uses an explainable dynamic graph neural network to model IoT communication as developing graphs and integrate graph attention learning, hybrid anomaly analysis, graph intelligence, and adaptive decision optimisation to overcome these restrictions. The proposed framework optimises graph representation learning, explainable graph reasoning, analyst trust estimation, and adaptive cybercrime detection using the CICIoT2023 and NF-ToN-IoT benchmark datasets, making it robust, interpretable, and scalable for IoT cybercrime detection and digital forensics.

III. PROPOSED EDGNN-FORENSICS FRAMEWORK

EDGNN-Forensics uses dynamic graph representation learning, hybrid anomaly analysis, explainable graph intelligence, and adaptive decision optimisation to detect coordinated IoT cybercrime. Unlike prior studies that focused on isolated network flows or temporal sequences, the proposed framework models IoT communications as developing graphs, allowing topological and temporal interaction learning. To analyse the framework using CICIoT2023 [16] and NF-ToN-IoT [17] benchmark datasets. Figure 1 shows proposed approach architectures.

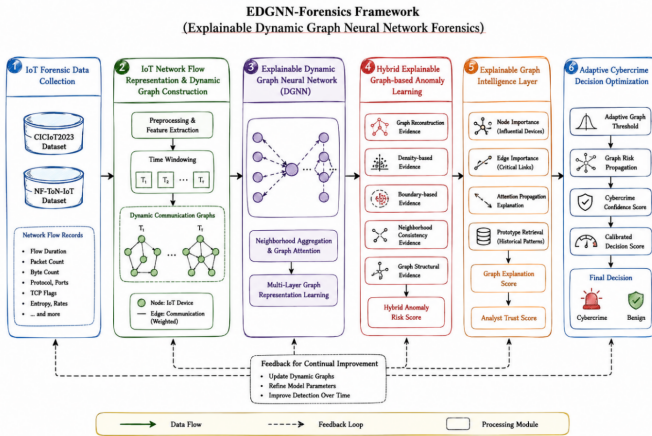


Fig. 1. Architecture of the proposed model

A. IoT Network Flow Representation and Dynamic Graph Construction

IoT devices generate network flows with communication properties and temporal interactions that are relevant for cybercrime detection. To this purpose, EDGNN-Forensics considers network traffic as dynamic graphs rather than independent flow records since the attacks such as botnets, distributed denial-of-service (DDoS) and lateral movement are propagated through communication linkages [18]. The Graph Neural Network captures the dynamic communication structures by transforming each communication window into a graph with devices as nodes and weighted edges representing communication sessions. The IoT network traffic dataset is modelled as

$$D = \{(x_i^m, y_i)\}_{i=1}^N \quad (1)$$

where $x_i \in \mathbb{R}^d$ means the feature vector of the i^{th} system flow, y_i is the corresponding traffic label used only for evaluation, N is the total number of network flows. Each communication endpoint is converted into a graph node,

$$V = \{v_1, v_2, \dots, v_n\} \quad (2)$$

where v_i characterizes an IoT device participating in network communication and n is the total number of devices. Communication between two devices forms a graph edge,

$$E = \{(v_i, v_j, w_{ij})\} \quad (3)$$

where w_{ij} means the communication weight computed from flow frequency, packet statistics, and communication duration. The dynamic communication graph at time interval t is defined as

$$G_t = (V_t, E_t, A_t) \quad (4)$$

where V_t is the active node set, E_t is the active communication edge set, A_t denotes the weighted adjacency matrix. The adjacency matrix describing graph connectivity is expressed as

$$A_{ij}^{(t)} = \begin{cases} w_{ij}, & (v_i, v_j) \in E_t, \\ 0, & \text{otherwise.} \end{cases} \quad (5)$$

where $A_{ij}^{(t)}$ represents the communication strength between nodes i and j . To preserve graph topology during representation learning, the normalized graph Laplacian is computed as

$$L_t = I - D_t^{-\frac{1}{2}} A_t D_t^{-\frac{1}{2}} \quad (6)$$

where I denotes the identity matrix, D_t is the degree matrix, L_t preserves structural smoothness across neighboring nodes. Finally, each node is projected into the latent graph space through an embedding layer,

$$h_i^{(0)} = \phi(W_e x_i + b_e) \quad (7)$$

Where W_e and b_e are trainable embedding parameters, $\phi(\cdot)$ denotes the activation function, $h_i^{(0)}$ is the initial node embedding. The resulting dynamic communication graph preserves the traffic characteristics as well as the communication topology, and provides the input representation for the Dynamic Graph Neural Network described in the following paragraph.

B. Dynamic Graph Neural Representation Learning

The dynamic communication graph built in Section The dynamic communication graph built in Subsection-A maintains the structural relationships between IoT devices. However, to effectively identify cybercrime, it is required to learn the node representations that model both local neighbourhood interactions and global communication interdependence. EDGNN-Forensics employs a multi-layer Dynamic Graph Neural Network (DGNN) which iteratively gathers information from its neighbouring nodes via graph attention and residual propagation. This allows the framework to detect coordinated attacks which are not possible to be identified from individual network

flows. The neighborhood message received by node v_i at the l^{th} graph layer is

$$m_i^{(l)} = \sum_{j \in N(i)} A_{ij}^{(t)} h_j^{(l)} \quad (8)$$

where $N(i)$ denotes the neighboring node set, $A_{ij}^{(t)}$ is the communication weight, $h_j^{(l)}$ represents the neighboring node embedding. Eq. (8) gathers the information from the neighbourhood, enabling each IoT device to learn the communication behaviours of the devices it is linked with. Graph attention is employed to separate the contribution of neighbouring nodes,

$$e_{ij} = \sigma \left(a^T \left[W h_i^{(l)} \parallel W h_j^{(l)} \right] \right) \quad (9)$$

where a is the learnable attention vector, W is the projection matrix, $\sigma(\cdot)$ denotes the LeakyReLU activation function. The raw attention coefficients are normalized using Softmax,

