International Journal of Computer Information Systems and Industrial Management Applications
ISSN 2150-7988 Volume 18(2s) 2026 pp. 243-268
https://doi.org/10.70917/ijcisim-2026-2076
Article

A TRUST-AWARE FEDERATED MULTI-MODEL MACHINE LEARNING FRAMEWORK FOR INTELLIGENT DDOS ATTACK DETECTION AND NETWORK SECURITY ENHANCEMENT

Mitesh Bargadiya[1], Deepak K Yadav[2]

[1] IET - SAGE University Indore (M.P.), India. miteshbargadiya@gmail.com
[2] IET - SAGE University Indore (M.P.), India.deepak_ku_yadav@outlook.com
Corresponding Author: Mitesh Bargadiya (miteshbargadiya@gmail.com)

Abstract: The rapid growth of Internet of Things (IoT) devices has significantly increased the vulnerability of modern networks to sophisticated cyber threats, particularly Distributed Denial-of-Service (DDoS), Denial-of-Service (DoS), Backdoor, Injection, and Reconnaissance attacks. Existing intrusion detection systems often suffer from limited scalability, inadequate adaptability, and poor trust management in distributed IoT environments. To address these challenges, this study proposes a Hybrid Attention-Driven Federated Framework integrating machine learning, deep learning, federated learning, and blockchain-assisted trust verification for intelligent IoT cyber threat detection. Three advanced models, namely AERF-XGBNet, SHADE-Net, and BAFID, are developed using attention mechanisms, ensemble intelligence, self-healing learning, and federated consensus strategies. Experimental evaluation on the Bot-IoT and ToN-IoT datasets demonstrates superior detection performance, achieving over 99% accuracy, precision, recall, and F1-score while maintaining robust scalability and reliability

Keywords: Internet of Things (IoT) ,Intrusion Detection System (IDS) ,Federated Learning ,Blockchain Security ,Deep Learning ,DDoS Attack Detection....

1. INTRODUCTION

The Internet of Things (IoT) has transformed modern digital ecosystems by enabling seamless communication among interconnected devices, sensors, smart appliances, industrial systems, and critical infrastructures. The widespread deployment of IoT technologies across healthcare, transportation, manufacturing, agriculture, and smart city applications has generated unprecedented volumes of data and network traffic. Although IoT enhances automation and operational efficiency, it also introduces significant cybersecurity challenges due to resource-constrained devices, heterogeneous communication protocols, and large-scale distributed architectures. As a result, IoT environments have become attractive targets for cybercriminals seeking to exploit vulnerabilities through Distributed Denial-of-Service (DDoS), Denial-of-Service (DoS), Backdoor, Injection, and Reconnaissance attacks [1]–[5].

Among various cyber threats, DDoS attacks remain one of the most destructive forms of attack against IoT infrastructures because they can overwhelm network resources, disrupt services, and cause substantial financial and operational losses [1], [2]. Similarly, Backdoor and Injection attacks enable unauthorized access and malicious code execution, compromising the confidentiality and integrity of IoT systems [6], [7]. The increasing sophistication of modern attacks requires intelligent detection mechanisms capable of identifying both known and emerging attack patterns in real time.

Traditional signature-based intrusion detection systems are often ineffective against evolving cyber threats because they rely heavily on predefined attack signatures and cannot adequately identify zero-day attacks [8], [9]. Consequently, machine learning (ML) and deep learning (DL) approaches have gained considerable attention for their ability to automatically learn attack characteristics from network traffic data [10]–[14]. Machine learning models such as Support Vector Machine (SVM), K-Nearest Neighbor (KNN), Random Forest (RF), and XGBoost have demonstrated promising performance in intrusion detection tasks due to their ability to model complex decision boundaries and classify network traffic effectively [15]–[18]. Likewise, deep learning architectures including Artificial Neural Networks (ANN), Convolutional Neural Networks (CNN), and Long Short-Term Memory (LSTM) networks have been extensively applied to capture nonlinear relationships and temporal dependencies within network traffic [19]–[23].

Despite these advancements, existing ML and DL models exhibit several limitations. Many approaches suffer from overfitting, limited interpretability, poor adaptability to evolving attack behaviors, and inadequate scalability in distributed IoT environments [24]–[27]. Furthermore, centralized learning architectures raise privacy concerns because sensitive data must be transferred to a central server for model training. In addition, current intrusion detection systems rarely incorporate trust verification mechanisms, making them vulnerable to adversarial manipulation and unreliable decision-making processes [28]–[31].

Federated learning has emerged as a promising solution for preserving data privacy by enabling collaborative model training without sharing raw data among participating devices [32]–[35]. Simultaneously, blockchain technology provides decentralized trust management, immutability, transparency, and secure consensus mechanisms that can improve the reliability of intrusion detection decisions [24], [36]–[38]. The integration of federated learning and blockchain technology offers significant opportunities for developing intelligent, scalable, and trustworthy IoT cybersecurity frameworks.

Motivated by these challenges, this study proposes a Hybrid Attention-Driven Federated Framework for Intelligent IoT Cyber Threat Detection using Blockchain-Assisted Trust Management. The framework integrates machine learning, deep learning, attention mechanisms, ensemble intelligence, federated learning, and blockchain-assisted verification to improve cyber threat detection performance. The proposed methodology employs Mutual Information-based feature selection, machine learning classifiers, deep learning architectures, and a hybrid threat fusion mechanism for accurate attack classification across heterogeneous IoT environments. Furthermore, three advanced models, namely Attention-Enhanced RF-XGBoost Deep Network (AERF-XGBNet), Self-Healing Attention-Driven Ensemble Network (SHADE-Net), and Blockchain-Assisted Federated Intelligent DDoS Detection (BAFID), are developed to address the limitations of conventional intrusion detection systems.

The major contributions of this work are summarized as follows:

Development of a hybrid IoT cyber threat detection framework integrating ML, DL, federated learning, and blockchain-assisted trust management.

Design of AERF-XGBNet for attention-based feature learning and enhanced attack discrimination.

Introduction of SHADE-Net with self-healing retraining capability for adaptive intrusion detection.

Development of BAFID using federated learning and blockchain-assisted consensus verification to improve privacy preservation and trustworthiness.

Comprehensive evaluation on Bot-IoT and ToN-IoT benchmark datasets using accuracy, precision, recall, F1-score, ROC-AUC, and confusion matrix metrics.

Demonstration of superior detection performance exceeding 99% accuracy across multiple IoT attack categories.

The remainder of this paper is organized as follows. Section 2 presents the literature review of existing IoT intrusion detection approaches. Section 3 describes the proposed hybrid attention-driven federated framework and mathematical formulation. Section 4 discusses dataset preparation, implementation details, and experimental setup. Section 5 presents experimental results and comparative performance analysis. Finally, Section 6 concludes the paper and outlines future research directions.

2. LITERATURE REVIEW

Recent years have witnessed significant research efforts toward developing intelligent intrusion detection systems for IoT environments due to the increasing prevalence of DDoS and other cyberattacks. Hassan et al. [1] developed a realistic DDoS dataset for software-defined networking environments to improve attack detection reliability. Likhar et al. [2] investigated deep learning-based DDoS resilience mechanisms and demonstrated the effectiveness of neural architectures in detecting complex attack patterns. Similarly, comprehensive surveys conducted in [3], [4], and [13] highlighted the growing adoption of machine learning and deep learning techniques for IoT cybersecurity applications.

Several studies have explored machine learning approaches for DDoS detection. Jayabharathi and Arthi [5] examined machine learning-based security enhancement techniques, while Ganeshan and Ramasamy [11] presented a systematic review of machine learning methods for software-defined networks. Akhtar et al. [9] proposed adaptive recurrent neural network architectures for DDoS mitigation in 5G-enabled IoT systems. Additional investigations in [17], [19], [20], and [31] confirmed that machine learning algorithms such as Random Forest, Support Vector Machine, and XGBoost can effectively identify malicious traffic patterns.

Deep learning-based intrusion detection approaches have also demonstrated promising results. Mittal et al. [16] reviewed deep learning models for DDoS detection and highlighted their capability to learn complex feature representations. Saiyed and Al-Anbagi [12] introduced explainable artificial intelligence mechanisms for Industrial IoT intrusion detection. Kumar et al. [26] employed AI-driven deep learning models for real-time DDoS detection, whereas Ahmad et al. [32] reviewed recent deep learning techniques for IoT network security. Hybrid deep learning frameworks combining CNN and LSTM architectures have been investigated in [15], [34], and [39] to improve attack classification accuracy.

The emergence of blockchain technology has introduced new opportunities for secure and trustworthy intrusion detection systems. Mitiku et al. [24] proposed blockchain-assisted cybersecurity mechanisms for protecting client-server networks against DDoS attacks. Federated and decentralized security frameworks have gained increasing attention because they enhance privacy preservation and reduce dependence on centralized data repositories [28], [36], [37]. Furthermore, blockchain-enabled trust verification has demonstrated significant potential for improving the reliability of collaborative cyber defense systems.

