THE SCIENTIFIC TEMPER (2026) Vol. 17 (4): 6103-6116

E-ISSN: 2231-6396, ISSN: 0976-8653

Doi: 10.58414/SCIENTIFICTEMPER.2026.17.4.21

https://scientifictemper.com/

RESEARCH ARTICLE

Securing Smart IoT Networks from Cyber Threats Using Explainable Zero Channel Attention-aided Ghost Convolution Neural Network Framework

S. Razool Begum[1], Dr. S Malathi[2*]

Abstract

The rise of advancements driven by the Internet of Things (IoT), presents numerous benefits, such as improved efficiency and enhanced quality of life. However, this interconnectedness also exposes critical vulnerabilities, making robust cyber-attack detection essential. Hence, this manuscript emphasizes the innovative explainable deep learning (XAI-DL) model for detecting and classifying multiclass cyber threat attacks in Internet of Things (IoT) platforms. Initially, the raw data samples collected from the BCCC-CIC-IDS dataset are preprocessed by performing a Pareto Scaling Normalization (PSN) and one-hot encoding processto improve the data quality.After preprocessing, the Zero Channel Attention-aided Ghost Convolution Neural Network (ZCAtt-GCNN) is proposed to detect and classify the various cyber threat attacks like Denial of Service (DoS) Attacks, Distributed Denial of Service (DDoS) Attacks, Web Attacks, file transfer protocol (FTP) Attacks, and Botnet Attacks. Furthermore, three XAI models are investigated for enhanced visualizations over the cyberattack detection: Shapley additive explanations (SHAP), Partial Dependence Plot-Individual Conditional Expectation (PDP-ICE), and Permutation Feature Importance (PFI). The proposed method is simulated via the Python platform and various performance measures like G-mean, Accuracy, Matthews Correlation Coefficient (MCC), Negative Predictive Value (NPV), computation time (CT), and false positive rate (FPR) are scrutinized, and associated with other techniques. The overall accuracy of 99.48%, G-mean of 99.18%, and FPR of 0.322 are obtained by the suggested framework for identifying various IoT Cyberattacks.

Keywords: Internet of Things (IoT), Cyberattack Detection, Multiclass Classification, Data Normalization, Network Security, Deep learning, Explainable Artificial Intelligence (XAI).

Introduction

In the modern digitally connected world, cyberattacks have become a common occurrence. This refers to unauthorized penetration into the digital assets of an organization's network (Shtayat 2023; Aslam, 2022). While firewalls protect the networks, their ability to monitor network packets to detect and prevent many malicious activities is limited. Network Anomaly Detection Systems (ADS) supplement network security through the ongoing monitoring and analysis of traffic data. Based on the detection method, network intrusion detection systems (NIDS) can be broadly categorized into two types: signature-based systems, and anomaly detection systems, or ADS (Faheem, 2022; Yamarthy, 2024).

The misuse detection method depends on a catalog of acknowledged attack signatures and compares incoming network traffic against these stored patterns. When there is a match, an alert is triggered. However, ADS detects deviations from normal traffic patterns to detect potential attacks (Dasari, 2024; Wanda, 2024). The signature-based approach is often successful for detecting known attacks, with a very small FAR but requires frequent up-to-date updates in an almost manual fashion to the signature database. The anomaly-based technique can successfully recognize new attacks presented over time but produce a high false positive rate (FPR) (Rajak , 2023; Mohsenabad, 2024).

[1]Research Scholar, Department of Computer Science, A. Veeriya Vandayar Memorial Sri Pushpam College (Autonomous), Poondi, Thanjavur(dt), 613503, India. (Affiliated to Bharathidasan University, Tiruchirappalli)

[2]Research Supervisor & Assistant Professor of Computer Science, Swami Dayananda College of Arts & Science, Manjakkudi, Tiruvarur(dt), Tamil Nadu, India. (Affiliated to Bharathidasan University, Tiruchirappalli)

***Corresponding Author:** Dr S Malathi, Research Supervisor & Assistant Professor of Computer Science, Swami Dayananda College of Arts & Science, Manjakkudi, Tiruvarur(dt), Tamil Nadu, India, E-Mail: visitmalathi@gmail.com

How to cite this article: Begum, S.R., Malathi, S. (2026). Securing Smart IoT Networks from Cyber Threats Using Explainable Zero Channel Attention-aided Ghost Convolution Neural Network Framework. The Scientific Temper, **17**(4):6103-6116.

Doi: 10.58414/SCIENTIFICTEMPER.2026.17.4.21

Source of support: Nil

Conflict of interest: None.

Received: 27/03/2026 **Accepted:** 15/04/2026 **Published:** 25/04/2026

Anomaly-based methods often employ ML and DL frameworks to determine traffic as normal or abnormal. Various ML schemes, including SVM, DT, and RF, are widely used in this domain (Vibhute, 2024). Many studies combine multiple classifiers, as ensemble learners are generally considered superior to individual classifiers due to computational, statistical, and representational advantages (Rizvi, 2024 ; Muthukumar, 2024). Moreover, DL frameworks like convolution neural networks (CNN), recurrent NN (RNN), and autoencoders (AE) are quite effective in analyzing time-varying attack patterns with minimal computations. Different research frameworks have investigated different methods of reducing DDoS attacks. The authors (Kuppusamy, 2011) suggested a good prevention technique with the use of the time-frequency algorithm for enhancing attack detection within network security. Subsequently, they developed further research by suggesting a target customer behavior-based approach (Kuppusamy, 2012) to facilitate the prevention of DDoS attacks by observing user behavior. In (Akilandeswari, 2022) proposed a blockchain approach using Bitcoin and Ethereum technology for DDoS attack control and prevention, providing a decentralized security solution. The authors suggested an ensemble DL-based cyberattack detection mechanism to improve the trustworthiness of IoT network nodes (Malathi, 2024). Though these methods have helped to make DDoS prevention and cybersecurity more effective, they remain problematic due to high computational complexity, latency, and responsiveness to changing patterns of attack, making real-time application a continued issue.

However, artificial intelligence (AI) algorithms often operate as black-box models, making their resolution progressions difficult for humans to interpret, comprehend, and sometimes even trust (Kumari, 2024; Yaras, 2024). This lack of transparency makes black-box AI unsuitable for applications where interpretability and explainability are critical, such as in cybersecurity. To address this challenge, researchers have developed PoAh enabled federated learning architecture for DDoS attack detection in IoT networks (Park, 2024).

Motivation: The rapid growth of IoT networks has resulted in significant security and reliability challenges concerning massive heterogeneous data generated by the numerous devices connecting to a network. In most cases, anomaly detection systems lack interpretability and can fail to explain why an instance is detected as anomalous. Many AI models often operate in a black-box fashion, impairs the trust and uptake of critical applications in IoT settings. Traditional approaches are based mostly on a single model and pre-defined rules, failing to fit the dynamic scenario of an IoT environment in most scenarios. Most of the models fail to capture the complex nature of different types of attacks: botnets and sensor faults leading to diluted detection accuracy. Furthermore, most recent studies fail to consider the importance of identifying critical features that support anomaly detection, thus limiting the potential to improve performance and focus on critical threat indicators. Additionally, the inability to explain and generalize in currently dominating systems greatly limits their usage in real-world applications, especially where compliance with regulations and users' trust is of crucial importance. These constraints necessitate a DL framework that is well-informed by XAI, to not only enhance the efficacy of anomaly detection but also provide clarity on the decision-making processes, thereby helping actionable intelligence and boosting stakeholders' confidence.

The foremost contributions of the proposed framework are encompassed as follows:

- To develop a preprocessing pipeline using one-hot encoding and Pareto Scaling Normalization (PSN) to standardize raw data samples from the BCCC-CIC-IDS dataset, ensuring improved feature distribution for model training.
- To propose a Zero Channel Attention-aided Ghost Convolution Neural Network (ZCAtt-GCNN) for the detection and classification of multiple cyber threats, including DoS, DDoS, Web Attacks, FTP Attacks, and Botnet Attacks.
- To incorporate explainable AI (XAI) models such as Shapley Additive Explanations (SHAP), Partial Dependence Plot-Individual Conditional Expectation (PDP-ICE), and Permutation Feature Importance (PFI) for enhanced interpretability and visualization of cyberattack detection results.
- To conduct extensive simulations on the Python platform, evaluating the proposed method using key performance metrics such as G-mean, Accuracy, MCC, NPV, Computation Time (CT), and FPR.
- To compare the performance of the ZCAtt-GCNN model against existing cyberattack detection techniques, demonstrating its efficiency in accurate classification and reduced false positive rates.