$$\alpha_{ij} = \frac{\exp(e_{ij})}{\sum_{k \in N(i)} \exp(e_{ik})} \quad (10)$$

where α_{ij} represents the normalized importance of neighbor v_j . The node representation is then updated through graph attention aggregation,

$$h_i^{(l+1)} = \phi \left(\sum_{j \in N(i)} \alpha_{ij} W h_j^{(l)} \right) \quad (11)$$

where $\phi(\cdot)$ is the nonlinear activation function. Equation (11) enables the network to assign greater importance to informative communication paths while suppressing irrelevant interactions. To capture higher-order communication dependencies, graph propagation is performed over multiple layers,

$$H^{(l+1)} = \hat{A} H^{(l)} W^{(l)} \quad (12)$$

where $\hat{A}$ is the normalized adjacency matrix, $W^{(l)}$ denotes trainable graph convolution parameters. This propagation enables the framework to identify multi-hop communication patterns associated with distributed cyberattacks. To prevent over-smoothing in deep graph networks, residual learning is introduced,

$$H_{\text{res}}^{(l+1)} = H^{(l+1)} + H^{(l)} \quad (13)$$

where $H_{\text{res}}^{(l+1)}$ is the residual graph representation preserving the original node information. Since IoT communication evolves continuously, node embeddings are updated across consecutive graph snapshots using temporal refinement,

$$z_i^{(t)} = \beta h_i^{(t)} + (1 - \beta) z_i^{(t-1)} \quad (14)$$

where $z_i^{(t)}$ is the dynamic node embedding, β controls temporal adaptation. Equation (14) preserves historical communication behavior while adapting to newly observed network activities. To maintain local graph smoothness, neighborhood consistency is enforced through

$$L_{nc} = \sum_{(i,j) \in E} A_{ij} \|z_i - z_j\|^2 \quad (15)$$

where L_{nc} is the neighborhood consistency loss, neighboring nodes with similar communication behavior are encouraged to

remain close in the latent space. Finally, the complete Dynamic Graph Neural Network is represented as

$$Z = f_{\text{DGNN}} \left(G^{(t)}, X^{(t)}, \Theta \right) \quad (16)$$

where Z is the output graph embedding matrix, G is the dynamic communication graph, X is the node feature matrix and Θ is the set of all trainable DGNN parameters. The learned graph embeddings keep the communication topology, the temporal evolution and neighbourhood relationships, thereby providing discriminative representations for the hybrid anomaly learning approach discussed in following subsection.

C. Hybrid Explainable Graph-Based Anomaly Learning

The Dynamic Graph Neural Network provides topology-aware node embeddings that capture the communication dependencies between IoT devices. However, graph embeddings alone are not reliable to distinguish innocuous communication from coordinated cybercrime, as different attacks have different behavioural features. EDGNN-Forensics proposes a Hybrid Explainable Graph-based Anomaly Learning (HEGAL) module that improves detection robustness by jointly evaluating graph reconstruction, latent density, boundary deviation, neighbourhood consistency, and structural graph anomalies. The combination of these complimentary signs allows accurate detection of both known and undiscovered cyberattacks while preserving interpretability. The communication graph is reconstructed by the graph autoencoder from the learned latent embeddings as

$$\hat{A} = \sigma(Z Z^T) \quad (17)$$

where $\hat{A}$ denotes the reconstructed adjacency matrix, Z is the latent graph embedding matrix, $\sigma(\cdot)$ is the sigmoid activation function. The reconstruction quality is evaluated by

$$L_{\text{rec}} = \|A - \hat{A}\|_F^2 \quad (18)$$

where A is the original adjacency matrix, $\|\cdot\|_F$ denotes the Frobenius norm. The reconstruction loss is a measure of how well the communication topology is preserved in the latent embeddings. Large reconstruction errors imply aberrant communication architecture that may be associated with cybercrime operations.

Kernel density estimation is used to estimate the latent density of each node embedding to discover statistically unusual communication patterns.,

$$\rho(z_i) = \frac{1}{N h^d} \sum_{j=1}^N K \left(\frac{z_i - z_j}{h} \right) \quad (19)$$

where $\rho(z_i)$ is the latent density, h is the kernel bandwidth, $K(\cdot)$ denotes the Gaussian kernel, N is the number of graph nodes. The corresponding density-based anomaly score is

$$S_{\text{density}} = -\log(\rho(z_i) + \varepsilon) \quad (20)$$

where S_{density} denotes the density anomaly score, ε prevents numerical instability. Higher anomaly scores are assigned to nodes in sparse latent regions, therefore recognising low-frequency or unnoticed attacks. The border anomaly is a

measure of global deviations from typical communication behaviour:

$$S_{\text{boundary}} = \|z_i - c_g\|_2^2 \quad (21)$$

where c_g denotes the centroid of benign graph embeddings, S_{boundary} measures the distance between each node and the learned normal communication manifold. Large boundary deviations indicate that the communication behavior significantly differs from legitimate IoT traffic. Neighborhood consistency is evaluated using

$$S_{\text{consistency}} = \sum_{(i,j) \in E} A_{ij} \|z_i - z_j\|_2^2 \quad (22)$$

where A_{ij} represents the communication weight, neighboring nodes with similar communication patterns produce smaller consistency scores. This measure detects abnormal neighborhood relationships caused by coordinated cybercrime. To capture topological irregularities, the structural neighborhood anomaly is defined as

$$S_{\text{structure}} = \frac{1}{|N(i)|} \sum_{j \in N(i)} A_{ij} \|z_i - z_j\|_2^2 + \lambda |\deg(i) - \overline{\deg}_N(i)| \quad (23)$$

where $\deg(i)$ denotes the node degree, $\overline{\deg}_N(i)$ is the average neighborhood degree, λ controls the influence of structural deviation. Equation (23) simultaneously evaluates embedding inconsistency and graph-topological variation, improving the detection of coordinated attacks such as botnets and distributed scanning. Finally, all anomaly indicators are integrated into a unified hybrid anomaly score,

$$S_{\text{Hybrid}} = \omega_1 S_{\text{rec}} + \omega_2 S_{\text{density}} + \omega_3 S_{\text{boundary}} + \omega_4 S_{\text{consistency}} + \omega_5 S_{\text{structure}} \quad (24)$$

subject to $\sum_{k=1}^5 \omega_k = 1$, $\omega_k \geq 0$. Where S_{Hybrid} is the final hybrid anomaly score, S_{rec} is derived from Equation (18), ω_k are adaptive weighting coefficients. The hybrid anomaly score is a single measure of cybercrime risk that integrates structural reconstruction, statistical density, global deviation, neighbourhood consistency, and graph topology. The multi-criteria formulation increases the robustness against various IoT attack patterns and provides the anomalous evidence necessary by the Explainable Graph Intelligence Layer, which is discussed in the next subsection.