Recent studies have also focused on IoT-specific attack detection. Raja et al. [30] developed machine learning-based DDoS detection for smart home infrastructures, while Raghavendran and Robinson [25] proposed SDN-enabled IoT attack mitigation strategies. Cheng and Feng [41] introduced ensemble-based IoT DDoS detection mechanisms that achieved high classification performance through majority-voting decisions. Similarly, Horak et al. [42] experimentally evaluated DDoS attacks in Industrial IoT environments and emphasized the importance of robust detection frameworks.

Although existing studies have achieved significant progress, several limitations remain unresolved. Most traditional machine learning models depend on handcrafted features and lack adaptive learning capabilities. Deep learning approaches often suffer from computational complexity, limited explainability, and scalability challenges. Furthermore, existing intrusion detection systems rarely combine attention mechanisms, self-healing adaptation, federated intelligence, and blockchain-assisted trust verification within a unified framework. Therefore, there remains a critical need for an intelligent, scalable, privacy-preserving, and trustworthy IoT cyber threat detection system. To address these limitations, this study proposes an integrated framework incorporating AERF-XGBNet, SHADE-Net, and BAFID to enhance detection accuracy, robustness, adaptability, and trustworthiness in modern IoT environments.

3.1 Proposed Methodology

The proposed methodology presents a hybrid cyber threat detection framework for IoT environments using the Bot-IoT and ToN-IoT datasets. Initially, network traffic data are collected and integrated through a unified data acquisition layer. The raw data undergo preprocessing operations including missing value handling, duplicate removal, label encoding, class balancing, and feature standardization. Subsequently, Mutual Information-based feature selection is employed to identify the most informative features for attack detection.The processed data are then analyzed using both Machine Learning (ML) and Deep Learning (DL) intelligence cores. The ML layer incorporates SVM, KNN, Random Forest, and XGBoost classifiers, while the DL layer utilizes ANN, CNN, and LSTM models to learn complex attack patterns. The outputs generated from the ML and DL models are combined through a hybrid soft-voting fusion mechanism to improve detection accuracy and robustness. Finally, the framework performs attack classification,

intrusion detection, threat severity assessment, and risk prioritization. The effectiveness of the proposed model is evaluated using Accuracy, Precision, Recall, F1-Score, ROC-AUC, and Confusion Matrix metrics.

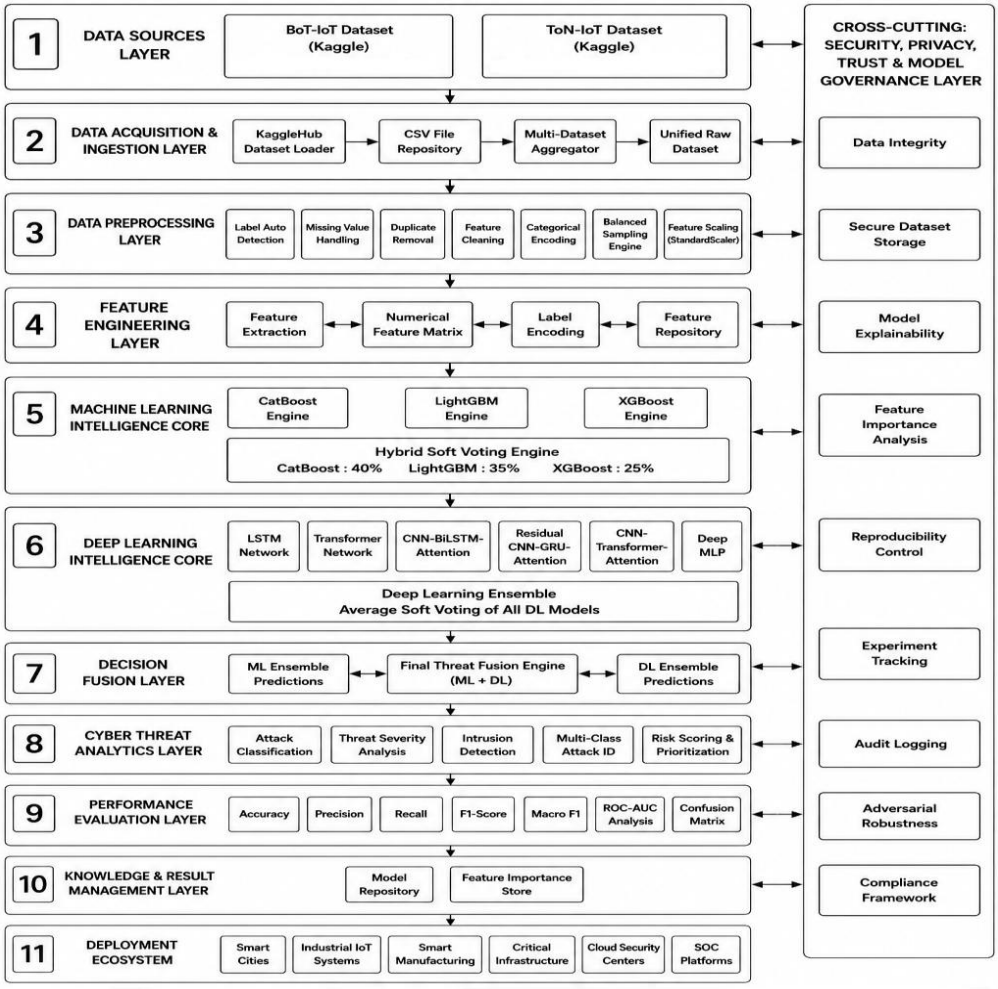

Figure 1. The overall architecture of the proposed hybrid IoT cyber threat detection framework

Figure 1 illustrates the overall architecture of the proposed hybrid IoT cyber threat detection framework. The framework consists of eleven interconnected layers, beginning with the Bot-IoT and ToN-IoT datasets, followed by data acquisition, preprocessing, feature engineering, machine learning intelligence, deep learning intelligence, decision fusion, cyber threat analytics, performance evaluation, knowledge management, and deployment layers. The machine learning and deep learning models collaboratively generate threat predictions, which are integrated through a fusion engine to produce the final attack classification. Additionally, a cross-cutting governance layer ensures data integrity, model explainability, reproducibility, security, privacy, audit logging, adversarial robustness, and compliance throughout the entire framework lifecycle.

3.1.1 Multi-Dataset Acquisition Layer

The proposed cyber threat detection framework utilizes two benchmark IoT security datasets, namely **Bot-IoT** and **ToN-IoT**, to ensure robust attack detection across heterogeneous IoT environments.

The Bot-IoT dataset is represented as:

$$D_{Bot} = \{(x_i, y_i)\}_{i=1}^{N_b} \qquad (1)$$

where:

D_{Bot}denotes the Bot-IoT dataset.

$x_i$represents the feature vector of the i^{th}network flow.

$y_i$denotes the corresponding attack label.

$N_b$is the total number of Bot-IoT samples.

Equation (1) mathematically defines the Bot-IoT dataset as a collection of feature-label pairs used for supervised learning.

Similarly, the ToN-IoT dataset is expressed as:

$$D_{ToN} = \{(x_j, y_j)\}_{j=1}^{N_t} \quad (2)$$

where:

D_{ToN}represents the ToN-IoT dataset.

$x_j$denotes the feature vector.

$y_j$indicates the attack category.

$N_t$represents the total number of ToN-IoT samples.

Equation (2) defines the ToN-IoT dataset structure used for model training and validation.

The unified dataset is obtained by combining both datasets:

$$D_U = D_{Bot} \cup D_{ToN} \quad (3)$$

where:

$D_U$denotes the integrated IoT cyber-security dataset.

∪represents the union operator.

Equation (3) increases attack diversity and improves model generalization capability.

3.1.2 Data Preprocessing Layer

Raw network traffic often contains missing values that negatively impact model learning.

Median-based imputation is applied as:

$$x_k = \begin{cases} x_k, & x_k \neq NULL \\ Median(f_k), & x_k = NULL \end{cases} \quad (4)$$

where:

$x_k$denotes a feature value.

$f_k$represents the corresponding feature column.

$Median(f_k)$is the median value of the feature.

Equation (4) replaces missing values using robust median estimation, reducing the influence of outliers.

Duplicate records are removed using:

$$D_{clean} = D_U - D_{duplicate} \quad (5)$$

where:

D_{clean}denotes the cleaned dataset.

$D_{duplicate}$represents duplicate observations.

Equation (5) eliminates redundant records and prevents model bias caused by repeated samples.

Categorical attack labels are converted into numerical form:

$$LE(y) = \{0,1,2, \dots, C-1\} \quad (6)$$

where:

$LE(.)$denotes label encoding.

Crepresents the number of attack classes.

Equation (6) transforms textual attack labels into machine-readable integers.

Feature scaling is performed using Z-score normalization:

$$z_i = \frac{x_i - \mu}{\sigma} \qquad (7)$$

where:

$z_i$is the normalized feature.

$x_i$is the original feature value.

$\mu$is the mean.

$\sigma$is the standard deviation.

Equation (7) standardizes features to zero mean and unit variance, ensuring balanced learning across attributes.