The upcoming sections are prearranged as follows: Section 2 outlays the related work, Section 3 deliberates over the suggested approaches, Section 4 presents the results and discussion, and Section 5 represents the conclusion of the proposed framework.

Related Works: A Brief Review

Among the several studies on cyberattack detection in IoT technology, several current works are deliberated in this section

The authors have introduced the XAI-based IDS framework for IoT networks. An IDS was designed using an LSTM model for detecting online attacks and also explaining the model's decisions. The model was trained and tested using an exclusive set of input attributes extracted using a novel SPIP framework. The method was tested on the

NSL-KDD, UNSW-NB15, and TON_IoT databases. While the above approach displayed excellent performance, its computational complexity was a major drawback because large-scale IoT environments would hamper its deployment in real-time scenarios (Keshk,2023). A Comparative and Hybrid Machine Learning Framework for IoT-Based Predictive Maintenance of Rotating Machinery have framed with XAI (Surendra, 2026).

(Mahbooba, 2021) have formulated the XAI to enhance trust management by examining the decision tree (DT) model in the context of IDS. Simple DT algorithms were employed, offering ease of interpretability and closely mimicking human decision-making processes by breaking down decisions into smaller, manageable sub-choices for IDS. This approach was tested using rules extracted from the widely recognized KDD benchmark dataset. Additionally, the performance of the decision tree method was compared to other advanced algorithms in terms of precision. However, limitations were encountered, including the decision tree model's sensitivity to noisy data and its tendency to overfit when handling complex datasets.

(Zebin*et, 2022)* have defined the XAI using a novel ML framework to address the detection and classification of DNS over HTTPS (DoH) attacks. The CIRA-CICDoHBrw-2020 dataset, publicly available, was utilized to build an accurate model for this purpose. XAI techniques were employed to reveal the contributions of various features, offering transparent and interpretable insights into the model's decision-making process. The interpretability of complex models may still present challenges, as feature importance alone does not always provide a complete understanding of the decision-making process.

The authors have encompassed the IDS that incorporated preprocessing methods and a DL approach for identifying DDoS attacks. To achieve this, several models utilizing DNN, CNN, and LSTM were assessed based on their detection accuracy and performance in real-time scenarios. The CIC-DDoS2019 dataset, a widely referenced resource in the field, was employed for testing the framework. Preprocessing steps, including feature elimination, random subset selection, feature extraction, duplication removal, and data normalization, were applied to enhance the dataset's quality and optimize the detection process. However, the preprocessing phase, while crucial for improving model accuracy, introduced additional overhead, which could affect deployment efficiency in large-scale systems (Akgun, 2022).

(Mughaid*et* , 2022) have outlined the ML techniques by dividing the dataset into training and testing subsets. The training data was utilized to build the detection model, while the test data was employed to validate the results. The model aimed to capture the inherent characteristics of email text and additional features to classify emails as phishing or non-phishing. Three different datasets were used for this purpose, and a comparison was conducted among them. It was observed that utilizing a higher number of features led to more accurate and efficient classification results. The model's performance heavily relied on the quality and diversity of the datasets, which might limit its applicability to unseen or evolving phishing techniques.

The authors have established various DL models like RNN, CNN, and DNN to identify cyberattacks across different network traffic streams. The Canadian Institute for Cybersecurity's CICDIoT2023 dataset was used to evaluate the effectiveness of the approach. The methodology incorporated data preprocessing techniques, robust scaling, label encoding for categorical variables, and prediction using DL models. The effectiveness of the approach was influenced by the dataset's representativeness, which might affect its applicability to other datasets or real-world traffic with different attack patterns (Abbas, 2024).

Proposed Methodology

This manuscript emphasizes the innovative explainable deep learning (XAI-DL) model for detecting and classifying multiclass cyber threat attacks in Internet of Things (IoT) platforms. Figure 1 indicates the workflow of the suggested framework.

Initially, the raw data samples collected from the BCCC-CIC-IDS dataset are preprocessed by performing a normalization process. The PSN technique is introduced to improve the data quality. After preprocessing, theZCAtt-GCNNis proposed to detect and classify the various cyber threat attacks like Denial of Service (DoS) Attacks, Distributed Denial of Service (DDoS) Attacks, Web Attacks, FTP Attacks, and Botnet Attacks. Furthermore, three XAI models are investigated for enhanced visualizations over the cyberattack detection: Shapley additive explanations (SHAP), Partial Dependence Plot-Individual Conditional Expectation (PDP-ICE), and Permutation Feature Importance (PFI).

Data Acquisition

The Network Intrusion Detection dataset available on Kaggle is designed to support the development and evaluation of machine learning models for identifying unauthorized network access. The dataset contains 5,000 records, each representing a network connection with various extracted features. These features include attributes such as the duration of the connection, protocol type (e.g., TCP, UDP, ICMP), network service (e.g., HTTP, FTP, SMTP), connection status flags, and the number of bytes exchanged between source and destination. Other features include binary indicators for specific behaviors, such as whether the connection involves the same host and port, the number of failed login attempts, and whether root shell access was obtained. Additionally, statistical features, such as the percentage of connections with errors (e.g., SYN or

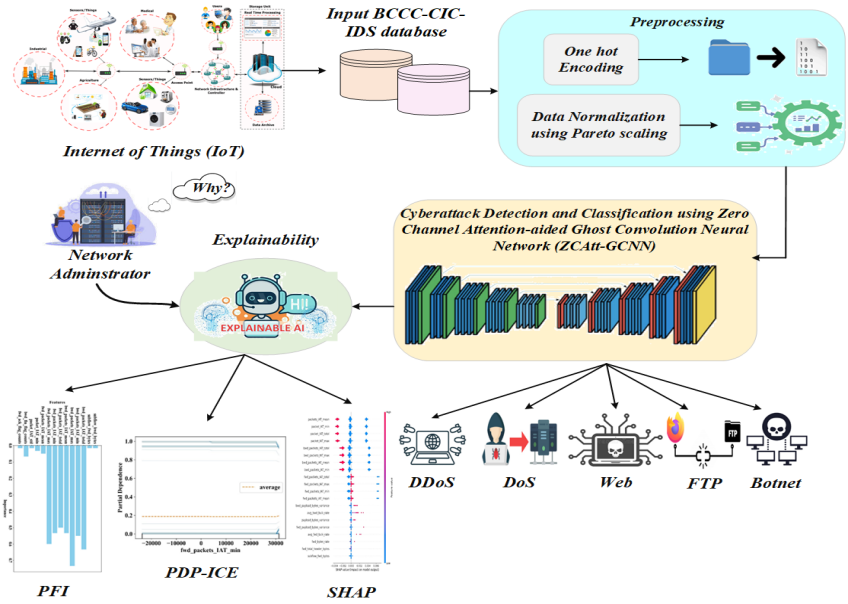


Figure 1: Workflow of the Suggested Framework

REJ errors), the count of connections to the same host or service in a given time frame, and rates of different types of services or hosts accessed, are also included. The dataset also provides categorical features, such as protocol type and network service, which are encoded for model compatibility.

Preprocessing Stage

The raw database collected from a freely accessible database contains the categorical features of each IoT sensor having insignificant values and are modified into numerical fields. To indicate the categorical parameters in a way applicable to learning models, a one-hot encoding is performed (Tokareva , 2021). This process converts the actual values into the binary series representation, where each exclusive value is indicated as a distinct column or attribute with a value of either 1 or 0. After encoding, data normalization is performed using the Pareto Scaling (PS) technique operates similarly to the z-score method but with a key difference: the scaling factor is the square root of the standard deviation (SD). As a result, the newly scaled features will exhibit a variance that matches the standard deviation of the original, unscaled features. Every sample $y'_{u,m}$ of the data is modified into $y_{u,m}$ and it can be formulated as,

$$y'_{u,m} = \frac{y_{u,m} - \lambda_u}{\sqrt{\sigma_u}} \quad (1)$$

Here, λ_u and σ_u represents the mean and SD of u^{th} feature respectively.

Cyberattack Detection using ZCAtt-GCNN Technique

The preprocessed data is then fed into the innovative Zero Channel Attention-aided Ghost Convolution Neural Network (ZCAtt-GCNN) to classify cyber attacks in the IoT framework. Traditional Attention-CNN (Att-CNN) assumes that data follows a specific distribution (usually Gaussian) in the latent space (Luan, 2024). In IoT environments, the data distribution can be highly non-Gaussian and complex, making it difficult for Att-CNNs to model such distributions accurately. The Proposed model addresses the limitation of traditional Att-CNNs assuming Gaussian distribution in the latent space, especially in IoT environments where the data distribution is often non-Gaussian and highly complex.