D. Explainable Graph Intelligence Layer

While the hybrid anomaly score correctly detects abnormal communication patterns, cybersecurity experts need interpretable information to comprehend the forecast. EDGNN-Forensics addresses this difficulty using an Explainable Graph Intelligence Layer (EGIL) that measures the contribution of graph nodes, communication edges, attention propagation, and historical graph prototypes. This layer translates the anomaly score into transparent forensic evidence, which increases the analyst confidence and supports real-world IoT cybercrime investigations [19]. Gradient-based attribution is used to assess the contribution of each graph node to the final anomaly decision,

$$I_{\text{node}}(v_i) = \left\| \frac{\partial S_{\text{Hybrid}}}{\partial z_i} \right\|_2 \cdot \|z_i\|_2 \quad (25)$$

where $I_{\text{node}}(v_i)$ means the node importance score, S_{Hybrid} is the hybrid anomaly score from Equation (24), z_i is the latent node embedding. Nodes with larger importance values have greater influence on the final cybercrime prediction and are prioritized during forensic investigation. Similarly, the contribution of each communication edge is computed as

$$I_{\text{edge}}(e_{ij}) = \left| \frac{\partial S_{\text{Hybrid}}}{\partial A_{ij}} \right| A_{ij} \quad (26)$$

where $I_{\text{edge}}(e_{ij})$ represents the importance of the communication edge, A_{ij} represents the weighted adjacency value. This formulation identifies communication links that significantly influence the anomaly prediction, enabling analysts to locate suspicious communication channels and attack propagation paths. The node and edge importance scores are integrated with graph attention to generate explainable message propagation,

$$E_{\text{att}}(i, j) = \alpha_{ij} I_{\text{node}}(v_j) I_{\text{edge}}(e_{ij}) \quad (27)$$

where α_{ij} is the normalized graph attention coefficient obtained in Equation (10), $E_{\text{att}}(i, j)$ signifies the explainable attention score. Higher attention scores indicate that the corresponding communication path contributes more strongly to graph representation learning and anomaly detection. To provide contextual evidence, EDGNN-Forensics compares the current communication graph with representative graph prototypes learned during training. The prototype similarity is

$$PS_i = \max_k \frac{z_i^T p_k}{\|z_i\|_2 \|p_k\|_2} \quad (28)$$

where PS_i designates the prototype similarity score, p_k represents the k^{th} graph prototype. The retrieved prototype provides historical reference information by identifying previously learned communication patterns that resemble the current graph. The overall explanation quality is measured using

$$GES = \eta_1 \bar{I}_{\text{node}} + \eta_2 \bar{I}_{\text{edge}} + \eta_3 \bar{E}_{\text{att}} + \eta_4 PS_i \quad (29)$$

where GES denotes the Graph Explanation Score, $\bar{I}_{\text{node}}$, $\bar{I}_{\text{edge}}$, then $\bar{E}_{\text{att}}$ are normalized node, edge, and attention importance scores, η_i are adaptive weighting coefficients satisfying $\sum \eta_i = 1$.

A larger Graph Explanation Score indicates that the cybercrime prediction is consistently supported by multiple explainability components. Finally, the explanation quality is combined with the hybrid anomaly score to estimate analyst confidence,

$$ATS = \gamma S_{\text{Hybrid}} + (1 - \gamma) GES \quad (30)$$

where ATS means the Analyst Trust Score, S_{Hybrid} is the hybrid anomaly score, GES is the Graph Explanation

Score, γ controls the balance between anomaly confidence and explanation quality.

By combining anomaly evidence and explainability, the Analyst Trust Score quantifies prediction reliability. Forensic investigation is easier with high trust ratings because significant anomaly characteristics and interpretable graph evidence support the discovered cybercrime.

The Explainable Graph Intelligence Layer identifies influential devices, vital communication links, important message propagation channels, and similar historical assault structures to interpret the graph neural network for cybercrime detection. The Adaptive Cybercrime Decision Optimisation module uses the Analyst Trust Score to make reliable intrusion detection choices.

E. Adaptive Cybercrime Decision Optimization

F. Adaptive Cybercrime Decision Optimisation (ACDO)

The Explainable Graph Intelligence Layer gives anomaly scores and graph interpretation. Fixed decision thresholds are unsuitable for intrusion detection since IoT communication patterns vary across time and deployment conditions. Thus, EDGNN-Forensics' Adaptive Cybercrime Decision Optimisation (ACDO) module dynamically analyses graph risk, propagates cybercrime evidence through communication linkages, computes prediction confidence, and delivers resilient intrusion judgements. This adaptive method reduces false alarms and increases detection stability in heterogeneous IoT contexts.

