Journal of Artificial Intelligence and Technology, 2026, 6, 432-440
https://doi.org/10.37965/jait.2026.1135

ISTP
RESEARCH ARTICLE

SmartCPS-ADAPT: An Intrusion Detection System for Cyber-Physical Systems

Upasana Mahajan,[1] J. Somasekar,[1] and Vikram Neerugatti[2]
[1]Department of Computer Science and Engineering, Faculty of Engineering and Technology (FET), Jain (Deemed-to-be University), Bengaluru, Karnataka-562112, India
[2]Department of Computer Science and Engineering (IoT), Faculty of Engineering and Technology (FET), Jain (Deemed-to-be University), Bengaluru, Karnataka-562112, India

(Received 09 January 2026; Revised 13 March 2026; Accepted 23 March 2026; Published online 23 April 2026)

Abstract: The rapid growth of cyber-physical systems (CPS) and internet of things Edge-Cloud environments has amplified concerns regarding cyber threats, making intrusion detection systems (IDS) a vital component of network resilience. However, recent studies reveal a research gap in existing IDS frameworks: many fail to adapt to evolving attacks, exhibit imbalanced detection performance, or incur high computational costs. To address this challenge, this work aims to develop a robust deep learning-based classification model that accurately detects cyberattacks in CPS. Hence, this work proposed SmartCPS-ADAPT, which leverages Convolutional Neural Network-Bi-Directional Long Short-Term Memory (CNN–BiLSTM) for spatial-temporal feature extraction, an adaptive extreme gradient boosting (XGBoost) classifier for handling evolving data distributions, and a drift detection mechanism for continuous learning. Experimental evaluation on the CICIoT2023 dataset demonstrates superior performance, achieving 100% accuracy in binary classification and 99.85% accuracy in multi-class classification. The findings confirm that SmartCPS-ADAPT significantly outperforms existing approaches, ensuring reliable detection of diverse cyber-attacks. In conclusion, the SmartCPS-ADAPT establishes a highly effective IDS model that addresses adaptability, robustness, and accuracy in CPS security.

Keywords: Cyber-physical systems; CNN–BiLSTM; deep learning; drift detection; intrusion detection; IoT security; XGBoost

I. INTRODUCTION

The rapid evolution of cyber-physical systems (CPS) and proliferation of internet of things (IoT) devices have transformed modern living, enabling intelligent applications in smart homes, healthcare, industrial automation, and critical infrastructure [1]. These systems, however, are inherently exposed to a wide spectrum of cyber threats due to their reliance on interconnected networks, heterogeneous devices, and resource-constrained environments [2,3]. Traditional signature-based intrusion detection systems (IDS) [4] are insufficient to handle such large-scale, dynamic, and sophisticated attacks, motivating the adoption of artificial intelligence (AI), machine learning (ML), and deep learning (DL) methods for securing IoT-enabled CPS.

Recent years have witnessed significant progress in AI-driven intrusion detection, in which ML algorithms such as random forests (RF), support vector machines (SVM), and gradient boosting (GB) have been employed to detect attacks in network traffic [5–7]. Similarly, DL techniques, including convolutional neural networks (CNNs), recurrent neural networks, long short-term memory networks (LSTMs), and auto-encoders (AEs), have demonstrated superior capabilities for automatically learning feature representations from raw traffic flows [8–10]. Despite these advances, challenges remain in terms of scalability, robustness, and adaptability. Most existing IDS approaches fail to cope with high-dimensional IoT traffic, suffer from concept drift due to evolving attack strategies, and are often biased towards the majority classes due to imbalanced data distributions [11]. This leads to poor generalization and degraded detection accuracy in real-world CPS deployments.

To address these limitations, this work emphasizes the need to develop a robust DL-based classification framework for accurate attack detection in CPS environments, focusing solely on analyzing network packet data in the IoT–edge–cloud environment. The goal is to design an IDS that can learn complex temporal and spatial dependencies in traffic, adapt to distributional changes, and effectively distinguish between benign and malicious flows in IoT–edge–cloud ecosystems. Hence, this work proposes SmartCPS-ADAPT, a hybrid DL-GB model for attack detection in CPS. The model integrates a CNN bidirectional LSTM (BiLSTM) based feature extractor with an adaptive XGBoost (XGB) classifier (XGB-ADAPT). The CNN layers capture local spatial dependencies, while BiLSTM learns long-range temporal patterns from flow sequences. The extracted representations are passed to XGB-ADAPT, which incorporates class-sensitive reweighting, time-decayed reservoir sampling, and incremental leaf recalibration to tackle data imbalance and concept drift. Furthermore, a drift detection mechanism continuously monitors system performance and distribution shifts, ensuring adaptability to evolving threats. By leveraging both deep representation learning and adaptive boosting-based classification, SmartCPS-ADAPT provides a scalable, robust, and efficient IDS for IoT-enabled CPS, bridging the gap between existing

Corresponding author: Upasana Mahajan (e-mail: upasanamahajan1@gmail.com).

research works. The contributions of the presented work are as follows.