The proposed GCNN network requires minimal convolution (Conv) filters and tiny conv kernel sizes. It is adopted as the backbone to encode feature representation which removes similar feature maps. Several Ghost bottleneck blocks are connected in stacked format and it is commonly termed as Ghost module. Assume the input $Z \in \mathbb{R}^{c \times h \times w}$ with a set of 1×1 conv filters which are used to minimize half of the input channels and generate squeezed features $Y \in \mathbb{R}^{c \times h \times w}$. In the next phase, the cheap linear process is introduced for every channel P to determine the ghost features(GF) $U \in \mathbb{R}^{c/2 \times h \times w}$. Particularly, the CL process is represented as the group conv which is equal to the input channels. The CL parameter operation is lesser than the conv operation, and the GF obtained by the linear operations can modify the redundant features obtained by conv filters. At the last, the outcome $O \in \mathbb{R}^{c \times h \times w}$ can be obtained using concatenating with Y and U. The outcome from the ghost module (GM) can be mathematically formulated as,

$$O = Concat\left(Conv_1(Y), CL\left(Conv_1(Y)\right)\right) \quad (2)$$

Here, $Concat$ and CL represents the concatenation and CL processes respectively. In the next phase, dual GMsare altered to produce the ghost bottleneck block (BB). It is separated into dual types based on various strides. The BB with stride 2 minimizes the spatial feature dimension by half to encapsulate the semantic details of particular data. Ultimately, the GhostNet is created by stacking ghost BB based on MobileNet architecture.Figure 2 indicates the architecture of ZCAtt-GCNN model.

Zero Channel Attentive (ZCAtt) Module

The aim of the ZCAtt module in cyberattack detection is to determine the region where the abnormal data is from the feature map. Linear transform increases the necessity of final channels. Particularly, it calculates the linear separability between the final and other channels according to the energy function as,

$$E = \frac{4\left(\sigma^2 + \beta\right)}{\left(g_c - \rho\right)^2 + 2\sigma^2 + 2\beta} \quad (3)$$

Here, ρ and σ^2 indicates the mean and variance of the feature g_c, β manipulates the constant to eliminate σ to move into the value 0 and it is set to the value 0.0001. The above equation indicates that the energy function is defined using the linear transformation of mean and variance for spatial features of every channel. Minimal value of E, the channel obtained will be more discriminated from the nearby channel hence, the maximum weight is to be determined. The reciprocal of equation (3) deliberates the necessity of the dataset.

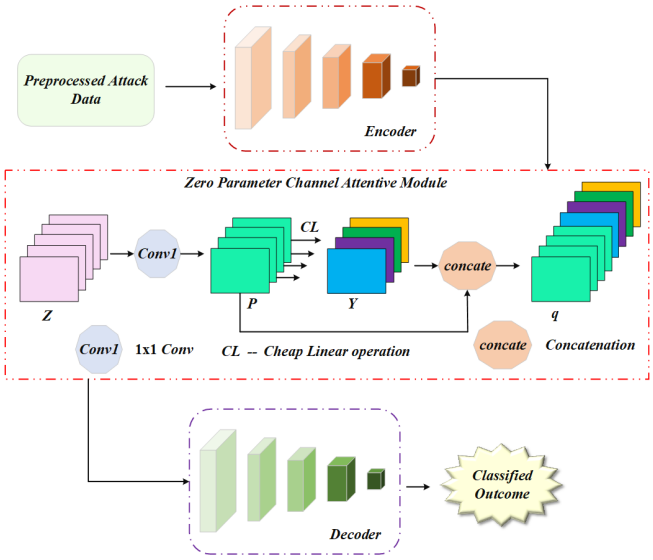


Figure 2: Architecture of ZCAtt-GCNN Model

The ZCAtt module is introduced to enhance the detection performance without increasing the parameters. Assume an input P, the global average pooling process is initially trained to minimize the size of P to 1×1 which generates the set of global attention map N. Using the energy function, the updated channel attention maps M are successively obtained by utilizing the sequential element-wise operation on N. It can be formulated as,

$$N = \sum_{a=1}^{A} \frac{\left(M_a - \rho\right)^2 + 2\left(\sigma^2 + \beta\right)}{4\left(\sigma^2 + \beta\right)} \quad (4)$$

Here, M_a indicates the a^{th} CAtt map. In the next phase, the attention map M is remodified using the sigmoid function. In the final stage, P is multiplied by M to generate the outcome q. The function of ZCAtt module can be expressed as,

$$q = P \times \left\langle sig\left(g_e\left(GAP(P)\right)\right)\right\rangle \quad (5)$$

Here, GAP and g_e indicates the global average pooling and linear transformation process respectively. The ZCAtt module has no training parameters and the enhanced channel feature maps can be generated using mathematical operations.

Moreover, different XAI techniques like Shapley additive explanations (SHAP), Partial Dependence Plot-Individual Conditional Expectation (PDP-ICE), and Permutation Feature Importance (PFI) are implemented for visualizing the distinct predictions made by the developed model (Sharma, 2021). These models are considered the global XAI schemes that assist in finding a customized set of particular input features that effectively explain the various functionalities of the developed model. A brief analysis of various XAI models is deliberated below:

SHAP-XAI Analysis

SHAP gives an interpretable representation of the individual pixels or features that contribute to the classification of cyberattack abnormalities. Calculating Shapley values brings to the surface the contribution of each feature toward the given prediction, where red represents those features that positively impact the prediction for a particular class. Blue colorrepresents those that reduce the probability of the predicted class. The Shapley values are obtained after understanding the joint feature interactions of attack abnormalities. These can be mathematically represented as in equation (6),

$$\phi_x = \sum_{S \subseteq M \setminus \{x\}} \frac{|S|!\left(N - |S| - 1\right)!}{N!}\left(g_u\left(S \cup \{x\}\right) - g_u(S)\right) \quad (6)$$

Here, $g_u\left(S \cup \{x\}\right)$ characterizes the accustomed forecast when the feature x is encompassed in the subset S, $g_u(S)$

symbolizes the forecast lacking a feature x, M indicates the total set of features, S denotes the subset of attributes without x, N indicates the total number of features, $\frac{|S|!(N-|S|-1)!}{N!}$ acting as the weighting factor for each subset. The SHAP framework transforms actual feature traits u_x into contributions to the classification using a binary indicator y'_x that signifies the existence or non-existence of the feature. The prediction for the replacement model $f(y')$ is computed using equation (7),

$$f(y') = \phi_0 + \sum_{x=1}^{N} \phi_x y'_x \tag{7}$$

Where, ϕ_0 represents the baseline or bias of the model, and ϕ_x indicates the influence of every feature x to the consequence. This decomposition ensures that the prediction is the sum of bias and individual feature contributions, showing in a clear and explainable manner how the input features influence the classification decision. This approach is exactly what's required in validating the classification of anomalies, showing the positive and negative impacts of specific features in the attack data.

PFI-XAI Analysis

The PFI is one of the model-independent schemes that computes the alteration in prediction error of the proposed model when permuting input feature values. This procedure deliberates the feature connection between all features and the target outcome. Hence, the model's error increases when the model contains the necessary features. Here, three major phases are involved by considering the trained model G, matrix features Z and the desired outcome q and the error estimation $\delta(q,G)$. The PFI value for the j^{th} feature can be depicted as,

$$s_{\geq} = \frac{e^{(permutation)}}{e^{(actual)}} \tag{8}$$

Finally, the attributes are determined in downward order using the obtained value of $S_{\geq}$.

PDP Explanation

The PDP is another widely utilized visualization technique that helps to provide the relationship between the outcome of the black box model and the feature values by plotting visualizations. The PDP is the global scheme that stores all the information and defines the relationship between prediction and a particular feature. It can be mathematically formulated as,

$$PD(z_s) = E(z_c)\left[G(z_s, z_c)\right] = \int f(z_s, z_c)\, dP(z_c) \tag{9}$$

Here, G indicates the proposed model, $z_s \in s$ deliberates the plotted input features, $z_c \in s$ signifies the input features f, and $dP(z_c)$ signifies the marginal distribution of z_c.