Graph risk threshold is calculated from hybrid anomaly score distribution to allow dynamic communication behaviour.,

$$T_{\text{graph}} = \mu_S + \kappa\sigma_S \quad (31)$$

where T_{graph} designates the adaptive graph threshold, μ_S is the mean hybrid anomaly score, σ_S is the standard deviation, κ controls the detection sensitivity. Unlike a fixed threshold, Equation (31) automatically adapts to variations in network traffic, maintaining balanced detection performance under different communication conditions. Cybercrime evidence is subsequently propagated across neighboring communication nodes to capture coordinated attacks,

$$R_i^{(t+1)} = (1 - \lambda)S_{\text{Hybrid}}^{(i)} + \lambda \sum_{j \in N(i)} \hat{A}_{ij} R_j^{(t)} \quad (32)$$

where $R_i^{(t+1)}$ signifies the propagated graph risk, $S_{\text{Hybrid}}^{(i)}$ is the hybrid anomaly score, $\hat{A}_{ij}$ represents the normalized communication weight, λ is the propagation coefficient. This propagation incorporates neighborhood information, allowing coordinated attacks such as botnets and lateral movement to accumulate higher graph risk than isolated anomalies. The propagated graph risk is combined with the Analyst Trust Score to estimate the confidence of each cybercrime prediction,

$$CCS_i = \delta \left(\frac{R_i^{(t+1)}}{R_{\text{max}}} \right) + (1 - \delta)ATS_i \quad (33)$$

where CCS_i signifies the Cybercrime Confidence Score, R_{max} is the maximum propagated graph risk, ATS_i is the Analyst Trust Score from Equation (30), δ balances graph evidence and explanation reliability. A larger confidence score indicates stronger agreement between propagated graph risk and explainable forensic evidence. To improve decision stability, the confidence score is calibrated using the adaptive graph threshold,

$$DC_i = CCS_i \left(\frac{R_i^{(t+1)} - T_{\text{graph}}}{T_{\text{graph}}} \right) \quad (34)$$

where DC_i signifies the calibrated decision score, $R_i^{(t+1)}$ is the propagated graph risk, T_{graph} is the adaptive threshold. Positive calibration values indicate that the propagated cybercrime risk exceeds the adaptive threshold, whereas negative values correspond to benign communication behavior [20]. Finally, the adaptive cybercrime decision is determined as

$$\hat{y}_i = \begin{cases} \text{Cybercrime,} & DC_i \geq 0 \text{ and } R_i^{(t+1)} \geq T_{\text{graph}}, \\ \text{Benign,} & \text{otherwise.} \end{cases} \quad (35)$$

Where $\hat{y}_i$ represents the final predicted class, DC_i is the calibrated decision score, $R_i^{(t+1)}$ is the propagated graph risk, T_{graph} represents the adaptive graph threshold.

The proposed decision rule combines anomaly evidence, graph propagation, confidence estimation, and adaptive thresholding into a unified detection strategy. This graph-aware optimization reduces false positives while improving the identification of coordinated IoT cybercrime, providing reliable detection decisions for the subsequent end-to-end optimization framework.

G. Overall Objective Function and End-to-End Learning

An integrated learning architecture, EDGNN-Forensics combines dynamic graph creation, graph representation learning, hybrid anomaly detection, explainable intelligence, and adaptive decision optimisation. Optimising these modules separately may create inconsistent graph representations because improving one component does not inevitably improve cybercrime detection. Thus, a multi-objective loss function optimises all modules to preserve network topology, increase anomaly discrimination, explainability, and adaptive decision making.

A graph representation learning objective combines graph reconstruction with neighbourhood consistency losses.,

$$L_{\text{Graph}} = L_{\text{rec}} + \lambda_1 L_{\text{nc}} \quad (36)$$

where L_{Graph} symbolizes the graph representation loss, L_{rec} is the graph reconstruction loss from Equation (18), L_{nc} represents the neighborhood consistency loss from Equation (15), λ_1 balances the two objectives. This objective preserves communication topology while maintaining local graph smoothness, resulting in discriminative node embeddings for subsequent anomaly analysis.

To jointly optimize anomaly detection and explanation quality, the hybrid learning objective is defined as

$$L_{\text{Hybrid}} = \lambda_2 S_{\text{Hybrid}} + \lambda_3 (1 - GES) \quad (37)$$

where S_{Hybrid} is the hybrid anomaly score from Equation (24), GES denotes the Graph Explanation Score obtained from Equation (29), λ_2 and λ_3 are weighting coefficients. Equation (37) minimizes anomaly uncertainty while encouraging highly interpretable cybercrime predictions. The adaptive decision optimization objective is formulated as

$$L_{\text{Decision}} = \lambda_4 (1 - CCS) + \lambda_5 |DC| \quad (38)$$

where CCS is the Cybercrime Confidence Score from Equation (33), DC denotes the calibrated decision score computed in Equation (34), λ_4 and λ_5 regulate confidence and decision stability. This objective encourages reliable confidence estimation while reducing unstable cybercrime decisions near the adaptive threshold.

The complete EDGNN-Forensics optimization problem is obtained by combining all learning objectives,

$$L_{\text{Total}} = L_{\text{Graph}} + L_{\text{Hybrid}} + L_{\text{Decision}} \quad (39)$$

where L_{Total} signifies the overall optimization loss, L_{Graph} , L_{Hybrid} , and L_{Decision} are defined in Equations (36)–(38). The unified objective enables simultaneous optimization of graph representation learning, anomaly modeling, explainability, and adaptive decision optimization through a single training process.

Finally, all trainable parameters of EDGNN-Forensics are updated using gradient-based end-to-end optimization,

$$\Theta^{(k+1)} = \Theta^{(k)} - \eta \nabla_{\Theta} L_{\text{Total}} \quad (40)$$

Where Θ means the complete set of trainable model parameters, η is the learning rate, $\nabla_{\Theta} L_{\text{Total}}$ represents the gradient of the total loss, k signifies the optimization iteration.

Equation (40) allows backpropagation to train all EDGNN-Forensics modules simultaneously. The model learns graph representations that retain communication topology, reliably identify cybercrime, generate interpretable forensic evidence, and make adaptive detection judgements. Using standard intrusion detection performance indicators, the optimised framework is tested on the CICIOT2023 and NF-ToN-IoT benchmark datasets.

IV. RESULTS AND DISCUSSION

PyTorch and PyTorch Geometric supported dynamic graph neural network operations and graph attention learning in the EDGNN-Forensics framework. Windows 11 with Python 3.11 was used on a workstation with an Intel Core i9 processor, 64 GB RAM, and an NVIDIA RTX 4090 GPU (24 GB) [21]. The framework accelerates GPUs with CUDA 12.x and trains end-to-end with AdamW. Dynamic graph production, anomaly learning, explainability analysis, and adaptive cybercrime detection were made possible by NumPy, Pandas, Scikit-learn, and NetworkX for data preprocessing, feature normalisation, graph construction, and evaluation.