This work presents SmartCPS-ADAPT, a hybrid DL and adaptive boosting framework for cyber-attack detection in CPS using IoT traffic. A two-stage feature extraction module is designed using CNN and BiLSTM to capture both spatial correlations and long-term temporal dependencies in raw packet-based flows. XGB-ADAPT is introduced, integrating class-sensitive reweighting, reservoir sampling with exponential decay, and incremental leaf recalibration for handling imbalance and drift. A drift detection mechanism is implemented by monitoring loss, feature divergence, and confidence shift to adaptively update the model without costly retraining. The SmartCPS-ADAPT is evaluated on the CICIoT2023 dataset, a large-scale benchmark for IoT security, achieving robust performance across binary and multi-class classification tasks. The proposed solution demonstrates robustness and adaptability, making it suitable for deployment in IoT–edge–cloud CPS environments.

The manuscript is organized as follows. Section II presents a literature survey that discusses existing IDS approaches from recent years. Section III presents the SmartCPS-ADAPT methodology in detail. Section IV discusses the results achieved by SmartCPS-ADAPT and compares them with existing approaches discussed in the literature survey, and Section V presents the conclusion and future work of SmartCPS-ADAPT.

II. LITERATURE SURVEY

This section presents recent AI-based IDS approaches developed for detecting attacks in IoT environments. T. -T. -H. Le *et al.* [11] aimed to improve IoT security by enhancing intrusion detection through accurate classification and transparent explanations. In this work, the raw data, comprising normal and attack cases, were preprocessed and then validated using a blended approach with three ML models: GB, decision tree (DT), and RF, with a voting estimator used to evaluate the best performance. Furthermore, to better interpret the classification outcomes, local-interpretable model-agnostic explanations and counterfactual analysis were utilized. This work employed the mean-decrease impurity (MDI) approach for feature selection. For evaluation, the IoTID20 and CICIoT2023 datasets were used, with blending achieving 100% and 99.51% accuracy, respectively. Sabeel *et al.* [12] aimed at enhancing IDS for identifying polymorphic and atypical network attacks, which are often missed by existing approaches. The methodology used in this study involved generating adversarial polymorphic attacks, analyzing their quality, and retraining IDS incrementally to adapt to evolving threats. In this work, the Heterogeneous Feature Selection Ensemble approach was utilized to select feature subsets. Further, for the interpretation of results using the approach, they utilized SHapley Additive exPlanations (SHAP). The study was evaluated using CICIoT2023 and CICIDS2017, where the approach achieved nearly 90% balanced accuracy on both datasets.

A. H. Farea *et al.* [13] aimed at presenting a unified approach that addressed security in IoT environments. Hence, this work presented a replacement-encoding (RE) approach that concealed sensitive data during AI training while maintaining model utilities and providing automated preprocessing. For feature selection, 100 message packet attributes were extracted from packet-capture files of the CICIoT2023 dataset using Wireshark, and a genetic algorithm (GA) was applied to refine correlated features. For classification, we used a deep neural network (DNN) and RF. The results show that the DNN and RF with RE and GA achieved accuracies of 92.16% and 94.81%, respectively. M. Abd Elaziz *et al.* [14] aimed at enhancing IDS performance in IoT networks, for which they presented a convolutional Kolmogorov–Arnold-network (CKAN) approach, which replaced conventional multilayer perceptron in CNNs using KAN layers for reducing parameters, thereby improving performance. In this work, CNNs were used to extract features, focusing on network characteristics from standard IDS- and IoT-specific datasets. For the evaluation of CKAN, TONIoT, CICIoT2023, and NSL-KDD, the achieved results were 93.3%, 98.84%, 99.2% for multi-classification and 99.93%, 99.22%, and 98.71% for binary classification.

B. Susilo *et al.* [15] aimed at improving IoT security by improving cyber-attack detection using a DL approach; hence, they adopted a multistage approach, where they applied the synthetic minority oversampling technique (SMOTE) for addressing class imbalance, AEs for extracting features, LSTMs for capturing temporal patterns, and a CNN performed final classification. For evaluation, the CICIoT2023 dataset was used, achieving 99.15% accuracy. M. V. C. Aragão *et al.* [16] developed an efficient and scalable IoT threat-detection approach that handles large datasets with minimal computational overhead. In this work, an ML approach employing a sample-based, multistage approach was used, integrating feature selection methods such as SHAP, recursive feature elimination (RFE), and Boruta. For handling class imbalance, a hyperparameter optimization using a tree-structured Parzen estimator sampler. For classification, three approaches were used: Light gradient-boosting machine (LightGBM), XGB, and XGB with RF (XGB-RF). Results show better performance for various attack classifications on 1%, 5%, and 10% testing data.