ICE Analysis

As opposed to PDP, which estimates how a model behaves but puts this estimation in the light of individual observations rather than averages across the whole data, the ICE method gives one distinct line for every observation reflecting in what way a prediction depends on a feature, using variations of the feature: Essentially, although PDP is an average of all these individual lines, ICE shines lighter on the interaction that may be unique and exist between a feature and the prediction for each of the observations. PDP is only good when interactions between features are relatively weak because it suppresses the heterogeneous patterns caused by the interactions among features. This is the case with ICE because it can expose these heterogeneous patterns. For a set of observations $\left\{\left(x_s^{(i)}, x_c^{(i)}\right)\right\}_{i=1}^{k}$ ICE plots $g_s^{(i)}$ against $x_s^{(i)}$ for each of the k observations, showing each observation's response to the feature. Averaging all ICE curves then gives the PDP.

Results and Discussion

The proposed scheme is analyzed and processed via the Python simulation platform. For the experimentation process, hyperparameters like learning rate, min-batch size, dropout, and several epochs are considered. The proposed method is processed under Intel(R) Core (TM) i5-4300M CPU with 4GB installed RAM using a 64-bit operating system. For the training process, 80% of training data, 10% of testing data, and 10% of validation data are considered which are in the ratio 8:1:1. Table 1 indicates the Hyperparameters of the Developed Method.

Assessment Measures

Performance indicators accuracy, False positive rate (FPR), negative predictive value (NPV), G-mean, and Matthew's correlation coefficient (MCC) are computed to better understand the proposed approach.

Accuracy

It determines the model's overall accuracy, accounting for both TP and TN. It is calculated using equation (10),

$$Accuracy = \frac{T_n + T_p}{T_p + F_n + F_p + T_n} \tag{10}$$

FPR Analysis

FPR is the proportion of actual negatives that were incorrectly classified as positives. It is also called the Type I

Table 1: Hyperparameters of the Developed Method

Parameters used	*Values*
Learning rate	0.001
Optimizer	Adam Optimizer
Batch size	4
Dropout	0.5
Total number of epochs	100

error rate. It is calculated using equation (11),

$$FPR = \frac{F_p}{F_p + T_n} \tag{11}$$

NPV Analysis

NPV is the proportion of predicted negative instances that are negative. It tells you how reliable a negative result is. It is calculated using equation (12),

$$NPV = \frac{T_n}{T_n + F_n} \tag{12}$$

G-mean Analysis

G-Mean is a measure that balances the classification performance of both the positive and negative classes. It is often used when the classes are imbalanced. It is calculated using equation (13),

$$G-mean = \sqrt{\left(\frac{T_p}{T_p + F_n}\right)\left(\frac{T_n}{T_n + F_p}\right)} \tag{13}$$

MCC Analysis

A metric used to assess the validity of multi-classifications, especially when there is an imbalance between the classes, and it is assessed using equation (14),

$$MCC = \left(\frac{T_p \times T_n - F_p \times F_n}{\sqrt{(T_p + F_p)(T_p + F_n)(T_n + F_p)(T_n + F_n)}}\right) \tag{14}$$

Here, T_n, T_p, F_n, F_p indicates the true negative (TN), true positive (TP), false negative (FN), and false positive (FP) respectively.

Confusion Matrix Analysis

In this section, the Confusion matrix (CM) is analyzed for the developed framework under normal and attacked classes. The confusion matrix is one of the most essential and effective analyses that assist in a better understanding of the proposed approach over misclassified outcomes.

The CM in Figure 3 represents the classification performance for five attack categories: DoS, DDoS, Web, FTP, and Botnet. The correctly classified instances for each category are 589 for DoS, 601 for DDoS, 601 for Web, 564 for FTP, and 554 for Botnet. Misclassification is observed across different categories, with DoS being misclassified as DDoS (2 instances), Web (2 instances), FTP (4 instances), and Botnet (2 instances). Similarly, DDoS has 5 instances misclassified as DoS, 1 as Web, 1 as FTP, and 4 as Botnet. Web attack misclassification includes 1 instance each as DoS, DDoS, and FTP, along with 4 instances as Botnet. FTP misclassification includes 1 instance as DoS, 5 as DDoS, 1 as Web, and 6 as Botnet, while a Botnet attack has 1 instance misclassified as DoS, 1 as DDoS, 6 as Web, and 1 as FTP.

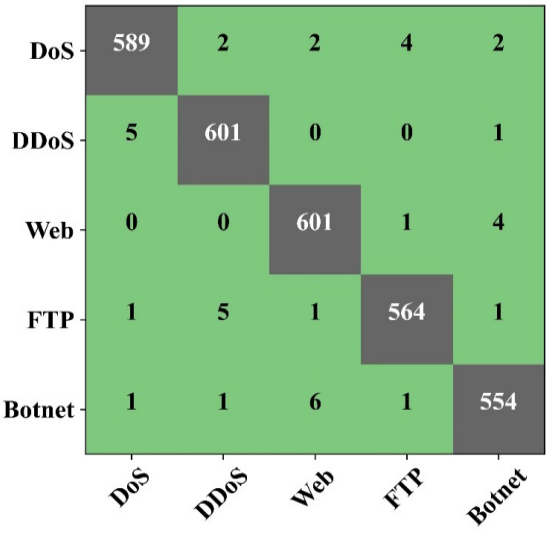


Figure 3: CM Analysis of the Suggested Framework

Simulation Analysis of Proposed Scheme over Traditional Methods

In this section, the effectiveness achieved by the introduced method over the existing schemes is deliberated via graphical illustration. Several existing methods like CNN, LSTM-AE, GCNN, VAE-BiGRU, and Att-CNN are compared with the proposedXAI-ZCAtt-GCNN technique framework. The comprehensive analysis of the attained efficacy is depicted below.

Figures 4(a) and 4(b) indicate the training and validation analysis for accuracy and loss respectively. The accuracy and loss curves illustrate the model's training performance over 100 epochs. The accuracy curve in Figure 4(a) shows that the training accuracy starts at approximately 0.82 and rapidly increases within the first epochs, stabilizing around 0.98 to 0.99 after approximately 20 epochs. The validation accuracy follows a similar trend but exhibits slight fluctuations, indicating good generalization with minor variance. The loss curve in Figure 4(b), on the other hand, demonstrates a steady decrease in both training and validation loss.

Initially, the loss is above 0.175 but drops sharply within the first 10 epochs and stabilizes near 0.025 after approximately 40 epochs. While the validation loss follows a similar trajectory, minor spikes suggest occasional variations in generalization. Overall, the minimal gap between training and validation curves indicates that the model does not suffer from severe overfitting. The validation accuracy and loss fluctuations are within an acceptable range, reinforcing the model's stability. The final accuracy of nearly 99% and the low loss value signify that the model performs effectively on the given dataset.

Figure 5 indicates the accuracy analysis for varying existing schemes. The graphical representation shows that

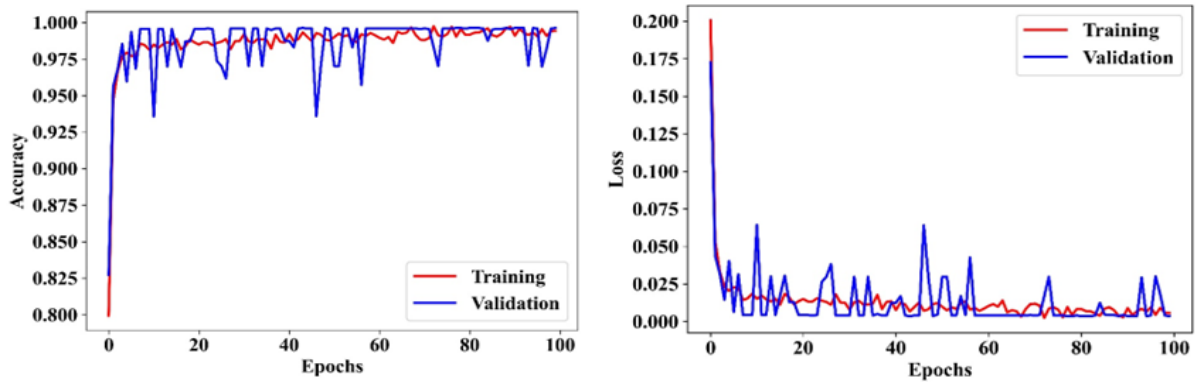


Figure 4: Training and Validation Analysis, (a) Accuracy and (b) Loss

the proposed XAI-ZCAtt-GCNNtechnique obtained better detection performance when associated with conventional schemes. For the DoS attack, the conventional CNN, GCNN, Att-CNN, LSTM-AE, VAE-BiGRU, and suggested XAI-ZCAtt-GCNNtechniques obtained an accuracy of 97.48%, 97.82%, 98.16%, 98.67%, 99.08%, and 99.42% respectively. For the DDoS attack, the conventional CNN, GCNN, Att-CNN, LSTM-AE, VAE-BiGRU, and suggested XAI-ZCAtt-GCNNtechniques obtained an accuracy of 97.76%, 98.06%, 98.43%, 98.81%, 99.25%, and 99.52% respectively. For the Web attack, the conventional CNN, GCNN, Att-CNN, LSTM-AE, VAE-BiGRU, and suggested XAI-ZCAtt-GCNNtechniques obtained an accuracy of 97.76%, 98.2%, 98.47%, 99.08%, 99.25%, and 99.52% respectively. For the FTP attack, the conventional CNN, GCNN, Att-CNN, LSTM-AE, VAE-BiGRU, and suggested XAI-ZCAtt-GCNNtechniques obtained an accuracy of 97.82%, 98.13%, 98.47%, 98.94%, 99.25%, and 99.52% respectively. For the botnet attack, the conventional CNN, GCNN, Att-CNN, LSTM-AE, VAE-BiGRU, and suggested XAI-ZCAtt-GCNNtechniques obtained an accuracy of 97.82%, 98.13%, 98.37%, 98.77%, 99.08%, and 99.42% respectively.