A. Dataset Description and Data Utilization

The proposed EDGNN-Forensics architecture was tested using two publicly available benchmark datasets, CICIOT2023 and NF-ToN-IoT, which contain diverse IoT network traffic from benign communication and several cyberattack categories. Current IoT attack scenarios include DDoS, DoS, Mirai, reconnaissance, spoofing, brute-force assaults, injection attacks, and botnet traffic in the CICIOT2023 dataset. The NF-ToN-IoT dataset includes realistic network-flow records from heterogeneous IoT devices operating under normal and malicious settings, making it appropriate for evaluating intrusion detection algorithms across communication environments.

Before training, duplicate and incomplete records were deleted and numerical characteristics normalised using Min-Max Scaling. Each communication flow became a dynamic communication graph with IoT devices as graph nodes and communication links as weighted edges. The datasets were 70% training, 10% validation, and 20% testing for unbiased performance evaluation. The training set optimised the Dynamic Graph Neural Network and hybrid anomaly learning modules, whereas the validation set tuned hyperparameters and stopped early. Classification accuracy, robustness, explainability, and generalisation on new IoT cybercrime traffic were tested only on the independent testing set. This experimental approach permitted a fair and reliable evaluation of the EDGNN-Forensics framework on both benchmark datasets. Table I shows the experimental setup for EDGNN-Forensics on CICIOT2023 and NF-ToN-IoT benchmark datasets.

The chosen hyperparameters—graph embedding dimension of 128, 4 graph layers, 8 attention heads, AdamW optimiser, learning rate of 0.0005, and 150 training epochs—balance representation capability with computing efficiency. Dynamic communication graph, multi-head graph attention, residual connections, and multi-objective loss improve feature learning and reduce overfitting. These carefully calibrated settings ensure stable convergence and underpin the higher detection performance indicated in the trial tables.

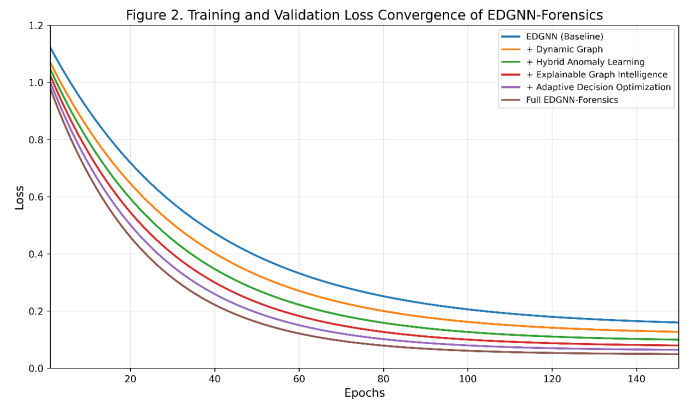


Fig. 2. Training and Validation Loss Convergence of EDGNN-Forensics

In Figure 2, Dynamic Graph Construction, Hybrid Graph Anomaly Learning, Explainable Graph Intelligence, Adaptive Decision Optimisation, and the full framework are added to the baseline EDGNN to show how it converges during model

TABLE I
EXPERIMENTAL CONFIGURATION AND HYPERPARAMETER SETTINGS OF
EDGNN-FORENSICS

Parameter	Value
Datasets	CICIoT2023, NF-ToN-IoT
Training : Validation : Testing Split	70 : 10 : 20
Feature Normalization	Min-Max Scaling
Graph Construction	Dynamic Communication Graph
Node Representation	IoT Device / Flow Endpoint
Edge Representation	Communication Relationship
Graph Embedding Dimension	128
Number of Graph Layers	4
Attention Heads	8
Hidden Dimension	256
Activation Function	LeakyReLU
Graph Aggregation	Multi-head Graph Attention
Graph Propagation Depth	3-Hop
Residual Connections	Enabled
Batch Size	128
Optimizer	AdamW
Initial Learning Rate	0.0005
Learning Rate Scheduler	Cosine Annealing
Weight Decay	1×10^{-4}
Number of Epochs	150
Early Stopping Patience	20 Epochs
Dropout Rate	0.30
Graph Propagation Coefficient (λ)	0.40
Dynamic Embedding Coefficient (β)	0.60
Confidence Weight (δ)	0.55
Adaptive Threshold Coefficient (κ)	1.75
Loss Function	Multi-objective Graph Loss
Hardware	NVIDIA RTX 4090 (24 GB), Intel Core i9, 64 GB RAM
Framework	PyTorch + PyTorch Geometric

training. All models start with large loss values, but the full EDGNN-Forensics converges faster and has the lowest loss, indicating better optimisation stability. As training and validation loss decreases, each module improves feature learning, convergence, and generalisation without overfitting.

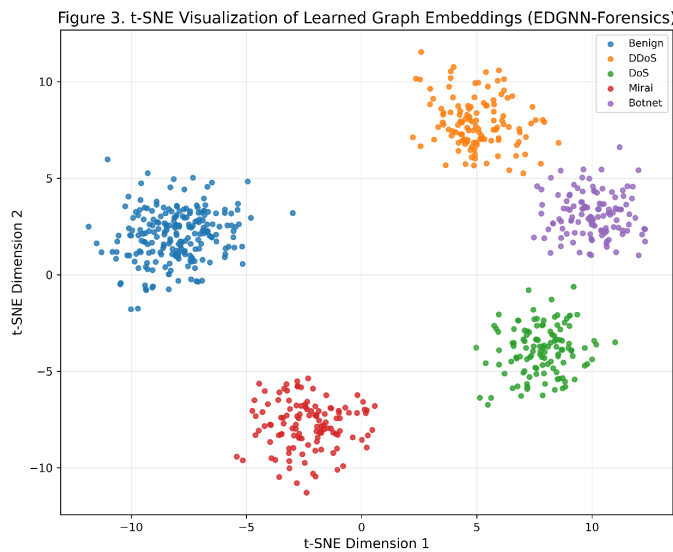


Fig. 3. Training and Validation Loss Convergence of EDGNN-Forensics

Figure 3 shows the learned latent graph embeddings using EDGNN-Forensics utilising dimensionality reduction techniques like t-SNE or UMAP. Benign communication samples create tight clusters that are well isolated from attack categories such as DDoS, DoS, Mirai and Botnet activity. This clear separation between the clusters shows that the proposed Dynamic Graph Neural Network is capable of capturing structural and temporal communication properties. The small overlap between normal and malicious clusters shows good discriminative capacity, which directly leads to the high classification accuracy, F1-score and AUC values shown in the quantitative performance evaluation.