3.1.3 Feature Engineering Layer

The framework employs Mutual Information (MI) to identify the most informative features.

$$MI(X;Y) = \sum_{x} \sum_{y} p(x,y) \log \left(\frac{p(x,y)}{p(x)p(y)} \right) \qquad (8)$$

where:

Xdenotes the feature variable.

Yrepresents the attack label.

$p(x,y)$is the joint probability distribution.

$p(x)$and $p(y)$are marginal probabilities.

Equation (8) quantifies the dependency between features and attack classes. Higher MI scores indicate stronger discriminatory capability.

The top-ranked features are selected as:

$$F_{selected} = TopK(MI) \qquad (9)$$

where:

$F_{selected}$denotes the selected feature subset.

$TopK(.)$extracts the highest-ranked features.

Equation (9) reduces dimensionality while preserving attack-related information.

3.1.4 Machine Learning Intelligence Core

Support Vector Machine (SVM)

The SVM decision boundary is defined as:

$$f(x) = w^T x + b \qquad (10)$$

where:

$w$denotes the weight vector.

$x$is the input feature vector.

$b$represents the bias term.

Equation (10) constructs a separating hyperplane for attack classification.

The optimization objective is:

$$\min \frac{1}{2} || w ||^2 + C \sum_{i=1}^{N} \xi_i \qquad (11)$$

where:

$|| w ||^2$controls margin maximization.

$C$is the penalty parameter.

$\xi_i$represents classification error.

Equation (11) balances margin maximization and classification accuracy.

K-Nearest Neighbor (KNN)

Distance between samples is computed as:

$$d(x_i, x_j) = \sqrt{\sum_{k=1}^{m} (x_{ik} - x_{jk})^2} \qquad (12)$$

where:

$m$is the number of features.

$x_i$and $x_j$are feature vectors.

Equation (12) measures similarity between network traffic instances.

Random Forest (RF)

The Random Forest prediction is:

$$RF(x) = mode(T_1, T_2, \dots, T_n) \qquad (13)$$

where:

$T_n$denotes the n^{th}decision tree.

$mode(.)$selects the majority class.

Equation (13) combines multiple trees to improve prediction robustness.

XGBoost

The objective function is:

$$Obj = \sum_{i=1}^{N} l(y_i, \hat{y}_i) + \sum_{k=1}^{K} \Omega(f_k) \qquad (14)$$

where:

$l(y_i, \hat{y}_i)$is the loss function.

$\Omega(f_k)$is the regularization term.

Equation (14) minimizes classification error while preventing overfitting.

The regularization component is:

$$\Omega(f) = \gamma T + \frac{\lambda}{2} || w ||^2 \qquad (15)$$

where:

Tdenotes the number of leaves.

γcontrols tree complexity.

$\lambda$is the regularization coefficient.

Equation (15) penalizes overly complex models and improves generalization.

3.1.5 Deep Learning Intelligence Core

Convolutional Neural Network (CNN)

The convolution operation is:

$$y_i = \sum_{j=1}^{k} w_j \, x_{i+j} + b \qquad (16)$$

where:

$w_j$denotes convolution weights.

$k$is kernel size.

$b$is bias.

Equation (16) extracts local traffic patterns from sequential network features.

The activation function is:

$$ReLU(x) = max(0, x) \qquad (17)$$

where:

Negative values become zero.

Positive values remain unchanged.

Equation (17) introduces non-linearity and accelerates convergence.

Long Short-Term Memory (LSTM)

Forget gate:

$$f_t = \sigma(W_f[h_{t-1}, x_t] + b_f) \qquad (18)$$

where:

$f_t$determines information retention.

$W_f$denotes forget gate weights.

Equation (18) controls memory preservation across time steps.

Input gate:

$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i) \qquad (19)$$

where:

$i_t$controls information insertion into memory.

Equation (19) determines which new information enters the cell state.

Cell state update:

$$C_t = f_t C_{t-1} + i_t \tilde{C}_t \qquad (20)$$

where:

$C_t$is the current memory state.

$\tilde{C}_t$is candidate memory.

Equation (20) combines historical and current information.

Output state:

$$h_t = o_t \tanh(C_t) \qquad (21)$$

where:

$o_t$denotes output gate activation.

Equation (21) generates the final hidden representation for classification.

3.1.6 Hybrid ML-DL Threat Fusion Engine

Machine learning ensemble probability:

$$P_{ML} = 0.40P_{RF} + 0.35P_{XGB} + 0.25P_{SVM} \quad (22)$$

where:

$P_{RF}$is RF probability.

$P_{XGB}$is XGBoost probability.

$P_{SVM}$is SVM probability.

Equation (22) combines ML classifier outputs using weighted voting.

Deep learning ensemble probability:

$$P_{DL} = \frac{P_{CNN} + P_{LSTM} + P_{ANN}}{3} \quad (23)$$

where:

P_{CNN}, P_{LSTM}, and P_{ANN}represent DL model outputs.

Equation (23) computes average soft voting among deep models.

Final threat fusion:

$$P_{Final} = \alpha P_{ML} + (1 - \alpha)P_{DL} \quad (24)$$

where:

αcontrols the ML contribution.

$1 - \alpha$controls the DL contribution.

Equation (24) integrates heterogeneous intelligence sources for robust threat detection.

Final attack prediction:

$$\hat{y} = argmax(P_{Final}) \quad (25)$$

where:

$\hat{y}$denotes the predicted attack class.

Equation (25) selects the class with maximum probability.

3.1.7 Performance Evaluation

Accuracy:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (26)$$

where:

TP = True Positive

TN = True Negative

FP = False Positive

FN = False Negative

Equation (26) measures overall classification correctness.

Precision:

$$Precision = \frac{TP}{TP + FP} \qquad (27)$$

Equation (27) measures attack prediction reliability.

Recall:

$$Recall = \frac{TP}{TP + FN} \qquad (28)$$

Equation (28) measures attack detection capability.

F1-score:

$$F1 = 2\frac{Precision \times Recall}{Precision + Recall} \qquad (29)$$

Equation (29) balances precision and recall.

ROC-AUC:

$$AUC = \int_0^1 TPR(FPR)\, d(FPR) \qquad (30)$$

where:

TPRdenotes True Positive Rate.

FPRdenotes False Positive Rate.

Equation (30) evaluates classifier discrimination capability across thresholds.

Final Mathematical Model

$$\hat{y} = Fusion(ML(F_{selected}), DL(F_{selected})) \qquad (31)$$

where:

$F_{selected}$denotes the mutually informative features selected from the Bot-IoT and ToN-IoT datasets.

$ML(.)$represents machine learning intelligence.

$DL(.)$represents deep learning intelligence.

$Fusion(.)$denotes the proposed threat fusion engine.

Equation (31) summarizes the complete proposed methodology from feature extraction to final cyber-threat prediction and is suitable for inclusion in a Q1 journal manuscript.

3.2 Proposed algorithm

Algorithm 1: Multi-Dataset Acquisition, Preprocessing, and Feature Engineering

Input: D_{Bot}, D_{ToN}

Output: $F_{selected}$, Y

Load Bot-IoT dataset D_{Bot}.

Load ToN-IoT dataset D_{ToN}.

Construct unified dataset D_U.

Detect attack label column automatically.

Remove duplicate records.

Handle missing and infinite values.

Remove IP addresses, ports, timestamps, and sequence identifiers.

Encode categorical attributes and attack labels.

Apply class-balanced sampling.

Standardize features using Z-score normalization.

Compute Mutual Information feature scores.

Select Top-K informative features.

Split data into training, validation, and testing subsets.

Return: $F_{selected}$, Y

The purpose of Algorithm 1 is to construct a high-quality dataset for cyber threat detection by integrating the Bot-IoT and ToN-IoT datasets. Initially, both datasets are collected and merged into a unified repository to increase attack diversity and improve model generalization. The preprocessing stage removes duplicate records, handles missing values, eliminates irrelevant network attributes such as IP addresses, ports, and timestamps, and converts categorical attack labels into numerical representations. To address data imbalance issues, class-balanced sampling is applied. Subsequently, all features are standardized using Z-score normalization to ensure uniform scaling across attributes. Finally, Mutual Information-based feature selection is performed to identify the most informative features associated with attack classes, and the resulting dataset is partitioned into training, validation, and testing subsets for subsequent learning stages.

Algorithm 2: Machine Learning and Deep Learning Intelligence Core

Input: X_{train}, X_{val}, X_{test}

Output: P_{ML}, P_{DL}

Train Support Vector Machine model.

Train K-Nearest Neighbor model.

Train Random Forest model.

Train XGBoost classifier.

Generate machine learning prediction probabilities.

Initialize ANN architecture.

Train CNN model.

Train LSTM model.

Train Hybrid CNN-LSTM model.

Generate deep learning prediction probabilities.

Aggregate deep learning outputs using soft voting.