Figure 6 indicates the FPR analysis for varying existing schemes. The graphical representation shows that the proposed XAI-ZCAtt-GCNNtechnique obtained better detection performance when associated with conventional schemes. For the DoS attack, the conventional CNN, GCNN, Att-CNN, LSTM-AE, VAE-BiGRU, and suggested XAI-ZCAtt-GCNNtechniques obtained an FPR of 1.44, 1.23, 1.022, 0.68, 0.46, and 0.29 respectively. For the DDoS attack, the conventional CNN, GCNN, Att-CNN, LSTM-AE, VAE-BiGRU, and suggested XAI-ZCAtt-GCNNtechniques obtained an FPR of 1.36, 1.19, 1.02, 0.76, 0.51, and 0.34 respectively. For the Web attack, the conventional CNN, GCNN, Att-CNN, LSTM-AE, VAE-BiGRU, and suggested XAI-ZCAtt-GCNNtechniques obtained an FPR of 1.58, 1.28, 1.15, 0.76, 0.55, and 0.38 respectively. For the FTP attack, the conventional CNN, GCNN, Att-CNN, LSTM-AE, VAE-BiGRU, and suggested XAI-ZCAtt-GCNNtechniques obtained an FPR of 1.38, 1.17, 1.01, 0.67, 0.46, and 0.25 respectively. For the botnet attack, the

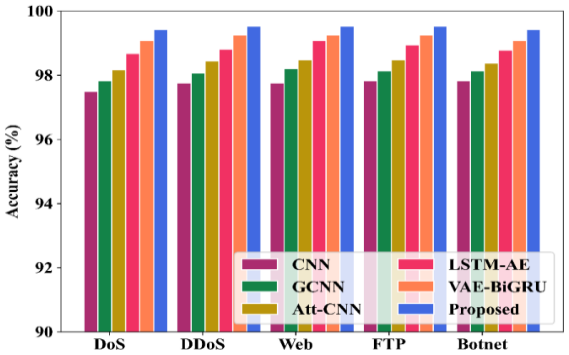


Figure 5: Accuracy Analysis for Varying Existing Schemes

conventional CNN, GCNN, Att-CNN, LSTM-AE, VAE-BiGRU, and suggested XAI-ZCAtt-GCNNtechniques obtained an FPR of 1.3, 1.13, 0.83, 0.67, 0.54, and 0.33 respectively.

Figure 7 indicates the NPV analysis for varying existing schemes. The graphical representation shows that the proposed XAI-ZCAtt-GCNNtechnique obtained better detection performance when associated with conventional schemes. For the DoS attack, the conventional CNN, GCNN, Att-CNN, LSTM-AE, VAE-BiGRU, and suggested XAI-

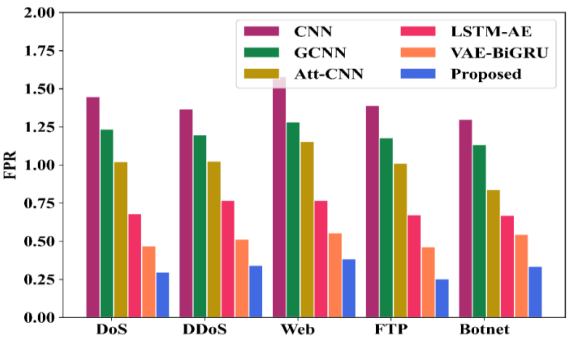


Figure 6: FPR Analysis for Varying Existing Schemes

ZCAtt-GCNNtechniques obtained an NPV of 0.985, 0.983, 0.987, 0.990, 0.993, and 0.995 respectively. For the DDoS attack, the conventional CNN, GCNN, Att-CNN, LSTM-AE, VAE-BiGRU, and suggested XAI-ZCAtt-GCNNtechniques obtained an NPV of 0.985, 0.987, 0.990, 0.992, 0.995, and 0.997 respectively. For the Web attack, the conventional CNN, GCNN, Att-CNN, LSTM-AE, VAE-BiGRU, and suggested XAI-ZCAtt-GCNNtechniques obtained an NPV of 0.987, 0.990, 0.992, 0.996, 0.996, 0.997 respectively. For the FTP attack, the conventional CNN, GCNN, Att-CNN, LSTM-AE, VAE-BiGRU, and suggested XAI-ZCAtt-GCNNtechniques obtained an NPV of 0.986, 0.988, 0.991, 0.993, 0.995, and 0.996 respectively. For the botnet attack, the conventional CNN, GCNN, Att-CNN, LSTM-AE, VAE-BiGRU, and suggested XAI-ZCAtt-GCNNtechniques obtained an NPV of 0.986, 0.988, 0.988, 0.991, 0.994, and 0.996 respectively.

Figure 8 indicates the G-mean analysis for varying existing schemes. The graphical representation shows that the proposed XAI-ZCAtt-GCNNtechnique obtained better detection performance when associated with conventional schemes. For the DoS attack, the conventional CNN, GCNN, Att-CNN, LSTM-AE, VAE-BiGRU, and suggested XAI-ZCAtt-GCNNtechniques obtained a G-mean of 95.9%, 96.43%, 96.96%, 7.72%, 98.42%, and 99.01% respectively. For the DDoS attack, the conventional CNN, GCNN, Att-CNN, LSTM-AE, VAE-BiGRU, and suggested XAI-ZCAtt-GCNNtechniques obtained a G-mean of 96.49%, 96.99%, 97.66%, 98.2%, 98.91%, and 99.33% respectively. For the Web attack, the conventional CNN, GCNN, Att-CNN, LSTM-AE, VAE-BiGRU, and suggested XAI-ZCAtt-GCNNtechniques obtained a G-mean of 96.8%, 97.45%, 97.93%, 98.87%, 98.97%, 99.39% respectively. For the FTP attack, the conventional CNN, GCNN, Att-CNN, LSTM-AE, VAE-BiGRU, and suggested XAI-ZCAtt-GCNNtechniques obtained a G-mean of 96.57%, 97.03%, 97.65%, 98.34%, 98.8%, 99.17% respectively. For the botnet attack, the conventional CNN, GCNN, Att-CNN, LSTM-AE, VAE-BiGRU, and suggested XAI-ZCAtt-GCNNtechniques obtained a G-mean of 96.39%, 96.92%, 97.07%, 97.87%, 98.47% and 99.03% respectively.