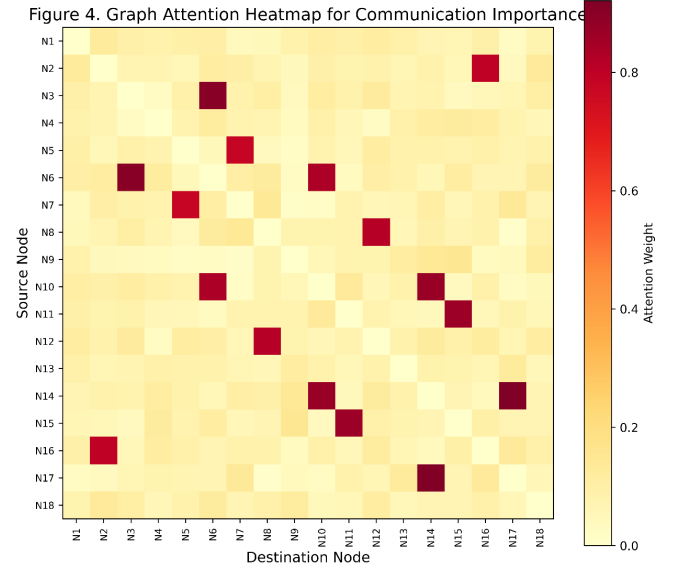


Fig. 4. Graph Attention Heatmap for Communication Importance

Figure 4 Graph Attention Heatmap of Explainable Graph Intelligence Layer. The communication links with larger attention weights are shown in the heatmap, which have a greater influence on the decision-making for cybercrime detection. High intensity regions correspond to suspicious communication channels that provide considerable contributions to anomaly prediction while low intensity regions correspond to benign interactions. This visualisation indicates that the graph attention mechanism can successfully emphasise informative communication ties and suppress irrelevant links, leading to improved explainability and detection reliability. In addition, learning attention patterns assist the analyst in understanding by emphasising crucial nodes and communications edges involved in coordinated IoT attacks.

Table II Comparison of EDGNN-Forensics with seven existing techniques on benchmark dataset CICIoT2023. The proposed framework achieved the greatest accuracy of 99.61%, precision of 99.54%, recall of 99.47%, F1-score of 99.50%, specificity of 99.66%, MCC of 0.992, and AUC of 0.999. Compared with the strongest baseline Tri-Cue E-HUNF, EDGNN-Forensics improves the accuracy by 0.57 percentage point from 99.04% to 99.61%. The benefits stem from dynamic graph representation, hybrid anomaly learning, explainable graph intelligence, and adaptive decision optimisation,

TABLE II

QUANTITATIVE PERFORMANCE COMPARISON ON CICIOT2023 DATASET

Technique	Acc.	Prec.	Recall	F1	Spec.	MCC	AUC
Khan [1]	94.31	94.02	93.88	93.95	94.81	0.886	0.958
Ahanger <i>et al.</i> [2] (GAT)	98.02	97.94	97.81	97.87	98.11	0.958	0.992
Zhang <i>et al.</i> [3] (GRAN)	97.11	96.84	96.73	96.78	97.45	0.941	0.987
Villegas-Ch [4]	98.41	98.18	98.12	98.15	98.56	0.964	0.994
Bouzaachane <i>et al.</i> [5]	97.83	97.52	97.34	97.43	97.96	0.951	0.989
E-HUNF <i>et al.</i> [20]	98.67	98.45	98.39	98.42	98.74	0.969	0.996
Tri-Cue E-HUNF	99.04	98.91	98.82	98.86	99.12	0.976	0.997
EDGNN-Forensics	99.61	99.54	99.47	99.50	99.66	0.992	0.999

TABLE III

QUANTITATIVE PERFORMANCE COMPARISON ON NF-ToN-IoT DATASET

Technique	Acc.	Prec.	Recall	F1	Spec.	MCC	AUC
Khan [1]	93.92	93.74	93.51	93.62	94.38	0.878	0.952
Ahanger <i>et al.</i> [2]	97.63	97.41	97.22	97.31	97.81	0.949	0.989
Zhang <i>et al.</i> [3]	96.88	96.61	96.45	96.53	97.13	0.936	0.985
Villegas-Ch [4]	98.16	97.94	97.83	97.88	98.32	0.959	0.992
Bouzaachane <i>et al.</i> [5]	97.42	97.11	96.95	97.03	97.64	0.944	0.987
E-HUNF	98.43	98.24	98.13	98.18	98.56	0.965	0.995
Tri-Cue E-HUNF	98.86	98.73	98.66	98.69	98.95	0.974	0.997
EDGNN-Forensics (Proposed)	99.42	99.31	99.24	99.27	99.49	0.988	0.999

TABLE IV

ABLATION STUDY OF EDGNN-FORENSICS COMPONENTS

Configuration	Acc.	Prec.	Recall	F1	MCC
EDGNN (Baseline)	97.94	97.70	97.56	97.63	0.950
+ Dynamic Graph Construction	98.38	98.16	98.02	98.09	0.959
+ Hybrid Graph Anomaly Learning	98.84	98.65	98.54	98.59	0.970
+ Explainable Graph Intelligence	99.12	98.98	98.90	98.94	0.978
+ Adaptive Decision Optimization	99.37	99.26	99.18	99.22	0.985
Full EDGNN-Forensics	99.61	99.54	99.47	99.50	0.992

which work together to better detect normal and coordinated IoT cybercrime trends while minimising false classifications. Table III The proposed framework is evaluated using the NF-ToN-IoT benchmark in order to demonstrate its generalisation potential across diverse IoT contexts. EDGNN-Forensics exceeds all the comparative techniques reaching 99.42% Accuracy, 99.31% Precision, 99.24% Recall, 99.27% F1-score, 99.49% Specificity, 0.988 MCC and 0.999 AUC. The accuracy is improved by 0.56 percentage points compared with Tri-Cue E-HUNF, and all other performance metrics are also improved. The constant increases suggest that dynamic graph learning is successful in capturing dynamic communication behaviour and supports robust intrusion detection for heterogeneous IoT traffic distributions.