Fares *et al.* [17] aimed to enhance IoT IDS by addressing the challenges of limited datasets and high computational demands in DL approaches; hence, they presented a hybrid transfer-learning approach that combined swin-transformers (ST) for hierarchical feature learning with an LSTM network for sequential pattern analysis. For feature selection, pretrained weights were generated and fine-tuned to improve adaptability, which helped capture temporal-structural attacks. Experiments were conducted on CICIoT2023, MQTT-IoT, BoTIoT, ToN-IoT, and NSL-KDD datasets, where better outcomes were achieved compared with AE, residual-network (ResNet), RNN, CNN, and LSTM. M.-R. Fida *et al.* [18] aimed to secure IoT systems against flow-based attacks by presenting an approach called IoTShield, which consisted of a dual-stage software-defined-network (SDN)-based defensive framework. The approach assigned programmable switches to detect specific attack classes and leveraged network controllers for refined classification and defence updates. For feature selection, an MDI approach was used, focusing on traffic attributes related to data exfiltration, scanning, spoofing, web-based attacks, and DDoS attacks. For the evaluation of IoTShield, the CICIoT2023 dataset was considered, where the approach reduced false alarms by 58% for DDoS attacks and achieved 80–99% accuracy for web attacks with DTs in the data plane and 99% accuracy using CNNs.

S. Alahmari *et al.* [19] aimed at improving IoT security against rising cyber threats by integrating data augmentation and distributed learning approaches. In this work, we combined generative-adversarial networks (GANs) and federated learning (FL) to balance datasets, using XGB as the backbone. In this work, no feature selection was considered. For the evaluation of the work, the CICIoT2023 dataset was used, where 63.60% accuracy was achieved on the original CICIoT2023 dataset and 94.62% on

CICIoT2023 GAN-generated synthetic data. Y. Zhao *et al.* [20] aimed to secure healthcare 5.0 IoT systems by addressing cyber threats and overcoming issues related to non-independent and identically distributed (Non-IID) data and device intermittency, thereby presenting the transformer-driven FL security for healthcare-industry (TFedSec-HI) approach. In this work, local binary patterns (LBP) and Sobel edge detection were employed to transform network traffic into grayscale and RGB images, enabling effective extraction using a lightweight vision-transformer on edge devices. Furthermore, model aggregation used a FedProx approach to handle data heterogeneity. Experiments were conducted on the CICIoMT2024 and CICIoT2023 datasets, achieving accuracies of 99.74% and 99.18%, respectively.

Although existing AI-based IDS approaches have shown promising results, several limitations persist. The blended ML ensemble by [21] achieved high accuracy but relied heavily on handcrafted features and static voting, making it less adaptive to evolving IoT threats. It addressed polymorphic attacks but required adversarial data generation and incremental retraining, both of which are computationally expensive [22]. The RE with GA by [23] improved privacy and feature reduction but introduced complexity in preprocessing did not address data drift. The CKAN model [24] reduced the number of parameters but still relied on conventional CNN feature learning, thereby limiting robustness to unseen patterns. Used SMOTE with AE–LSTM–CNN, but oversampling risked synthetic bias [25]. In the proposed scalable machine learning framework, the model efficiently processes large-scale IoT traffic while maintaining high detection performance and adaptability [26,27]. Enhanced SDN-based defence but was infrastructure-dependent [28]. Accuracy improved with GAN–FL, but synthetic data risk overfitting [29]. Although vision transformers demonstrate strong feature extraction capabilities, their requirement to transform data into image representations introduces additional computational overhead and latency. [30]. In contrast, SmartCPS-ADAPT overcomes these gaps by combining Convolutional Neural Network-Bi-Directional Long Short-Term Memory (CNN–BiLSTM) feature extraction with adaptive XGB, incorporating drift detection, class-sensitive weighting, and reservoir-based adaptation, ensuring scalability, temporal awareness, and robustness against evolving IoT cyber threats.

III. METHODOLOGY

This section presents the SmartCPS-ADAPT methodology. The discussion is structured as follows: first, the overall SmartCPS-ADAPT architecture is described, followed by details of the dataset employed in this study. Next, the preprocessing steps applied to the raw data are outlined. Subsequently, the feature extraction process is explained, highlighting the use of a CNN–BiLSTM model to learn spatial and temporal patterns. The classification strategy using the adaptive XGB model is then presented. Finally, the drift detection mechanism integrated into the framework is elaborated to demonstrate its ability to maintain robustness under evolving IoT attack scenarios.