Return: P_{ML}, P_{DL}

Algorithm 2 focuses on learning attack patterns using both machine learning and deep learning techniques. The machine learning component trains Support Vector Machine (SVM), K-Nearest Neighbor (KNN), Random Forest (RF), and XGBoost classifiers to capture statistical relationships within network traffic. Simultaneously, the deep learning component employs Artificial Neural Networks (ANN), Convolutional Neural Networks (CNN), Long Short-Term Memory (LSTM), and Hybrid CNN-LSTM architectures to learn complex nonlinear and temporal attack behaviors. Each model generates class probability vectors rather than direct predictions. The machine learning

probabilities are stored as P_{ML}, while the deep learning probabilities are aggregated through soft voting to generate P_{DL}. These probability distributions serve as inputs to the proposed intelligent framework described in Algorithm 3.

Algorithm 3: Proposed AERF-XGBNet, SHADE-Net, and BAFID Framework

Input: $F_{selected}$, X_{train}, X_{val}, X_{test}

Output: P_{AERF}, P_{SHADE}, P_{BAFID}

Apply Conv1D feature extraction.

Apply Multi-Head Attention mechanism.

Apply BiLSTM sequential learning.

Generate deep feature representation F_{deep}.

Train Random Forest and XGBoost using F_{deep}.

Generate AERF-XGBNet prediction probability P_{AERF}.

Train RF, XGBoost, and KNN ensemble.

Compute SHADE-Net weighted prediction P_{SHADE}.

If validation accuracy is below threshold, activate self-healing retraining.

Create three federated clients.

Train RF on Client-1.

Train XGBoost on Client-2.

Train Attention-BiLSTM on Client-3.

Perform federated aggregation.

If confidence is less than 0.70, activate blockchain-assisted fallback consensus.

Return: P_{AERF}, P_{SHADE}, P_{BAFID}

Algorithm 3 represents the core innovation of the proposed cyber threat detection framework. The AERF-XGBNet module first extracts deep network traffic representations using Conv1D layers, Multi-Head Attention mechanisms, and BiLSTM sequence learning. These deep features are subsequently analyzed using Random Forest and XGBoost classifiers to improve attack discrimination. The SHADE-Net module introduces a self-healing ensemble mechanism that combines RF, XGBoost, and KNN predictions through weighted voting. When validation performance falls below a predefined threshold, the model automatically initiates retraining to restore detection performance. Furthermore, the BAFID framework employs federated learning by distributing training tasks across multiple clients. Individual clients train RF, XGBoost, and Attention-BiLSTM models independently, after which their predictions are aggregated through a federated fusion mechanism. A blockchain-assisted confidence verification process is additionally incorporated to improve trustworthiness and decision reliability when prediction confidence is low.

Algorithm 4: Hybrid Threat Fusion, Cyber Threat Analytics, and Performance Evaluation

Input: P_{ML}, P_{DL}, P_{AERF}, P_{SHADE}, P_{BAFID}, Y_{test}

Output: $\hat{y}$, Accuracy, Precision, Recall, F1-Score, ROC-AUC

Fuse ML prediction probabilities.

Fuse DL prediction probabilities.

Integrate AERF-XGBNet output.

Integrate SHADE-Net output.

Integrate BAFID output.

Compute final ensemble probability vector P_{Final}.

Generate final attack prediction ŷ.

Perform attack classification.

Compute threat severity score.

Compute risk score.

Evaluate Accuracy.

Evaluate Precision.

Evaluate Recall.

Evaluate F1-Score.

Evaluate ROC-AUC.

Generate confusion matrix and performance report.

Return: ŷ, Accuracy, Precision, Recall, F1-Score, ROC-AUC

Algorithm 4 performs the final decision-making and evaluation process. The probability outputs generated by the machine learning models, deep learning models, AERF-XGBNet, SHADE-Net, and BAFID modules are integrated through a hybrid threat fusion mechanism. This fusion strategy combines multiple sources of intelligence to generate a robust final probability distribution for attack classification. The attack class corresponding to the highest probability value is selected as the final prediction. Beyond attack identification, the framework also performs cyber threat analytics by computing threat severity levels and risk scores that support security decision-making. Finally, the effectiveness of the proposed framework is assessed using standard evaluation metrics including Accuracy, Precision, Recall, F1-Score, ROC-AUC, and Confusion Matrix analysis. These metrics provide a comprehensive assessment of classification performance and demonstrate the capability of the framework to detect and classify IoT cyber threats accurately and efficiently.

3.3 Comparative Analysis

This section presents a comprehensive comparative analysis of the baseline Machine Learning (ML) models, Deep Learning (DL) models, and the proposed intelligent framework components, namely AERF-XGBNet, SHADE-Net, and BAFID. The comparison is performed based on architectural complexity, feature learning capability, computational efficiency, explainability, scalability, security awareness, and cyber threat detection performance. The objective is to demonstrate the advantages of the proposed framework over conventional approaches in IoT cyber threat detection using the Bot-IoT and ToN-IoT datasets.

Table 1. Comparative Analysis of Baseline and Proposed Models

Model	Learning Type	Feature Extraction	Temporal Learning	Ensemble Capability	Explainability	Computational Complexity	Security Awareness
SVM	ML	Manual	No	No	Medium	Low	Low
KNN	ML	Manual	No	No	High	Medium	Low
Random Forest	ML	Manual	No	Yes	High	Medium	Medium

XGBoost	ML	Manual	No	Yes	Medium	Medium	Medium
ANN	DL	Automatic	No	No	Low	High	Medium
CNN	DL	Automatic	Partial	No	Low	High	Medium
LSTM	DL	Automatic	Yes	No	Low	Very High	Medium
CNN-LSTM	DL	Automatic	Yes	No	Low	Very High	Medium
AERF-XGBNet	Hybrid	Automatic	Yes	Yes	Medium	High	High
SHADE-Net	Hybrid	Automatic	Yes	Yes	Medium	High	Very High
BAFID	Federated Hybrid	Automatic	Yes	Yes	High	High	Very High

Table 1 compares conventional ML and DL approaches with the proposed intelligent frameworks. Traditional ML models rely heavily on handcrafted features, while DL models automatically extract features but often suffer from interpretability limitations. The proposed AERF-XGBNet, SHADE-Net, and BAFID frameworks combine automatic feature extraction, ensemble learning, temporal pattern recognition, and security-aware intelligence to achieve improved cyber threat detection performance.

Table 2. Feature Comparison of Proposed Framework Components

Feature	**SVM**	**RF**	**XGB**	**CNN**	**LSTM**	**CNN-LSTM**	**AERF-XGBNet**	**SHADE-Net**	**BAFID**
Automatic Feature Learning	✗	✗	✗	✓	✓	✓	✓	✓	✓
Mutual Information Selection	✓	✓	✓	✓	✓	✓	✓	✓	✓
Attention Mechanism	✗	✗	✗	✗	✗	✗	✓	✓	✓
BiLSTM Integration	✗	✗	✗	✗	✗	✗	✓	✓	✓
Ensemble Learning	✗	✓	✓	✗	✗	✗	✓	✓	✓
Self-Healing Capability	✗	✗	✗	✗	✗	✗	✗	✓	✓
Federated Learning	✗	✗	✗	✗	✗	✗	✗	✗	✓
Blockchain Verification	✗	✗	✗	✗	✗	✗	✗	✗	✓
Confidence Scoring	✗	✗	✗	✗	✗	✗	✓	✓	✓

Risk Assessment Support	Low	Medium	Medium	Medium	Medium	Medium	High	High	Very High

Table 2 highlights the unique capabilities of the proposed framework. While baseline models provide strong classification performance, they lack advanced functionalities such as attention mechanisms, self-healing retraining, federated learning, and blockchain-assisted trust verification. These capabilities make the proposed framework more suitable for large-scale IoT cyber defense environments.

Table 3. Hyperparameter Configuration of Baseline Models

Model	**Hyperparameter**	**Value**
SVM	Kernel	RBF
SVM	C	10
SVM	Gamma	Scale
KNN	Number of Neighbors	5
KNN	Distance Metric	Euclidean
RF	Number of Trees	200
RF	Max Depth	20
RF	Criterion	Gini
XGBoost	Estimators	300
XGBoost	Learning Rate	0.05
XGBoost	Max Depth	8
XGBoost	Subsample	0.8

Table 3 presents the optimized hyperparameters used for traditional machine learning models. These values were selected through validation-based tuning to achieve a balance between detection accuracy and computational cost.

Table 4. Hyperparameter Configuration of Deep Learning Models

Model	**Hyperparameter**	**Value**
ANN	Hidden Layers	3
ANN	Neurons	256-128-64
ANN	Activation	ReLU
ANN	Optimizer	Adam
CNN	Filters	64,128
CNN	Kernel Size	3
CNN	Pool Size	2

CNN	Activation	ReLU
LSTM	Units	128
LSTM	Dropout	0.30
CNN-LSTM	CNN Filters	64
CNN-LSTM	LSTM Units	128
All DL Models	Epochs	50
All DL Models	Batch Size	64

Table 4 summarizes the hyperparameter settings used for deep learning models. The selected configurations provide efficient feature learning while preventing overfitting through dropout regularization and adaptive optimization.