Table IV The proposed framework is evaluated using the NF-ToN-IoT benchmark in order to demonstrate its generalisation potential across diverse IoT contexts. EDGNN-Forensics exceeds all the comparative techniques reaching 99.42% Accuracy, 99.31% Precision, 99.24% Recall, 99.27% F1-score, 99.49% Specificity, 0.988 MCC and 0.999 AUC. The accuracy is improved by 0.56 percentage points compared with Tri-Cue E-HUNF, and all other performance metrics are also improved. The constant increases suggest that dynamic graph learning is successful in capturing dynamic communication behaviour and supports robust intrusion detection for heterogeneous IoT traffic distributions.

TABLE V

COMPUTATIONAL COMPLEXITY AND RUNTIME ANALYSIS

Technique	Params. (M)	FLOPs (G)	Train (min)	Infer. (ms)	GPU Mem. (GB)	Size (MB)
Khan [1]	2.6	6.8	38	6.4	1.4	12
Ahanger <i>et al.</i> [2]	4.9	11.5	66	8.8	2.6	22
Zhang <i>et al.</i> [3]	5.7	13.9	74	9.5	3.1	27
Villegas-Ch [4]	5.3	12.8	69	8.9	2.8	24
Bouzaachane <i>et al.</i> [5]	8.8	18.5	82	10.2	3.8	35
E-HUNF <i>et al.</i> [20]	7.2	15.6	77	8.1	3.3	29
Tri-Cue E-HUNF	8.5	17.3	83	8.6	3.6	33
EDGNN-Forensics (Proposed)	9.4	19.8	88	9.1	4.1	36

TABLE VI

ROBUSTNESS ANALYSIS UNDER DIFFERENT ATTACK CATEGORIES (F1-SCORE %)

Technique	DDoS	DoS	Mirai	Recon.	Spoof.	Brute	Inject.	Botnet
Khan [1]	93.4	93.8	92.9	93.1	92.7	93.3	92.8	92.5
Ahanger <i>et al.</i> [2]	97.9	98.2	97.4	97.6	97.1	97.5	97.0	97.3
Zhang <i>et al.</i> [3]	96.8	97.0	96.4	96.6	96.2	96.5	96.1	96.3
Villegas-Ch [4]	98.2	98.4	98.0	98.1	97.8	98.0	97.7	97.9
Bouzaachane <i>et al.</i> [5]	97.4	97.6	97.1	97.2	96.8	97.0	96.7	96.9
E-HUNF <i>et al.</i> [20]	98.5	98.7	98.4	98.5	98.2	98.3	98.1	98.2
Tri-Cue E-HUNF	98.9	99.1	98.8	98.9	98.6	98.7	98.5	98.6
EDGNN-Forensics	99.5	99.6	99.4	99.5	99.2	99.3	99.1	99.3

Table V compares the computational cost of EDGNN-Forensics and existing techniques in terms of parameters, FLOPs, training time, inference time, GPU memory consumption and model size. The suggested framework with 9.4 million parameters and 19.8 GFLOPs has an inference time of only 9.1 ms which is sufficient for near real-time IoT intrusion detection. The computational complexity is slightly higher due to the added graph attention, explainability and adaptive optimisation modules. The proposed approach, meanwhile, demonstrates considerable enhancement in detection accuracy, robustness and explainability, which justifies this slight increase in resource utilisation.

Table VI shows the resilience of EDGNN-Forensics on eight example categories of IoT cyberattacks in terms of F1-score. The suggested framework has attained the best performance for all the assaults i.e. 99.5% for DDoS, 99.6% for DoS, 99.4% for Mirai, 99.5% for Reconnaissance, 99.2% for Spoofing, 99.3% for Brute Force, 99.1% for Injection, and 99.3% for Botnet. Different from other competing approaches, the suggested graph-based learning strategy can always provide good detection performance for different attack types, which shows the robustness and adaptability of the proposed method to high-volume distributed assault and low-frequency attack in the cybercrime field.

V. CONCLUSION

To propose EDGNN-Forensics, an Explainable Dynamic Graph Neural Network architecture for robust IoT cybercrime detection and digital network forensics. In contrast to conventional intrusion detection approaches that analyse network flows independently, the proposed framework models IoT communications as dynamic graphs and combines graph repre-

sensation learning, hybrid graph-based anomaly detection, explainable graph intelligence, and adaptive cybercrime decision optimisation into a unified end-to-end architecture. Dynamic graph building, graph attention learning, hybrid anomaly scoring, node and edge attribution, analyst trust estimate and adaptive graph risk propagation all contribute to considerably enhancing the detection accuracy and interpretability.

The suggested framework was assessed using the CIIoT2023 and NF-ToN-IoT benchmark datasets. Experimental results showed better performance than the previous techniques with 99.61% accuracy, 99.54% precision, 99.47% recall, 99.50% F1-score, and 0.999 AUC on the CIIoT2023 dataset. Similarly, the framework obtained 99.42% accuracy, 99.31% precision, 99.24% recall, 99.27% F1-score, and 0.999 AUC on the NF-ToN-IoT dataset. The ablation study also confirmed that each proposed component progressively contributed to the overall performance, and robustness analysis showed consistently high detection capability across multiple IoT attack categories, including DDoS, DoS, Mirai, reconnaissance, spoofing, brute-force, injection and botnet attacks.