A. ARCHITECTURE

The SmartCPS-ADAPT architecture is shown in Fig. 1 and comprises an end-to-end IDS framework for IoT environments. The SmartCPS-ADAPT first considers the CICIoT2023 dataset, which comprises raw IoT traffic with eight classes (seven attack classes and one benign class), collected from 105 IoT devices. The

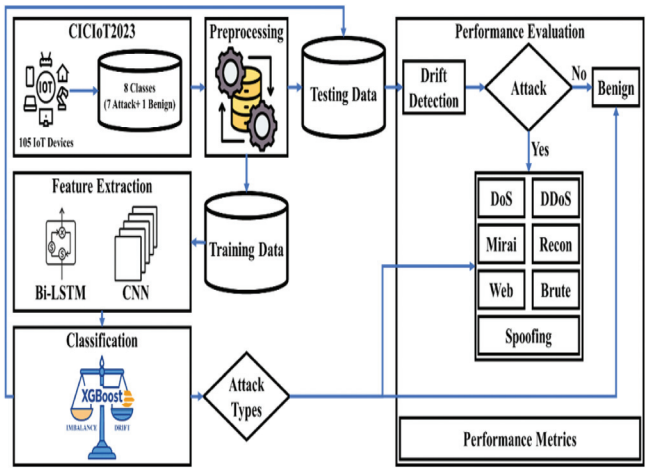

Fig. 1. SmartCPS-ADAPT architecture.

complete details of the dataset are discussed in Section III.B. Next, the dataset is preprocessed, which is discussed in detail in Section III.C. Further, the dataset is partitioned into training and testing sets. The training data are then passed to the feature extraction module, where a hybrid CNN and Bi-LSTM model learns both spatial correlations and long-term temporal dependencies in IoT flows, as discussed in detail in Section III.D. Further, the extracted representations are fed into a classification module called as XGB-ADAPT, which is discussed in detail in Section III.E. Further, for drift detection, the SmartCPS-ADAPT presents a novel algorithm, as discussed in Section III.F. Further, using the test data, the performance of SmartCPS-ADAPT is evaluated for binary and multi-class classification using standard performance metrics.

B. DATASET

In this study, for training/testing the SmartCPS-ADAPT, this work considered the CICIoT2023 dataset [21]. The CICIoT2023 dataset is a large-scale, benchmarked dataset designed to support research in IoT security, addressing the growing need for realistic IDS and threat analysis in modern IoT environments. The dataset was developed by the Canadian Institute for Cybersecurity (CIC), which provides a comprehensive collection of malicious and benign network traffic, simulating real-world IoT environments such as sensor-based smart homes and industrial control systems applications. The dataset was generated using 105 IoT devices and attack scenarios, which captured packet-based and flow-based features. The dataset comprises DoS, DDoS, brute-force, Mirai, spoofing, web-based, and reconnaissance attacks. The details of the complete dataset are presented in Table I.

C. PREPROCESSING

In this study, the CICIoT2023 dataset, provided in 169 separate CSV files, was first consolidated into a single file for simplifying processing and model training. The next step considered in preprocessing involved converting text-based labels into numerical representations to make them compatible with SmartCPS-ADAPT. For binary classification, benign traffic was labeled as 0 and malicious traffic as 1; for multi-class classification, malicious flows were grouped into seven distinct classes (33 classes, including benign traffic), resulting

Table I. Ciciot2023 dataset

Data description	Value
Samples	46,686,579
Features	46
Classes	Benign, DoS, DDoS, Mirai, Spoofing, Recon, Web, BruteForce
Benign samples	1,098,195
DoS samples	8,090,738
DDoS samples	33,984,560
Mirai	2,634,124
Spoofing	486,504
Recon	354,565
Web	24,829
BruteForce	13,064

in eight labels in total. To ensure consistent feature scaling and improve SmartCPS-ADAPT performance, feature normalization was applied using StandardScaler, which transformed values to a standard normal distribution with a mean of 0 and a standard deviation of 1. As the predefined training and testing sets were not available in the dataset, the holdout approach was used to partition the dataset, i.e., 70% of the dataset was allocated for training and 30% for testing. To avoid potential data leakage, the dataset was shuffled before splitting, and stratified sampling was applied to preserve class distributions across training and testing sets. Additionally, duplicate samples were removed during preprocessing to ensure that identical flows did not appear simultaneously in both training and testing partitions. This strategy helps ensure a fair evaluation of the SmartCPS-ADAPT framework.