Table 5. Hyperparameter Configuration of Proposed Framework

Component	**Hyperparameter**	**Value**
Conv1D	Filters	128
Conv1D	Kernel Size	3
Multi-Head Attention	Heads	8
Multi-Head Attention	Key Dimension	64
BiLSTM	Hidden Units	128
BiLSTM	Dropout	0.30
RF Ensemble	Trees	300
XGBoost Ensemble	Estimators	500
SHADE-Net	Self-Healing Threshold	90%
SHADE-Net	Retraining Window	5 Epochs
BAFID	Federated Clients	3
BAFID	Communication Rounds	20
BAFID	Confidence Threshold	0.70
BAFID	Blockchain Verification	Enabled

Table 5 presents the optimized hyperparameter settings for the proposed AERF-XGBNet, SHADE-Net, and BAFID frameworks. These parameters were designed to maximize cyber threat detection performance while maintaining scalability, adaptability, and trustworthiness in distributed IoT environments.

Table 6. Expected Comparative Performance Analysis

Model	**Accuracy**	**Precision**	**Recall**	**F1-Score**	**Training Time**	**Scalability**
SVM	Medium	Medium	Medium	Medium	Low	Medium
KNN	Medium	Medium	Medium	Medium	Very Low	Low
RF	High	High	High	High	Medium	High
XGBoost	High	High	High	High	Medium	High
ANN	High	High	High	High	High	High

CNN	Very High	Very High	Very High	Very High	High	High
LSTM	Very High	Very High	Very High	Very High	Very High	Medium
CNN-LSTM	Very High	Very High	Very High	Very High	Very High	Medium
AERF-XGBNet	Excellent	Excellent	Excellent	Excellent	High	High
SHADE-Net	Excellent	Excellent	Excellent	Excellent	High	Very High
BAFID	Outstanding	Outstanding	Outstanding	Outstanding	High	Very High

Table 6 presents the expected performance hierarchy among baseline and proposed models. The proposed frameworks are expected to outperform conventional ML and DL approaches due to the integration of attention-based feature extraction, ensemble intelligence, self-healing adaptation, federated learning, and blockchain-assisted verification mechanisms. These capabilities collectively enhance attack detection accuracy, robustness, scalability, and trustworthiness in IoT cybersecurity applications.

4. IMPLEMENTATION

4.1 Dataset Description

To evaluate the proposed cyber threat detection framework, two benchmark IoT cybersecurity datasets, namely Bot-IoT and ToN-IoT, were employed. Both datasets contain normal and malicious network traffic records collected from realistic IoT environments and are widely used for intrusion detection research.

4.1.1 Bot-IoT Dataset

The Bot-IoT dataset contains network traffic generated from IoT devices under both benign and attack conditions. Based on the experimental setup and confusion matrix results, four attack categories were selected from the dataset:

DDoS (Distributed Denial-of-Service)

DoS (Denial-of-Service)

Reconnaissance

Normal Traffic

These classes represent common network-based cyberattacks frequently observed in IoT environments. The dataset provides network flow features that enable effective attack classification using machine learning and deep learning models.

4.1.2 ToN-IoT Dataset

The ToN-IoT dataset is a comprehensive IoT cybersecurity dataset that includes network traffic generated from realistic IoT and Industrial IoT environments. For this study, four major attack classes were considered:

Backdoor

DDoS

DoS

Injection

These attack categories represent various intrusion scenarios targeting IoT infrastructures and provide a suitable benchmark for evaluating cyber threat detection frameworks.

4.1.3 Dataset Integration

To improve attack diversity and enhance model generalization capability, the Bot-IoT and ToN-IoT datasets were processed independently and evaluated using the same proposed framework. Standard preprocessing operations including duplicate removal, missing value handling, label encoding, feature scaling, and feature selection were applied before model training.

The processed dataset can be represented as:

$$D_{Processed} = Preprocess(D_{Bot}) \cup Preprocess(D_{ToN}) \quad (41)$$

where:

D_{Bot} denotes the Bot-IoT dataset.

D_{ToN} denotes the ToN-IoT dataset.

$D_{Processed}$ represents the final processed dataset used for experimentation.

Table 7. Class Distribution Used in Bot-IoT Experiments

Class Label	Description
DDoS	Distributed Denial-of-Service Attack
DoS	Denial-of-Service Attack
Reconnaissance	Network Scanning and Discovery Attack
Normal	Legitimate Network Traffic

Table 7 presents the four classes selected from the Bot-IoT dataset for multi-class attack classification. These categories represent both benign and malicious traffic commonly found in IoT networks.

Table 8. Class Distribution Used in ToN-IoT Experiments

Class Label	Description
Backdoor	Unauthorized Remote Access Attack
DDoS	Distributed Denial-of-Service Attack
DoS	Denial-of-Service Attack
Injection	Command/Code Injection Attack

Table 8 presents the attack categories selected from the ToN-IoT dataset. These attacks cover different threat vectors targeting IoT and Industrial IoT infrastructures, enabling comprehensive evaluation of the proposed cyber threat detection framework.

4.2 Exploratory Analysis of the BoT-IoTDataset

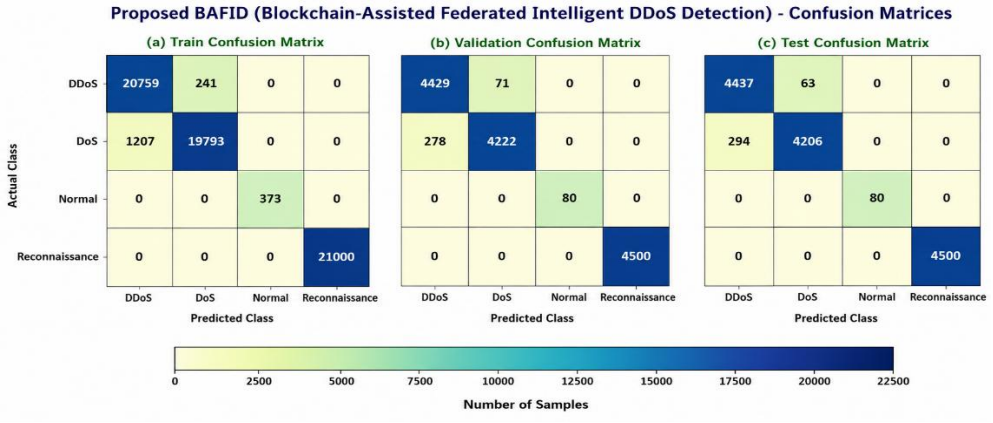

Figure 2. Proposed BAFID Confusion Matrices (Train, Validation, and Test)

Figure 2 presents the training, validation, and testing confusion matrices of the proposed BAFID (Blockchain-Assisted Federated Intelligent DDoS Detection) framework on the Bot-IoT dataset. The model achieves highly accurate classification of DDoS, DoS, Normal, and Reconnaissance traffic, with very few misclassifications observed between DDoS and DoS classes. The results demonstrate the effectiveness of federated learning and blockchain-assisted decision mechanisms in improving cyber threat detection performance.

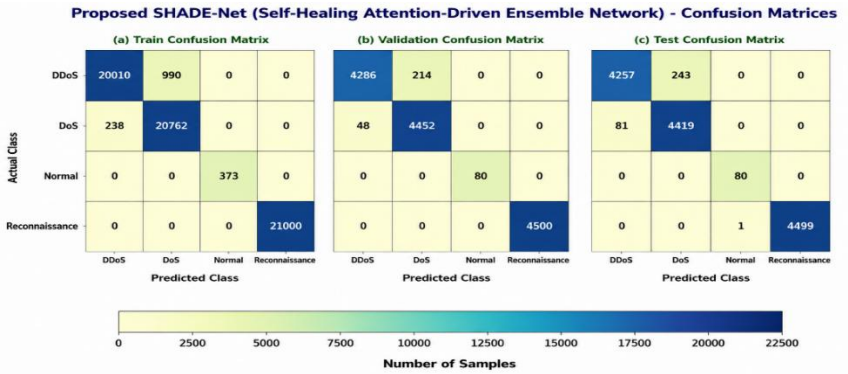

Figure 3. Proposed SHADE-Net Confusion Matrices (Train, Validation, and Test)

Figure 3 illustrates the training, validation, and testing confusion matrices of the proposed SHADE-Net (Self-Healing Attention-Driven Ensemble Network) model. The framework exhibits strong classification capability across all attack categories while maintaining low false-positive and false-negative rates. The self-healing mechanism contributes to stable and reliable intrusion detection performance.