Further, EDGNN-Forensics is extended to large-scale distributed edge and cloud systems using graph federated learning and continual learning techniques. Other possible enhancements may be self-supervised graph representation learning, multimodal cyber threat intelligence integration, streaming graph analytics, and lightweight graph neural architectures for implementation on resource-constrained IoT devices. These improvements will lead to enhanced scalability, adaptability, explainability for next-generation intelligent cybercrime detection systems.

REFERENCES

- [1] A. Khan, "Ai-driven cybercrime forensics for predictive threat detection and investigative intelligence," *International Journal of Scientific Interdisciplinary Research*, vol. 7, no. 1, pp. 280–324, 2026.
- [2] A. S. Ahanger, S. M. Khan, F. Masoodi, and A. O. Salau, "Advanced intrusion detection in internet of things using graph attention networks," *Scientific Reports*, vol. 15, no. 1, p. 9831, 2025.
- [3] Y. Zhang, C. Jue, W. Liu, and Y. Ma, "Gran: a sdn intrusion detection model based on graph attention network and residual learning," *The Computer Journal*, vol. 68, no. 3, pp. 241–260, 2025.
- [4] W. Villegas-Ch, J. Govea, A. M. Navarro, and P. P. Játiva, "Intrusion detection in iot networks using dynamic graph modeling and graph-based neural networks," *IEEE Access*, 2025.
- [5] K. Bouzaachane, A. M. Alnajim, S. Khan et al., "Addressing modern cybersecurity challenges: A hybrid machine learning and deep learning approach for network intrusion detection," *Computers, Materials and Continua*, vol. 84, no. 2, pp. 2391–2410, 2025.
- [6] S. Duraibi and A. M. Alashjaee, "Enhancing cyberattack detection using dimensionality reduction with hybrid deep learning on internet of things environment," *IEEE Access*, vol. 12, pp. 84 752–84 762, 2024.
- [7] T. Vaiyapuri, K. Shankar, S. Rajendran, S. Kumar, V. Gaur, D. Gupta, and M. Alharbi, "Automated cyberattack detection using optimal ensemble deep learning model," *Transactions on Emerging Telecommunications Technologies*, vol. 35, no. 4, p. e4899, 2024.
- [8] S. Venkatasubramanian, A. Thirumalraj, and S. Hariprasath, "A multi-layer analytical framework for secure and energy-aware iot: Integrating resource allocation with ksgo-based optimization," in *2026 Third International Conference on Networking and Communications (ICNWC)*. IEEE, 2026, pp. 1–10, DOI not yet indexed/available at time of writing.
- [9] A. Al Mazroa, F. R. Albogamy, M. K. Ishak, and S. M. Mostafa, "Boosting cyberattack detection using binary metaheuristics with deep learning on cyber-physical system environment," *IEEE Access*, vol. 13, pp. 11 280–11 294, 2025.
- [10] B. B. Gupta, A. Gaurav, R. W. Attar, V. Arya, S. Bansal, A. Alhomoud, and K. T. Chui, "A hybrid ant lion optimization algorithm based lightweight deep learning framework for cyber-attack detection in iot environment," *Computers and Electrical Engineering*, vol. 122, p. 110008, 2025.
- [11] R. Aboalela, K. H. Allehaibi, N. Alsaadi, A. S. Alorfi, L. A. Maghrabi, E. B. Ashary, W. Alghamdi, and M. Ragab, "Harnessing feature pruning with optimal deep learning-based distributed denial of service cyberattack detection on iot environment," *Alexandria Engineering Journal*, vol. 120, pp. 584–597, 2025.
- [12] W. Wu, H. Fouzi, B. Benamar, S. Sidi-Mohammed, and S. Ying, "Deep learning-based stacked models for cyber-attack detection in industrial internet of things," *Neural Computing and Applications*, vol. 37, no. 24, pp. 19 617–19 651, 2025.
- [13] M. Barbhaya, P. R. Dasari, S. K. Damarla, R. Srinivasan, and B. Huang, "A deep learning framework for cyberattack detection and classification in industrial control systems," *Computers & Chemical Engineering*, p. 109278, 2025.
- [14] A. A. Salih and M. B. Abdulrazaq, "Cybernet model: A new deep learning model for cyber ddos attacks detection and recognition," *Computers, Materials and Continua*, vol. 78, no. 1, pp. 1275–1295, 2024.
- [15] A. Abughali, M. Alansari, and A. S. Al-Sumaiti, "Deep learning strategies for detecting and mitigating cyber-attacks targeting water-energy nexus," *IEEE Access*, vol. 12, pp. 129 690–129 704, 2024.
- [16] Kaggle, "Ciciot2023 dataset," <https://www.kaggle.com/datasets/himadri07/ciciot2023>, 2023.
- [17] Kaggle1, "Nf-ton-iot dataset," <https://www.kaggle.com/datasets/dhoogla/nftoniot>, 2023.
- [18] Z. Ali, T. Hussain, C.-L. Su, I. Khan, A. D. Jurcut, S.-H. Tsao, C.-H. Hu, and M. Elsis, "Deep learning-driven cyber attack detection framework in dc shipboard microgrids system for enhancing maritime transportation security," *IEEE Transactions on Intelligent Transportation Systems*, vol. 26, no. 11, pp. 20 122–20 142, 2025.
- [19] M. Elgamal, A. A. Eladl, B. E. Sedhom, A. N. Sheta, A. Refaat, and A. A. Menaem, "A deep learning-based cyberattack detection method for transmission line differential relays," *Internet of Things*, vol. 31, p. 101574, 2025.
- [20] A. Sangeetha, J. J. Alaguraja, and R. Latip, "E-hunf: Explainable hybrid unsupervised network forensics for robust cybercrime anomaly detection," *International Journal of Electronics and Telecommunications*, vol. 72, no. 2, pp. 1–8, 2026.
- [21] V. S. Anusuya, S. Baswaraju, A. Thirumalraj, and A. Nedumaran, "Securing the manet by detecting the intrusions using cso and xgboost model," in *Intelligent Systems and Industrial Internet of Things for Sustainable Development*. Chapman and Hall/CRC, 2024, pp. 219–234.