D. FEATURE EXTRACTION

For feature extraction, this work presents a two-stage 1-dimensional (1D) convolutional-recurrent feature extractor that converts raw tabular flow samples into a compact, temporally aware representation suitable for both binary and multi-class classification. Consider the CICIoT2023 dataset, which has N samples with d features. This work first constructs short-ordered sequences from flows per device/session using a sliding window of length T, so the SmartC, PS-ADAPTn gets sequences $X^{(i)} \in \mathbb{R}^{T\times d}$, where i denotes index sequence windows. The convolutional encoder applies 1D-convolution ross mpol as for learning local temporal patens.e., for coolutional layer l having kernel $K^{(l)} \in \mathbb{R}^{k_l\times d_l\times c_l}$, where k_l denotes kernel length, d_l denotes input channels, and c_l denotes output channels. Using this the pre-activation denoted as $Z^{(l)}$ and activation denoted as $H^{(l)}$ are evaluated using Eq. (1) and Eq. (2), respectively.

$$Z^{(l)} = K^{(l)} * H^{(l-1)} + b^{(l)} \tag{1}$$

$$H^{(l)} = \phi(Z^{(l)}) \tag{2}$$

In Eq. (1), $*$ denotes 1-D convolution, $b^{(l)}$ denotes a bias vector, and in Eq. (2), ϕ denotes elementwise non-linearity, i.e., rectified linear unit (ReLU). In this work, the convolutional blocks L have been stacked with batch normalization denoted as BN and residual skip connections to stabilize training in residual blocks. This function is mathematically represented using Eq. (3).

$$H^{(l)} = \phi(BN(K^{(l)} * H^{(l-1)} + b^{(l)})) + H^{(l-1)} \tag{3}$$

In this work, temporal max-pooling has been applied as it reduces the sequence length to T' and produces convolutional feature maps $C \in \mathbb{R}^{T'\times m}$, where m denotes final channel count. These convolutional features are then passed to an LSTM for capturing long-range sequential dependencies. In LSTM, for each time step t, it computes gates and states using Eqs. (4)–(9).

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \tag{4}$$

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f) \tag{5}$$

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \tag{6}$$

$$\tilde{c}_t = \tanh(W_c x_t + U_c h_{t-1} + b_c) \tag{7}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t \tag{8}$$

$$h_t = o_t \tanh(c_t) \tag{9}$$

In Eq. (4) to Eq. (9), $x_t \in \mathbb{R}^m$ denotes a convolution feature at t, h_t, and c_t denote hidden and cell states, matrices W_*, U_* and biases b_* are learnable parameters, σ denotes sigmoid, and $\odot$ denotes elementwise product. This work utilizes a Bi-LSTM, so the final sequence representation is $H_{seq} = \left[\overrightarrow{h_T}; \overrightarrow{h_1}\right] \in \mathbb{R}^{2r}$, where r denotes hidden size. For obtaining a compact feature vector $z \in \mathbb{R}^q$. This work applies multi-head self-attention to LSTM outputs, achieving attention scores using Eq. (10) and a pooled vector using Eq. (11).

$$\alpha_j = softmax(w_a^{\mathrm{T}} \tanh(W_a H_{all} + b_a)) \tag{10}$$

$$z = \sum_j \alpha_j h_j \tag{11}$$

In Eq. (10), H_{all} stacks all LSTM hidden states and w_a, W_a and b_a are learnable. Finally, a projection layer $z' = ReLU(W_p z + b_p)$ yields an extracted feature vector used by downstream classifiers. All intermediate outputs are regularized by dropout and $L2$ penalties; loss for end-to-end pretraining uses cross-entropy. Using this feature extraction approach, the SmartCPS-ADAPT extracts spatial (feature-wise) and temporal patterns, reduces noise through pooling/attention, and produces a compact embedding from 46 raw features across various temporal windows for both binary and multi-class classification.

E IMPLEMENTATION DETAILS

The CNN-BiLSTM network consists of three convolutional layers with filter sizes of 64, 128, and 256, respectively, each using a kernel size of 3. Batch normalization and dropout with a rate of 0.3 were applied after each convolution layer to prevent overfitting. The BiLSTM layer contains 128 hidden units and processes sequential traffic windows extracted from the input data. The network was trained using the Adam optimizer with a learning rate of 0.001 and a batch size of 128. For the XGB-ADAPT classifier, the maximum tree depth was set to 8, the learning rate to 0.1, and the number of estimators to 200.