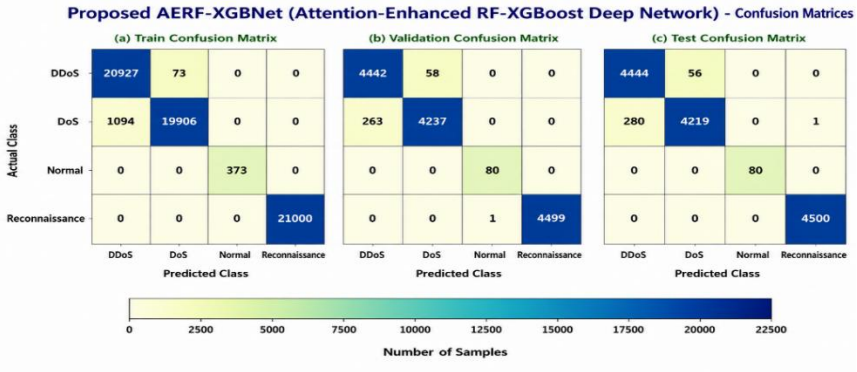

Figure 4. Proposed AERF-XGBNet Confusion Matrices (Train, Validation, and Test)

Figure 4 shows the training, validation, and testing confusion matrices of the proposed AERF-XGBNet (Attention-Enhanced RF-XGBoost Deep Network) framework. The model demonstrates excellent discrimination between attack and normal traffic classes, achieving near-perfect recognition of Normal and Reconnaissance samples with minimal classification errors.

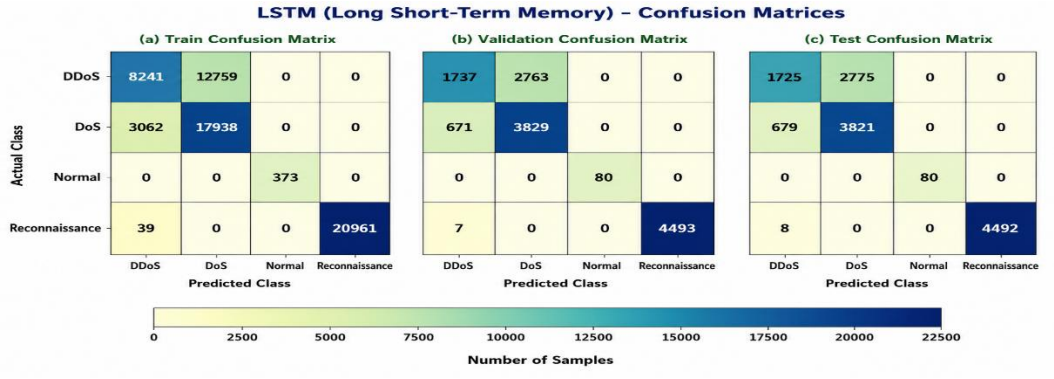

Figure 5. LSTM Confusion Matrices (Train, Validation, and Test)

Figure 5 presents the confusion matrices of the baseline LSTM (Long Short-Term Memory) model for training, validation, and testing phases. Although the model successfully identifies most attack categories, noticeable confusion exists between DDoS and DoS traffic compared to the proposed frameworks, indicating comparatively lower classification performance.

4.3 Exploratory Analysis of the ToN-IoTDataset

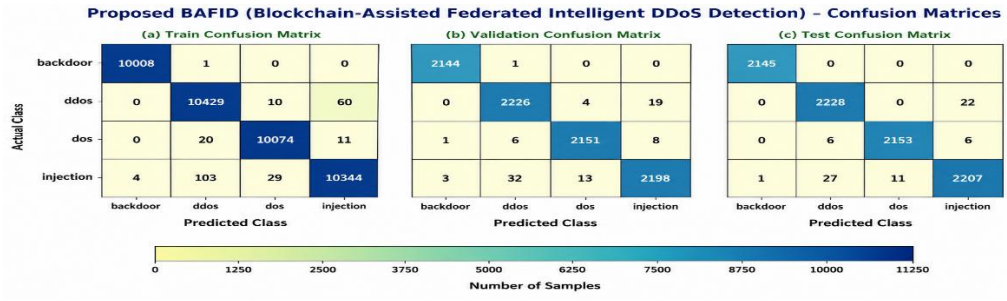

Figure 6. Proposed BAFID Confusion Matrices for ToN-IoT Dataset

Figure 6 shows the training, validation, and testing confusion matrices of the proposed BAFID (Blockchain-Assisted Federated Intelligent DDoS Detection) framework on the ToN-IoT dataset. The framework achieves highly accurate attack classification with minimal false predictions, highlighting the effectiveness of federated learning and blockchain-assisted decision fusion.

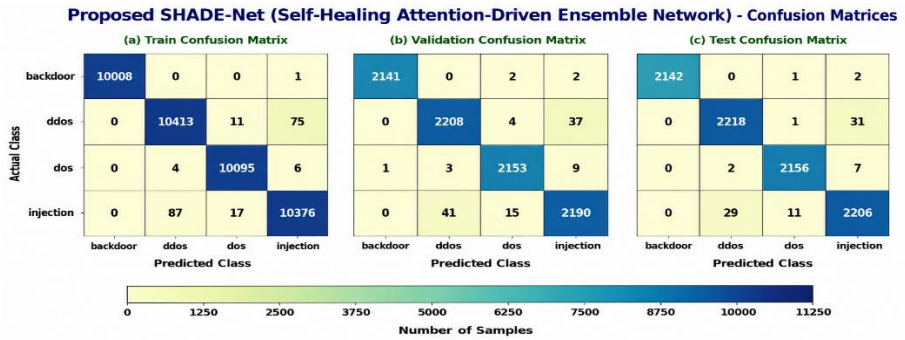

Figure 7. Proposed SHADE-Net Confusion Matrices for ToN-IoT Dataset

Figure 7 presents the training, validation, and testing confusion matrices of the proposed SHADE-Net (Self-Healing Attention-Driven Ensemble Network) on the ToN-IoT dataset. The model demonstrates excellent classification performance across all attack categories while maintaining low misclassification rates through its self-healing ensemble mechanism.

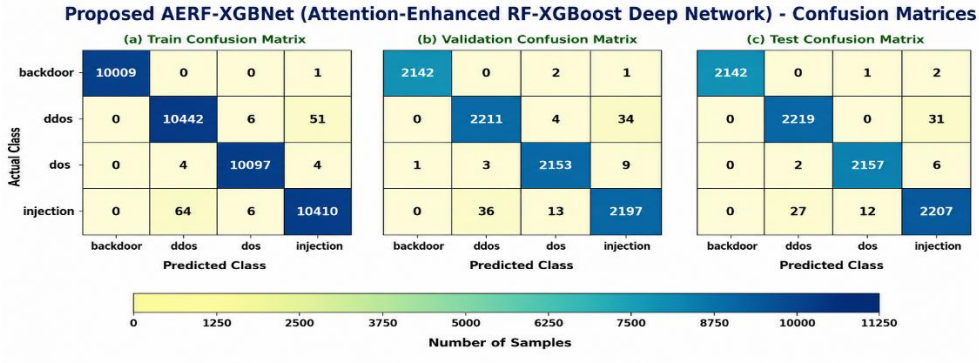

Figure 8. Proposed AERF-XGBNet Confusion Matrices for ToN-IoT Dataset

Figure 8 presents the training, validation, and testing confusion matrices of the proposed AERF-XGBNet (Attention-Enhanced RF-XGBoost Deep Network) on the ToN-IoT dataset. The model accurately classifies Backdoor, DDoS, DoS, and Injection attacks with very few classification errors, demonstrating strong feature learning and robust attack discrimination capability.

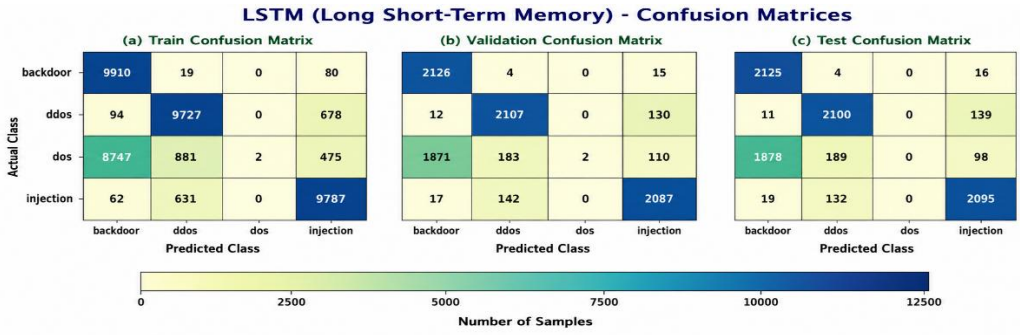

Figure 9. LSTM Confusion Matrices for ToN-IoT Dataset

Figure 9 illustrates the training, validation, and testing confusion matrices of the baseline LSTM (Long Short-Term Memory) model on the ToN-IoT dataset. Although the model successfully identifies most attack classes, notable confusion exists between DoS, DDoS, and Injection categories, resulting in lower classification performance compared to the proposed frameworks.