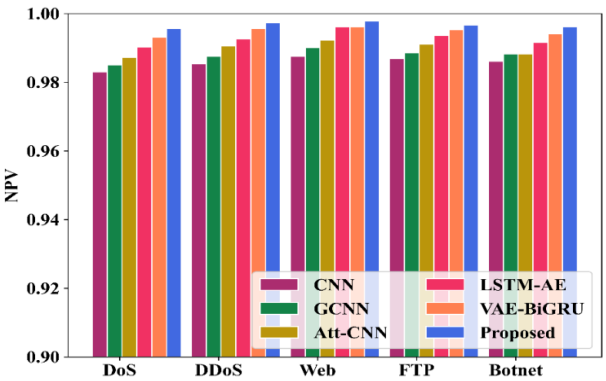


Figure 7: NPV Analysis for Varying Existing Schemes

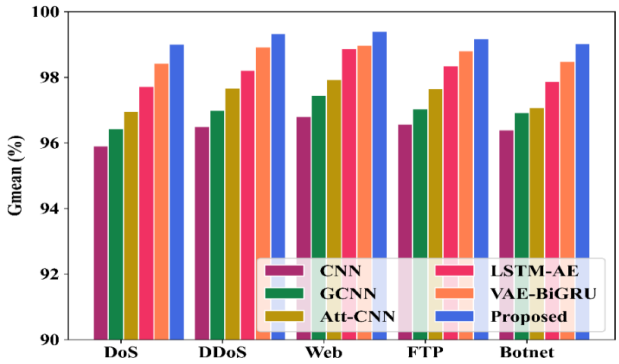


Figure 8: G-mean Analysis for Varying Existing Schemes

Figure 9 indicates the MCC analysis for varying existing schemes. The graphical representation shows that the proposed XAI-ZCAtt-GCNNtechnique obtained better detection performance when associated with conventional schemes. The existing CNN, GCNN, Att-CNN, LSTM-AE, VAE-BiGRU, and suggested XAI-ZCAtt-GCNNtechniques obtained an MCC of 92.91%, 93.97%, 94.95%, 96.43%, 97.45%, 98.38% respectively. The experimental outcome shows that the developed XAI-ZCAtt-GCNNtechnique is effective

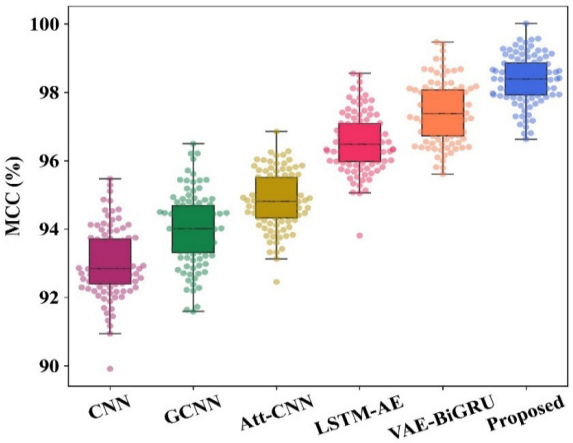


Figure 9: MCC Analysis for Varying Existing Schemes

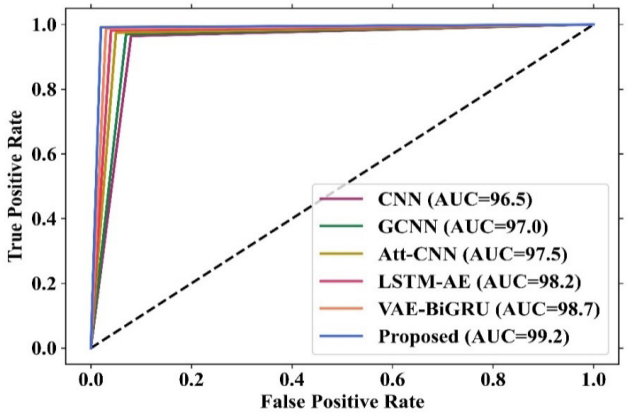


Figure 10: ROC Analysisof Different Techniques

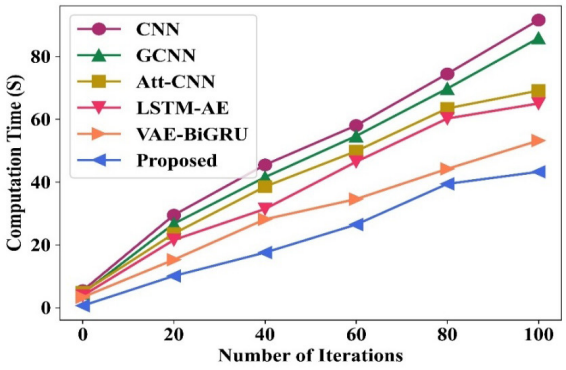


Figure 11: Computation Time Analysisof Varying Iterations

in identifying various network attacks with minimal error and computations.

The ROC curve graph in Figure 10 evaluates classification performance using the Area Under Curve (AUC) metric for different models. The CNN model achieves an AUC of 96.5, while GCNN performs slightly better with an AUC of 97.0. The Att-CNN model further improves the classification performance with an AUC of 97.5. The LSTM-AE model outperforms these with an AUC of 98.2, followed by the VAE-BiGRU model with an AUC of 98.7. The proposed model achieves the highest AUC of 99.2, indicating superior classification accuracy and robustness compared to other approaches.

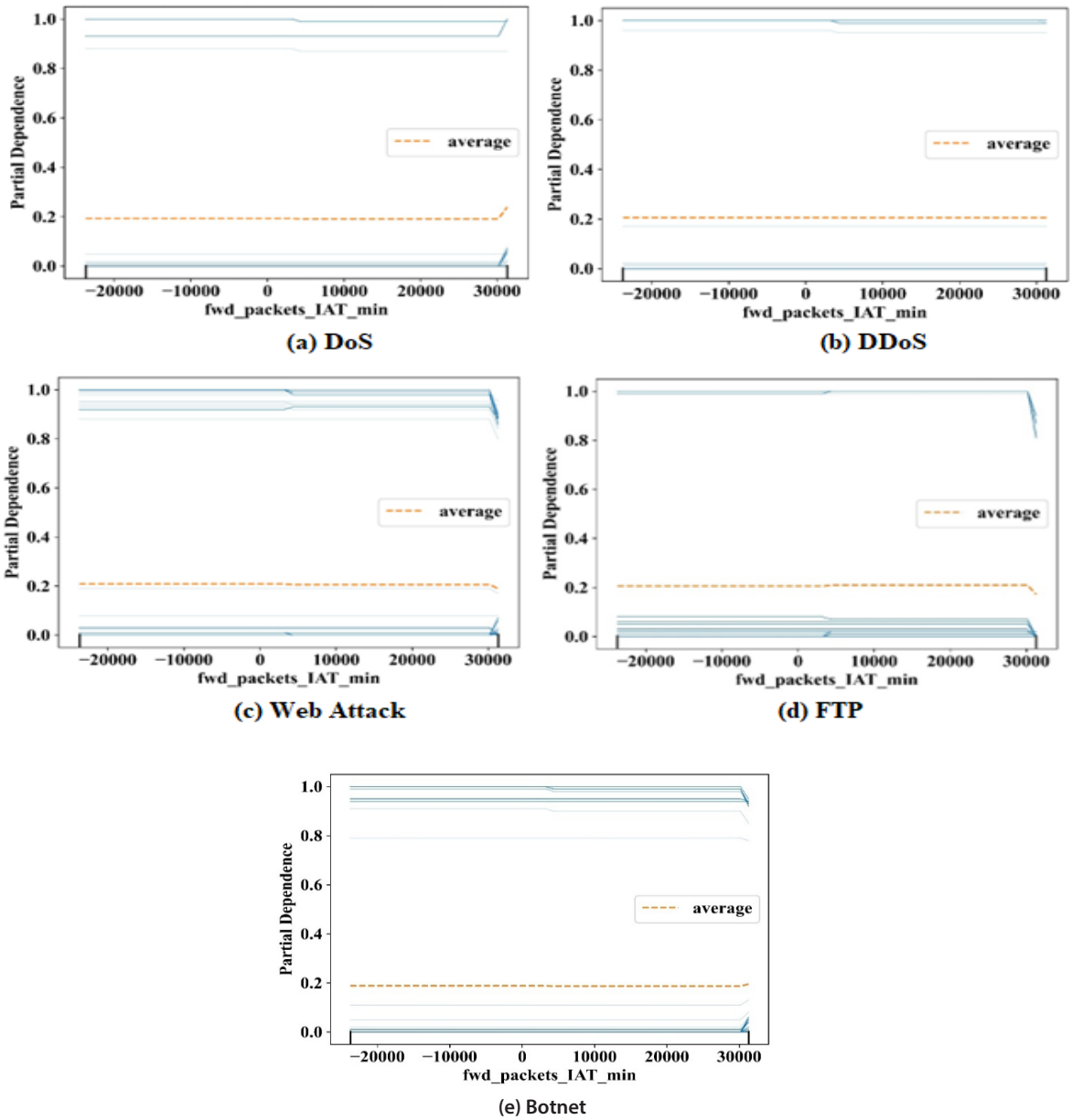
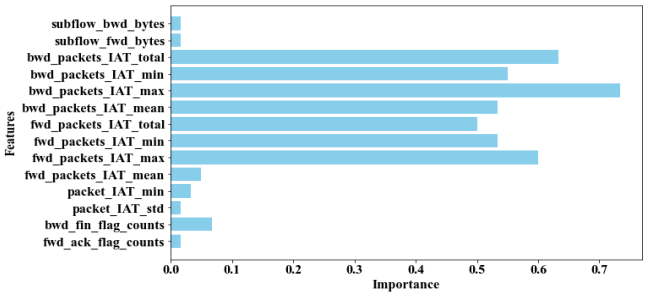
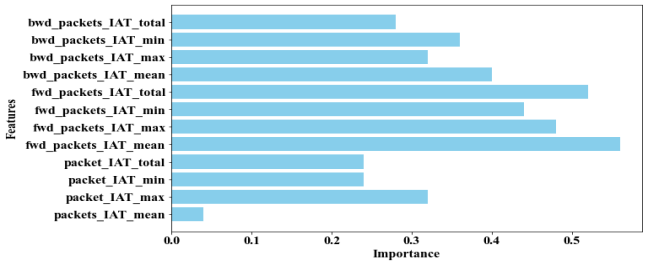