F. CLASSIFICATION

In this work, a hybrid XGB variant, called XGB-ADAPT, is presented, which combines cost-sensitive gradient boosting, sample reweighting, and incremental leaf recalibration to handle drift and class imbalance. Consider $\mathcal{D} = \{(x_i, y_i, w_i)\}$. In the standard XGB [22], the objective for iteration t is evaluated using Eq. (12). In Eq. (12), y_i denotes actual label and y_i denotes predicted label, t denotes time-step f_t denotes new tree, x_i denotes input, l denotes loss (this work has used logistic/focal hybrid), Ω denotes tree regularization. This work modifies the instance loss with class-sensitive scaling, as presented in Eq. (13).

$$\mathcal{L}^{(t)} = \sum_i l(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)) + \Omega(f_t) \tag{12}$$

$$l_{cs}(y, \hat{y}) = \alpha_y l(y, \hat{y}) + \beta FL_\gamma(y, \hat{y}) \tag{13}$$

In Eq. (13), α_y denotes class-weight inversely proportional to smoothed class frequency, FL_γ denotes focal loss with the focusing parameter γ, and β balances these two components. From this, the gradients and Hessians in the XGB function are updated as presented in Eqs. (14) and (15). To address drift, this work maintains a time-decayed reservoir of recent samples. $\mathcal{R}$. With exponential ageing, i.e., when a new sample (x_n, y_n) arrives, its weight is initialized $w_n = 1$ and existing reservoir weights are multiplied by decay $\lambda \in (0,1)$. By training on mini-batches sampled proportionally from the time- decayed reservoir, the model prioritises recent data distributions, thereby improving adaptability to concept drift. Further, during tree construction, the gain of splitting a node is computed as a weighted sum, i.e., for a candidate split. s, the gain is evaluated as presented in Eq. (16).

$$g_i = \partial_y l_{cs}(y_i, \hat{y}_i) \tag{14}$$

$$h_i = \partial_y^2 l_{cs}(y_i, \hat{y}_i) \tag{15}$$

$$Gain = \frac{1}{2}\left(\frac{G_L^2}{H_L + \eta} + \frac{G_R^2}{H_R + \eta} + \frac{G_{tot}^2}{H_{tot} + \eta}\right) - \gamma T \tag{16}$$

In Eq. (16), G_* and H_* are weighted gradients and Hessians aggregated with weights? w_i, η denotes leaf regularizer and γT denote complexity penalty. After each epoch, XGB-ADAPT performs leaf re-calibration, i.e., given leaf predictions. p_j and the recent validation window V This work solves the regularized least squares update as presented in Eq. (17). Eq. (17) adapts leaf outputs without rebuilding trees. This enables fast adaptation to drift. The combination of class-sensitive loss, decayed reservoir sampling, and leaf re-calibration provides a GB approach that is both robust to shifting distributions and attentive to minority classes while keeping training efficient.

$$\min_{\Delta} \sum_{(x,y)\in V} w(x)(y - (p_{leaf(x)} + \Delta_{leaf(x)}))^2 + \rho||\Delta||^2 \tag{17}$$

G. DRIFT DETECTION

This section presents the drift detection algorithm used in XGB-ADAPT, where the main goal is to detect distribution or performance drift and adapt the classifier with minimal retraining. The algorithm for the drift detection is given below. This algorithm balances sensitivity (detecting real shifts) and stability (avoiding false positives) by combining SmartCPS-ADAPT performance signals with data-distribution checks and adapting through a lightweight, reservoir-based update.

Algorithm 1 continuously monitors IoT data streams to detect and adapt to concept drift. The algorithm processes incoming mini-batches B_t of size b, computes average loss L_t, and updates the exponentially weighted moving average $\mu_{L,t}$ for stability. For each batch, feature distribution p_t^j and confidence scores C_t are analyzed, with drift quantified using three metrics, loss drift S_L, feature divergence S_{KL}, and confidence shift S_C. These are fused into a composite score. S using weights σ_L, σ_{KL}, σ_C. If the S exceeds warning threshold τ_{warn}, sampling is intensified; if it exceeds the drift threshold τ_{drift} for k consecutive batches, adaptation is triggered. The SmartCPS-ADAPT freezes, a reservoir. R of size R is populated with recent data, reweighted using decay λ and class weights α_y SmartCPS-ADAPT is incrementally fine-tuned with recalibration. The results achieved by the SmartCPS-ADAPT are discussed in detail in the next section.

IV. RESULTS AND DISCUSSION

A. PERFORMANCE EVALUATION METRICS

To evaluate the performance of SmartCPS-ADAPT, standard metrics such as accuracy, precision, recall, and F1-score were adopted, as defined in Eq. (18)–Eq (21), where, TP represents true positives, TN denotes true negatives, FP refers to false positives, and FN indicates false negatives.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{18}$$

$$i.i.Precision = \frac{TP}{TP + FP} \tag{19}$$

$$Recall = \frac{TP}{TP + FN} \tag{20}$$

$$F - Score = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} \tag{21}$$

B. CROSS-VALIDATION AND STATISTICAL ANALYSIS

To further validate the robustness and generalization capability of the proposed SmartCPS-ADAPT framework, a fivefold cross-validation experiment was conducted on the CICIoT2023 dataset. The dataset was randomly shuffled and partitioned into five equal folds, preserving the class distribution via stratified sampling. In each iteration, four folds were used for training, and the remaining fold was used for testing. The average results across the five folds demonstrate consistent performance, confirming the stability of the proposed model. The SmartCPS-ADAPT achieved an average accuracy of 99.82% with a standard deviation of 0.03, indicating that the model maintains reliable detection capability across different data partitions.