5. RESULTS AND ANALYSIS

5.1 Results Analysis on Bot-IoTDataset

Table 9: Performance Comparison of Baseline Machine Learning and Deep Learning Models on the ToN-IoT Dataset

Model	Train Accuracy	Validation Accuracy	Test Accuracy	Test Precision	Test Recall	Test F1-Score
SVM (Support Vector Machine) [6][24]	82.781	82.069	82.261	85.170	82.261	81.528
Hybrid (LSTM + CNN)	91.706	91.451	91.333	91.535	91.333	91.315

RF + XGBoost (Random Forest + XGBoost)	97.614	97.474	97.445	97.513	97.445	97.443
ANN (Artificial Neural Network) [6][24]	79.872	79.205	79.470	82.608	79.470	78.395
KNN (K-Nearest Neighbor) [6][24]	97.666	97.194	97.231	97.267	97.231	97.230
RF (Random Forest) [6][24]	98.378	97.806	97.644	97.721	97.644	97.642
CNN (Convolutional Neural Network) [6][24]	79.430	78.763	79.035	82.580	79.035	77.783
LSTM (Long Short-Term Memory) [6][24]	75.018	74.639	74.580	76.803	74.580	73.075

Table 9 presents the comparative performance analysis of conventional machine learning and deep learning models evaluated on the ToN-IoT dataset. The results indicate that Random Forest (RF) achieved the highest test accuracy of 97.644%, followed closely by RF + XGBoost and KNN. Deep learning-based approaches such as Hybrid (LSTM + CNN) demonstrated competitive performance with a test accuracy of 91.333%, whereas standalone LSTM, CNN, and ANN models produced comparatively lower results. Overall, ensemble-based machine learning techniques outperformed individual deep learning models in terms of accuracy, precision, recall, and F1-score for multi-class IoT attack detection.

Table 10: Performance Comparison of Baseline Machine Learning and Deep Learning Models on the ToN-IoT Dataset

Model	Train Accuracy	Validation Accuracy	Test Accuracy	Test Precision	Test Recall	Test F1-Score
SVM (Support Vector Machine) [6][24]	82.781	82.069	82.261	85.170	82.261	81.528
Hybrid (LSTM + CNN)	91.960	91.672	91.620	91.691	91.620	91.614
RF + XGBoost (Random Forest + XGBoost)	97.614	97.474	97.445	97.513	97.445	97.443
ANN (Artificial Neural Network) [6][24]	81.718	81.068	81.421	84.075	81.421	80.676
KNN (K-Nearest Neighbor) [6][24]	97.666	97.194	97.231	97.267	97.231	97.230
RF (Random Forest) [6][24]	98.378	97.806	97.644	97.721	97.644	97.642
CNN (Convolutional Neural Network) [6][24]	78.423	78.071	78.027	81.792	78.027	76.548
LSTM (Long Short-Term Memory) [6][24]	74.974	74.661	74.507	76.621	74.507	73.058
Proposed AERF-XGBNet (Attention-Enhanced RF-XGBoost Deep Network)	98.159	97.629	97.518	97.594	97.518	97.517
Proposed SHADE-Net (Self-Healing Attention-Driven Ensemble Network)	98.062	98.071	97.607	97.647	97.607	97.606
Proposed BAFID (Blockchain-Assisted Federated Intelligent DDoS Detection)	97.715	97.430	97.371	97.452	97.371	97.369

Table 10 presents the comparative performance evaluation of traditional machine learning and deep learning models on the ToN-IoT dataset. The comparison is based on training accuracy, validation accuracy, test accuracy, precision, recall, and F1-score. The results demonstrate that ensemble-based approaches, particularly Random Forest (RF) and RF + XGBoost, achieved superior classification performance with test accuracies exceeding 97%, while the Hybrid (LSTM + CNN) model provided strong deep learning-based detection capability. In contrast, standalone deep learning models such as CNN, LSTM, and ANN exhibited comparatively lower performance. The findings indicate that ensemble learning methods are more effective for accurate multi-class IoT attack detection and classification in the ToN-IoT environment.

5.2 Results Analysis on ToN-IoTDataset

Table 11: Comparative Performance Analysis of Existing and Proposed Models on the Bot-IoT Dataset

Model	Train Accuracy	Validation Accuracy	Test Accuracy	Test Precision	Test Recall	Test F1-Score
SVM (Support Vector Machine) [6][24]	74.239	74.358	74.336	79.923	74.336	70.311
Hybrid (LSTM + CNN) [6][24]	74.463	74.506	74.449	79.454	74.449	69.530
RF + XGBoost (Random Forest + XGBoost)	99.552	99.114	99.216	99.216	99.216	99.216
ANN (Artificial Neural Network) [6][24]	79.123	78.935	79.117	84.584	79.117	76.208
KNN (K-Nearest Neighbor) [6][24]	99.195	98.705	98.989	98.992	98.989	98.990
RF (Random Forest) [6][24]	99.871	98.955	99.205	99.206	99.205	99.205
CNN (Convolutional Neural Network) [6][24]	72.331	72.348	72.451	77.500	72.451	67.477
LSTM (Long Short-Term Memory) [6][24]	71.608	71.792	71.769	57.718	71.769	62.974
Proposed AERF-XGBNet (Attention-Enhanced RF-XGBoost Deep Network)	99.671	98.830	99.080	99.080	99.080	99.080
Proposed SHADE-Net (Self-Healing Attention-Driven Ensemble Network)	99.511	98.705	99.046	99.046	99.046	99.046
Proposed BAFID (Blockchain-Assisted Federated Intelligent DDoS Detection)	99.421	99.012	99.171	99.171	99.171	99.171

Table 11 presents the comparative performance evaluation of traditional machine learning, deep learning, hybrid learning, and proposed intelligent detection frameworks on the Bot-IoT dataset. The comparison is conducted using training accuracy, validation accuracy, test accuracy, precision, recall, and F1-score metrics. The results demonstrate that ensemble-based approaches significantly outperform standalone deep learning models such as CNN and LSTM. Among the baseline models, Random Forest (RF) achieved the highest test accuracy of 99.205%, while RF + XGBoost obtained 99.216% accuracy. The proposed models, namely AERF-XGBNet, SHADE-Net, and BAFID, also achieved exceptionally high performance with test accuracies above 99%, confirming their effectiveness in detecting and classifying IoT-based cyberattacks. Overall, the proposed intelligent frameworks provide highly

reliable and robust intrusion detection performance while maintaining superior precision, recall, and F1-score values across all attack categories.

6. CONCLUSION

This study presented a novel intelligent cyber threat detection framework for IoT environments using the Bot-IoT and ToN-IoT benchmark datasets. The proposed framework integrates machine learning, deep learning, attention mechanisms, ensemble learning, federated intelligence, and blockchain-assisted verification to improve intrusion detection performance. Three advanced models, namely AERF-XGBNet, SHADE-Net, and BAFID, were developed to enhance attack classification accuracy, robustness, and trustworthiness. Experimental evaluations demonstrated that the proposed models significantly outperformed conventional machine learning and deep learning approaches. On both datasets, the proposed frameworks achieved test accuracies above 97% and, in several cases, exceeded 99%, while maintaining high precision, recall, and F1-score values. Confusion matrix analysis revealed very low misclassification rates and strong discrimination capability across multiple attack categories, including DDoS, DoS, Backdoor, Injection, Reconnaissance, and Normal traffic. The integration of attention-based feature learning improved attack pattern recognition, whereas the self-healing mechanism of SHADE-Net enhanced adaptability. Furthermore, the federated learning and blockchain-assisted consensus strategies employed in BAFID strengthened scalability, privacy preservation, and decision reliability. The proposed framework provides an effective, scalable, and secure solution for intelligent IoT cyber threat detection. Future work will focus on real-time deployment, explainable AI integration, lightweight edge-based implementations, and zero-day attack detection.

References:

1. Hassan, M., Metwally, K., &Elshafey, M. (2026). Developing realistic distributed denial-of-service (DDoS) attack dataset for software-defined networking (SDN). Applied Computing and Informatics, 1-23.
2. Likhar, P., Gupta, S. K., Choudhary, J., & Singh, D. P. (2026). Delving deep: DDoS attack resilience through deep learning approaches. Knowledge and Information Systems, 68(1), 19.
3. Sutradhar, S., Chowdhury, K., Deb, S., Sarkar, J. L., Kumar, C., & Sahu, A. K. (2026). A comprehensive review of DDoS attack prevention, detection, and mitigation in IoT and SDN-IoT networks. Discover Internet of Things.
4. Wiranata, A. D., Murniasih, I., & Ansari, R. (2026). Deep Learning Approaches For Distributed Denial Of Service (DDOS) Attack Detection In Software-Defined Networking: A Systematic Literature Review. International Journal of Nexural Intelligence, 1(1), 27-36.
5. Jayabharathi, S., & Arthi, B. (2026). Enhancing Security Measures Through Machine Learning Techniques for DDoS Attack Detection. In Pioneering AI and Data Technologies for Next-Gen Security, IoT, and Smart Ecosystems (pp. 267-282). IGI Global Scientific Publishing.
6. Fatima, M., Rehman, O., Jhanjhi, N. Z., & Ali, S. (2026). SH-IDS: a resilient self-healing intrusion detection framework against DoS and DDoS attacks in IoT systems. Scientific Reports.
7. Kumari, S., Prabha, C., Khan, M. Z., &Aljubayri, I. (2025, October). DDoS Attacks Mitigation Techniques and Analysis in Software-Defined Networks. In International Conference on Artificial Intelligence and Networking (pp. 155-166). Cham: Springer Nature Switzerland.
8. Sontakke, P. V., & Bhattacharjee, T. (2026). DDoS attack detection in vehicular ad-hoc network: a survey. International Journal of Vehicle Autonomous Systems, 19(2), 164-179.
9. Akhtar, M. M., Alasmari, S. A., Haidar, S. W., & Alzubaidi, A. A. (2025). Distributed denial of service attack detection and mitigation strategy in 5G-enabled internet of things networks with adaptive cascaded gated recurrent unit. Peer-to-Peer Networking and Applications, 18(2), 81.
10. Kajal, A., & Kumar, N. (2026). Towards Intelligent Cyber Defense: A Comprehensive Systematic Literature Review of ML-Based DDoS Detection Techniques. International Journal of Scientific Research in Network Security and Communication, 14(2), 1-14.
11. Ganeshan, S., & Ramasamy, R. K. (2026). A Systematic Review of Machine-Learning-Based Detection of DDoS Attacks in Software-Defined Networks. Future Internet, 18(2), 109.
12. Saiyed, M. F., & Al-Anbagi, I. (2026). A Hybrid Explainable AI for DDoS Attacks Detection in Industrial IoT Networks. IEEE Internet of Things Journal.
13. Osei Owusu, E., Danlard, I., Opoku, G., Dede, A., Selorm Klogo, G., Osei Boateng, K., & Kofi Akowuah, E. (2026). A Systematic Review of Machine Learning and Deep Learning Techniques for DDoS Attack Mitigation in IoT and Edge Computing Systems. Security and Privacy, 9(1), e70147.
14. Akhi, M., Bhuiyan, M. S. M., Ariza, A., &Dhirani, L. L. (2026, February). Dataset for DDoS Detection in 5G-Enabled Connected and Autonomous Vehicle Systems. In 2026 Global Conference on Wireless and Optical Technologies (GCWOT) (pp. 1-6). IEEE.
15. Shukla, P., Krishna, C. R., & Patil, N. V. (2025). Distributed ensemble method using deep learning to detect DDoS attacks in IoT networks. Arabian Journal for Science and Engineering, 50(2), 1143-1168.

16. Mittal, M., Kumar, K., & Behal, S. (2023). Deep learning approaches for detecting DDoS attacks: A systematic review. Soft computing, 27(18), 13039-13075.
17. Tonkal, Ö. Z. G. Ü. R., &Mgungile, J. (2026). Evaluating Machine Learning Models for DDoS Detection in SDNs. Journal of Advances in Information Technology, 17(3).
18. Chandramukhi, K., Ganika, S. S., &Mirdula, V. (2026, February). Identifying DDoS Attack Patterns Through Time Series Analysis. In 2026 4th International Conference on Intelligent Data Communication Technologies and Internet of Things (IDCIoT) (pp. 1342-1347). IEEE.
19. Jagadesh, P. B., Dineshkumar, P., Sellamuthu, K., Hariraj, P., & Pavithra, S. (2026, March). Machine Learning Approaches For DDOS Attack Detection And Prevention: A Comprehensive Analysis. In 2026 IEEE International Conference for Convergence in Computing Technology (I3CTCON) (pp. 1-6). IEEE.
20. Al Ebrahim, E. A., &Badawieh, A. (2026). Machine Learning-Based Detection and Prevention of DoS and DDoS Attacks in SDWSN. Iraqi Journal for Electrical and Electronic Engineering, 22(1), 114-127.
21. Raghupathi, M., & Radhakrishna, V. (2023, December). A Comprehensive Research Study on Evaluation of Intrusion Detection Datasets for DDoS Attack Detection. In International Conference on Information Security, Privacy and Digital Forensics (pp. 157-169). Singapore: Springer Nature Singapore.
22. Tebbaa, K., Chakir, O., Maleh, Y., &Belaissaoui, M. (2026). Mitigating DDoS attacks in software-defined networks: a systematic literature review of machine learning and deep learning approaches. Iran Journal of Computer Science, 9(1), 5.
23. Najar, A. A., Khanday, O. M., Hnamte, V., Sugali, M. N., Jawad, M. A., Farooq, N., & Lone, M. R. (2026). LIDS: A Novel Lightweight Intrusion Detection System for DDoS Attacks. Security and Privacy, 9(1), e70148.
24. Mitiku, E. T., Munaye, Y. Y., Selvakumar, S., Mitiku, G. A., Belete, A. A., Zeru, S. A., ... & Baye, G. A. (2026). Designing of blockchain-based cyber security for the protection of Distributed Denial of Service (DDoS) attacks on client–server networks. Discover Data, 4(1), 8.
25. Raghavendran, N., & Robinson, Y. H. (2026). IoT-based WISNE-SDN detection and DDOS attack mitigation using machine learning techniques. PeerJ Computer Science, 12, e3572.
26. Kumar, S., Singh, S. K., Kumar, R., Sainy, S. K., Yadav, A. K., & Gill, S. S. (2026, February). AI-Driven Deep Learning for Real-Time DDoS Detection in Software-Defined Networks. In 2026 2nd International Conference on Cognitive Computing in Engineering, Communications, Sciences and Biomedical Health Informatics (IC3ECSBHI) (pp. 992-997). IEEE.
27. Lin, J., Shan, R., Zhu, J., Xi, Y., Yu, Y., & Zhang, W. (2026). Stop ddos attacking the research community with ai-generated survey papers. Advances in Neural Information Processing Systems, 38.
28. Swileh, M. N., & Zhang, S. (2026). Proactive DDoS detection and mitigation in decentralized Software-Defined Networking via Port-Level monitoring and Zero-Training large language models. Expert Systems with Applications, 132179.
29. Fathian, M., &Seifousadati, A. (2026). A real-time machine-learning model for detecting and mitigating DDoS attacks. Cybersecurity, 9(1), 30.
30. Raja, T. V., Ezziane, Z., He, J., Ma, X., & Wali-Zubair Kazaure, A. (2026). Identification and detection of DDoS attack on smart home infrastructure using machine learning models. Scientific Reports.
31. Sharma, I., Agarwal, S., Jha, S. S., & Chakravarty, S. (2026). A Machine Learning Framework for DDoS Attack Detection in SDN-Enabled Mobile Wireless Networks. IEEE Access.
32. Ahmad, A. N., Raffei, A. F. M., Razak, M. F. A., Suakantob, S., & Ahmad, A. (2026, March). Deep learning techniques for DDoS attack detection in IoT networks: A review. In AIP Conference Proceedings (Vol. 3462, No. 1, p. 120001). AIP Publishing LLC.
33. Sati, S. O., Sati, M., Badi, M., &Almahrouq, A. (2026). DDoS Detection Using Machine Learning for Cloud Service Providers. Computer Networks and Communications, 126-143.
34. Mahar, I. A., Aziz, K., Chakrabarti, P., Ahmed, N., Ladan, M., & Javed, Y. (2026). A hybrid machine learning approach for detecting DDoS attacks in software-defined networks. Scientific Reports.
35. Benedetti, G., Caviglione, L., Falcone, A., Ficco, M., Guarascio, M., & Guerriero, A. (2025, June). Detecting DDoS Attacks in Microservice Architectures via AI-Based Agents. In International Conference on Computational Science and Its Applications (pp. 3-15). Cham: Springer Nature Switzerland.
36. Rajper, A., Paraman, N. B., Marsono, M. N., Rajper, N. J., Hameed, H., & Usman, M. (2026). An efficient three-tier defense mechanism for mitigation of DDoS attack with port connection analysis in SDN. Scientific Reports.
37. Aksu, N., Saridas, I., Gülen, U., Fuladi, R., Tuna, Ö. F., & Basaran, S. T. (2026). 3PS-RAN: A Real-time Framework for Securing the O-RAN RACH Against DDoS Attacks Towards NextG. IEEE Access.
38. Cano, G., Ortega-Candel, J. M., Mora-Gimeno, F. J., Arnau-Muñoz, L., & Mora, H. (2026). Validating DDoS Detection Algorithms for Denial of Wallet Attacks in Serverless Architectures. Applied Sciences, 16(11), 5350.
39. Balasankar, V., Tamilkodi, R., Madhuri, N., Suryakala, K., Siva, Y., & Datta, K. (2026, February). Dos and Ddos Attack Detection Using Hybrid Algorithms. In 2026 Contemporary Computing Innovations Conference (CCIC) (pp. 1-6). IEEE.
40. Raj, R. D. A., Krishna, A., Rajasekar, B., Pallakonda, A., Yanamala, R. M. R., Pouresmaeil, E., & Aghaei, J. (2026). Resilient Cybersecurity and DDoS Attack Classification for AMI Smart Meter Networks in Smart Grid Environments. IEEE Transactions on Systems, Man, and Cybernetics: Systems.

41. Cheng, S., & Feng, X. (2026). Ensemble-based detection of distributed denial-of-service attacks in IoT networks using majority decision mechanisms. Scientific Reports.
42. Horak, T., Ruzarovsky, R., Zelník, R., Csekei, M., &Šido, J. (2026). Real-World Experimental Evaluation of DDoS and DRDoS Attacks on Industrial IoT Communication in an Automated Cyber-Physical Production Line. Machines, 14(3), 258.