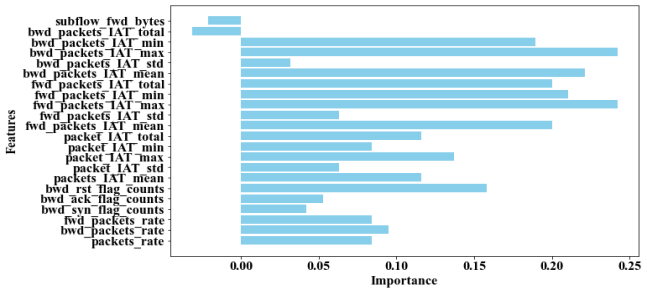
Figure 12: PDP-ICE-XAI Plot for Different Classes



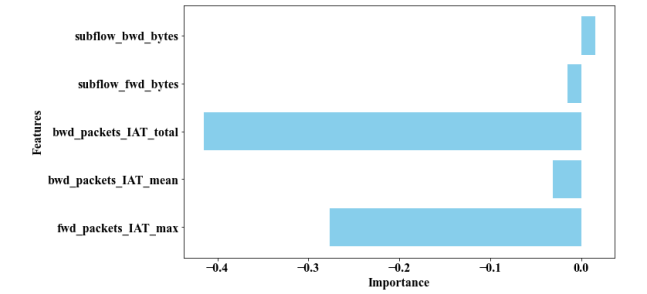
(a) DoS Attack



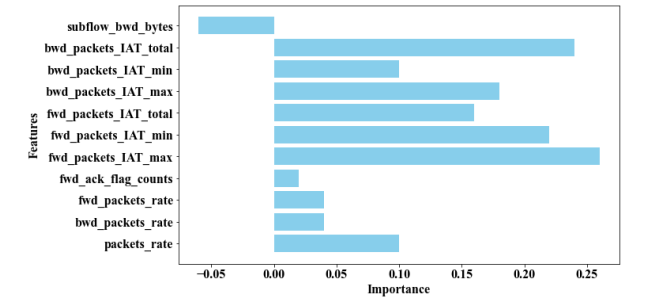
(b) DDoS Attack



(c) Web Attack



(d) FTP Attack



(e) Botnet Attack

Figure 13: PFI-XAI Plot for Different Classes

Figure 11 signifies thecomputation time analysisof varying iterations. At 10 iterations, the computation times are 8s for CNN, 10s for GCNN, 12s for Att-CNN, 14s for LSTM-AE, 15s for VAE-BiGRU, and 6s for the proposed model. As the number of iterations increases to 20, the times increase to 18s for CNN, 22s for GCNN, 25s for Att-CNN, 28s for LSTM-AE, 30s for VAE-BiGRU, and 12s for the proposed model. At 40 iterations, CNN requires 35s, GCNN takes 40s, Att-CNN takes 42s, LSTM-AE requires 45s, VAE-BiGRU takes 48s, while the proposed model completes in 25s. At 60 iterations, CNN reaches 50s, GCNN takes 58s, Att-CNN takes 60s, LSTM-AE requires 65s, VAE-BiGRU takes 68s, and the proposed model remains the most efficient at 38s. As training progresses to 80 iterations, CNN takes 65s, GCNN reaches 72s, Att-CNN takes 75s, LSTM-AE requires 80s, VAE-BiGRU takes 85s, while the proposed model completes in 50s. Finally, at 100 iterations, CNN reaches 78s, GCNN requires 85s, Att-CNN takes 88s, LSTM-AE needs 92s, VAE-BiGRU reaches 96s, and the proposed model achieves the lowest computation time at 62s.

XAI Performance Analysis

In this section, the obtained explanations are illustrated for the proposed XAI-ZCAtt-GCNN model using SHAP, LIME, and PFI model agnostic plots. A brief analysis of the obtained XAI plots is deliberated below:

PDP-ICE Analysis

The PDP-ICE plot in Figure 12 shows the relationship between fwd_packets_IAT_min and the model's partial dependence. In Figure 2(a-e), the solid lines represent individual conditional expectation (ICE) curves for different attack samples, while the dashed line represents the overall average partial dependence. This plot helps in understanding how a specific feature impacts model predictions on average while also capturing instance-level variations. The slight increase in the average dependence towards the right side indicates that larger values of fwd_packets_IAT_min contribute more positively to model predictions.

PFI XAI Analysis

The feature importance plot in Figure 13(a-e) shows the relative contribution of each feature based on Permutation Feature Importance (PFI). The subflow_bwd_bytes and subflow_fwd_bytes features have the highest importance, indicating their strong influence on model performance. Features like bwd_packets_IAT_min and fwd_packets_IAT_total also contribute significantly, while fwd_ack_flag_counts has the least impact. This analysis helps identify the most crucial features and can guide feature selection or further model optimization.

SHAP XAI Analysis

The SHAP Summary Plot in Figure 14 visualizes the contribution of each feature to the model's output using