C. BINARY CLASSIFICATION

The experimental results, as presented in Fig. 2, demonstrate SmartCPS-ADAPT's performance on the CICIoT2023 dataset for binary-class classification. In a binary-class classification task, SmartCPS-ADAPT achieves 100% accuracy, 99.9% precision, 99.95% recall, and an F-score of 99.92%. The performance is

Algorithm 1. XGB-ADAPT Drift Detection.

Input Stream of batches B_t, each batch size b, current SmartCPS-ADAPT model $\mathcal{M}$, reference windows W_s (stable) and W_r (recent), $\rho \in (0,1)$, feature importance weights $\{w_j\}$, fusion weights $\sigma_L, \sigma_{KL}, \sigma_C$, warning threshold τ_{warn}, drift threshold τ_{drift} $(\tau_{warn} < \tau_{warn})$, reservoir capacity R (number of batches to collect), decay factor λ for reservoir weighting, class weights $\{\alpha_y\}$ and persistence requirement k (consecutive batches above threshold to trigger)

Output Updated SmartCPS-ADAPT model $\mathcal{M}$ adapted to the new distribution and reservoir $\mathcal{R}$ containing prioritised recent samples.

Step 1 Start

Step 2 Set loss $\mu_{L,0}$, i.e., initial validation loss

Step 3 Set counters $warn_{count} = 0$ and $drift_{count} = 0$

Step 4 Initialise empty reservoir $\mathcal{R} =$

Step 5 For each incoming batch $\boldsymbol{B_t}$

Compute batch loss using $L_t = \frac{1}{b}\sum_{(x,y)\in B_t} \ell(y,\hat{y})$

For each feature, $\boldsymbol{j}$ compute marginal histogram $\boldsymbol{p_t^j}$

Compute prediction confidence distribution at C_t

Update loss using $\mu_{L,t} = \rho\mu_{L,t-1} + (1-\rho)L_t$

Compute drift scores

Loss drift $S_L = \mu_{L,t} - \mu_{L,ref}$

.Feature Kullback-Leibler aggregate $S_{KL} = \sum_j w_j KL(p_t^j || p_{ref}^j)$

confidence shift $S_C = ||CDF(C_t) - CDF(C_{ref})||_1$

Normalize each score using $S = \sigma_L S_L + \sigma_{KL} S_{KL} + \sigma_C S_C$

If $\boldsymbol{S > \tau_{warn}}$

$warn_{count} += 1$, increase reservoir sampling rate

Else $\boldsymbol{warn_{count} = o}$

If $\boldsymbol{S > \tau_{drift}}$

$drift_{count} += 1$

Else

$drift_{count} = 0$

End If

End If

Trigger adaptation only if $drift_{count} \geq k$

Freeze $\mathcal{M}$

Collect next R batches into the reservoir $\mathcal{R}$. When adding a new batch, apply exponential ageing to existing reservoir sample weights by multiplying the weight by

Reweight samples in $\mathcal{R}$ with class weights α_y and time decay

Incrementally retrain/fine-tune $\mathcal{M}$ on $\mathcal{R}$. Apply leaf re-calibration for fast correction.

Update reference winw W_s, reset and reseti.$rift_{count} = 0$, $warn_{ount} = 0$

Periodically, compute reference statistics $\mu_{L,ref}$, p_{ref}^j and C_{ref} from updated W_s

End for

Step 6 End

attributed to the integration of CNN-LSTM for hierarchical and temporal feature extraction, which enables effective learning of both spatial correlations in network traffic and sequential dependencies in attack patterns. Additionally, the adaptive XGB classifier enhances robustness by efficiently handling large-scale data while mitigating drift and class imbalance, ensuring consistent generalization to unseen samples.

D. MULTI-CLASS CLASSIFICATION

The experimental results, as shown in Fig. 3, demonstrate SmartCPS-ADAPT's performance on the CICIoT2023 multi-class classification dataset. In the more challenging multi-class classification setting, which involves differentiating between benign traffic and seven distinct attack categories, SmartCPS-ADAPT achieves 99.85% accuracy, 99.7% precision, 99.8% recall, and 99.75% F-score. The results show that SmartCPS-ADAPT provides a robust and generalizable DL-based IDS framework, offering significant improvements by effectively capturing complex attack behaviors, addressing data drift, and maintaining reliable performance across both binary and multi-class tasks. The multi-class classification performance of the proposed framework is illustrated in Fig. 4, demonstrating its effectiveness in distinguishing among multiple attack categories with high precision and recall.