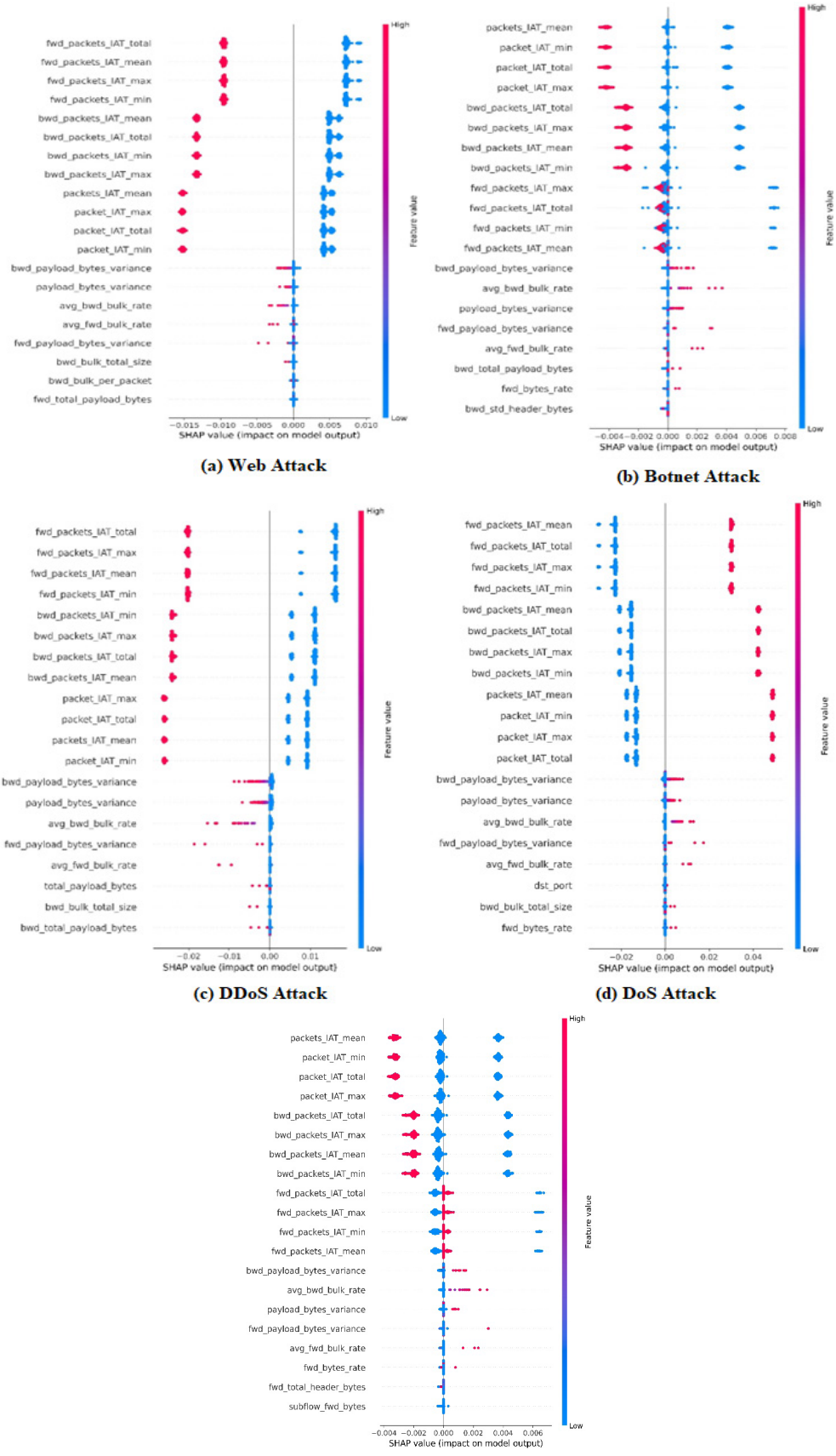


Figure 14: SHAP-XAI Plot for Different Classes

SHAP values. Each dot in Figure 14(a-e) represents an instance in the dataset, and its position on the x-axis indicates the SHAP value, showing the impact on the prediction. The color gradient from blue (low feature values) to red (high feature values) helps understand how different feature values influence predictions. Features like fwd_packets_IAT_total, fwd_packets_IAT_mean, and bwd_packets_IAT_min have a significant impact, indicating their strong contribution to model decisions. The positive or negative SHAP values denote whether increasing a particular feature increase or decreases the prediction probability.

Conclusion

The suggested framework presents an innovative XAI-DL framework for detecting and classifying multiclass cyber threats in Internet of Things (IoT) platforms. By leveraging the BCCC-CIC-IDS dataset, the proposed ZCAtt-GCNN, coupled with the PSN-based data normalization scheme and one-hot encoding process, effectively enhances the accuracy of cyberattack detection. The integration of SHAP, PDP-ICE, and PFI provides deeper insights into model decisions, improving interpretability and trust in the classification process. The extensive evaluation of the Python simulation platform demonstrates the efficiency of the proposed framework, achieving an accuracy of 99.48%, G-mean of 99.18%, MCC of 98.38%, and False Positive Rate (FPR) of 0.322, surpassing traditional methods. However, the potential generalization issue is that models trained on specific datasets may not perform optimally in real-world dynamic IoT environments with continuously evolving threats. Moreover, the model's performance could be affected by adversarial attacks, requiring additional mechanisms for robustness. In the future, the developed scheme will be extended by incorporating federated learning that can enhance data privacy by enabling decentralized training across multiple IoT nodes to handle zero-day attacks and advanced persistent threats (APTs).

References

Abbas, S., et al.(2024). Evaluating deep learning variants for cyber-attacks detection and multi-class classification in IoT networks, PeerJ Computer Science. https://doi.org/10.7717/peerj-cs.1793

Akgun, D., Hizal, S., & Cavusoglu, U. (2022). A new DDoS attacks intrusion detection model based on deep learning for Cybersecurity, Computers & Security, 18(1). 10.1016/j.cose.2022.102748.

Akilandeswari, R., & Malathi, S. (2022). Design and implementation of controlling with preventing DDOS attacks using bitcoin by Ethereum blockchain technology, Journal of Transportation Security, 15(3), 281–297. 10.1007/s12198-022-00245-x

Aslam, N., et al. (2022). Interpretable machine learning models for malicious domains detection using Explainable Artificial Intelligence (XAI), Sustainability, 14(12). https://doi.org/10.3390/su14127375.

Dasari., S., & Kaluri, R. (2024). An effective classification of DDoS attacks in a distributed network by adopting hierarchical machine learning and hyper parameters optimization techniques, IEEE Access. 10.1109/ACCESS.2024.3352281

Faheem, M.A., Kakolu, S., & Aslam, M. (2022).The role of explainable AI in Cybersecurity: Improving analyst trust in automated threat assessment systems, Iconic Research and Engineering Journals, 6(4), 173–182.

Keshk, M., et al. (2023). An explainable deep learning-enabled intrusion detection framework in IoT networks, Information Sciences, 639(7). 10.1016/j.ins.2023.119000.

Kuppusamy, K., & Malathi, S. (2011). An effective prevention of attacks using gI time frequency algorithm under DDoS, International Journal of Network Security and its Applications, 3(6). DOI : 10.5121/ijnsa.2011.3619

Kuppusamy, K., & Malathi, S. (2012). Prevention of attacks under DDoS using target customer behavior, International Journal of Computer Science Issues, 9(5).

Kumari, P & Jain, A.K. (2024). Timely detection of DDoS attacks in IoT with dimensionality reduction, Cluster Computing, 27(6),1-19. 10.1007/s10586-024-04392-9

Luan, F., Mu, X., & Yuan, S. (2024). Ghost Module Based Residual Mixture of Self-Attention and Convolution for Online Signature Verification, *Computers Materials & Continua*, 79(1).

Mahbooba, B., Timilsina, M., Sahal, R., & Serrano, M. (2021). Explainable artificial intelligence (XAI) to enhance trust management in intrusion detection systems using decision tree model, Complexity. 10.1155/2021/6634811

Malathi., S & Razool Begum, S. (2024). Enhancing trustworthiness among IoT network nodes with ensemble deep learning-based cyber attack detection, Expert System with Applications. https://doi.org/10.1016/j.eswa.2024.124528.

Mohsenabad, H. N., & Tut, M.A. (2024). Optimizing Cybersecurity attack detection in computer networks: A comparative analysis of bio-inspired optimization algorithms using the CSE-CIC-IDS 2018 dataset, Applied Sciences, 14(3). https://doi.org/10.3390/app14031044

Mughaid A., et al. (2022). An intelligent cyber security phishing detection system using deep learning techniques, Cluster Computing., 25(6). https://doi.org/10.1007/s10586-022-03604-4

Muthukumar, S., & Riyaz Ahamed, K. (2024). A novel framework of DDoS attack detection in network using hybrid heuristic deep learning approaches with attention mechanism, Journal of High Speed Network. DOI: 10.3233/JHS-230142

Park, J. H., Yotxay,S., Singh, S. K., & Park, J. H. (2024). PoAh-enabled federated learning architecture for DDoS attack detection in IoT networks, Hum.-Centric Computing and Information Science, 14(3). https://doi.org/10.22967/HCIS.2024.14.003

Rajak, A., & Tripathi, R. (2023). DL-SkLSTM approach for cyber security threats detection in 5G enabled IIoT, International Journal of Information Technology, 16(2). 10.1007/s41870-023-01651-7

Rizvi, F., et al. (2024). An evolutionary KNN model for DDoS assault detection using genetic algorithm-based optimization, Multimedia Tools and Applications, 83(35). 10.1007/s11042-024-18744-5

Sharma, B., et al. (2021). Explainable artificial intelligence for intrusion detection in IoT networks: A deep learning-based approach, Expert Systems with Applications. https://doi.org/10.1016/j.eswa.2023.121751

Shtayat, M.M., et al. (2023). An explainable ensemble deep learning

approach for intrusion detection in industrial Internet of Things, IEEE Access. 10.1109/ACCESS.2023.3323573

Tokareva, A.O., et.al. (2021). Normalization methods for reducing interbatch effect without quality control samples in liquid chromatography-mass spectrometry-based studies, *Analytical and Bioanalytical Chemistry*. https://doi.org/10.1007/s00216-021-03294-8

Vibhute, A.D., Patil, C.H., Mane, A.V & Kale, K.V. (2024). Towards detection of network anomalies using machine learning algorithms on the NSL-KDD benchmark datasets, Procedia Computer Science, 233(1), 960–969.

Wanda, P., & Hiswati, M. E. (2024). Belief-DDoS: stepping up DDoS attack detection model using DBN algorithm, International Journal of Information Technology, 16(1), 271–278.

Yamarthy A. K., & Koteswararao,C. (2024). MDepthNet based phishing attack detection using integrated deep learning methodologies for cyber security enhancement, Cluster Computing, 27(5). 10.1007/s10586-024-04313-w

Yaras, S., & Dener, M. (2024). IoT-based intrusion detection system using new hybrid deep learning algorithm, Electronics, 13(6).

Zebin, T., Rezvy, S., & Luo, Y. (2022). An explainable AI-based intrusion detection system for DNS over HTTPS (DoH) attacks, IEEE Transactions on Information Forensics Security, 17(1). 10.1109/TIFS.2022.3183390

Surendra Singh Bisht, et al. (2026). A Comparative and Hybrid Machine Learning Framework for IoT-Based Predictive Maintenance of Rotating Machinery, The Scientific Temper, 17(2).