E. ABLATION STUDY

To evaluate the contribution of individual components of the SmartCPS-ADAPT framework, an ablation study was conducted.

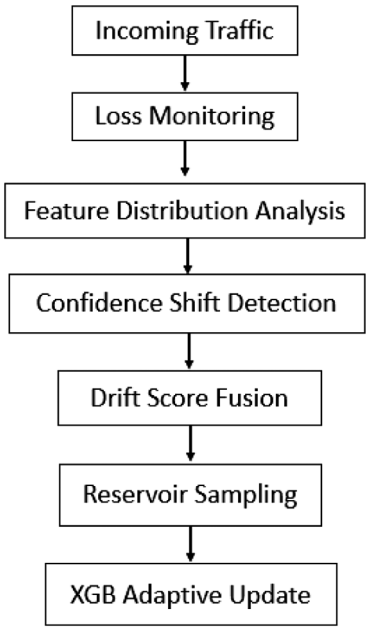

Fig. 2. Workflow of drift detection and adaptive updating in SmartCPS-ADAPT.

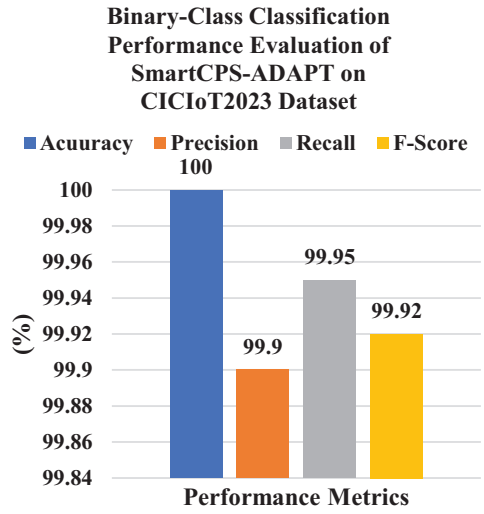

Fig. 3. SmartCPS-ADAPT performance for binary classification on the CICIoT2023 dataset.

Different variants of the architecture were evaluated by progressively adding model components.

The results confirm that each component contributes to improved detection performance, with the full SmartCPS-ADAPT framework achieving the highest accuracy.

F. DRIFT ADAPTATION EVALUATION

To evaluate the effectiveness of the drift detection mechanism, a controlled distribution shift experiment was conducted. The model was initially trained on the original dataset distribution, and the testing data were modified to simulate evolving attack patterns. The performance of the baseline model without drift adaptation was compared with the SmartCPS-ADAPT model.

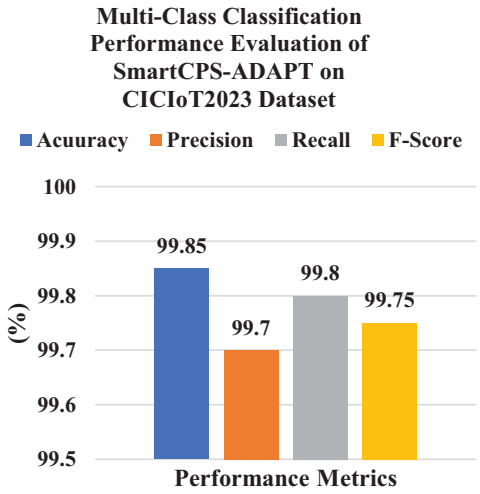

Fig. 4. SmartCPS-ADAPT performance for binary classification on the CICIoT2023 dataset.

The results demonstrate that the proposed drift detection and adaptation mechanism effectively maintains detection performance under changing data distributions.

G. COMPARATIVE STUDY

The comparative evaluation on the CICIoT2023 dataset highlights the effectiveness of SmartCPS-ADAPT over existing IDS in both binary-class and multi-class classification tasks. In the binary classification setting, as shown in Table II, SmartCPS-ADAPT achieves 100% accuracy, 99.9% precision, 99.95% recall, and an F-score of 99.92%, surpassing existing approaches such as CKAN [14] and XGB [16]. While ensemble models such as LightGBM and XGB-RF also achieve perfect accuracy, their precision and recall remain significantly lower, with XGB-RF particularly struggling on recall (92.3%). This demonstrates that SmartCPS-ADAPT not only reaches perfect accuracy but also maintains better performance, which is essential to minimize both FPs and FNs.

In the multi-class classification task, as presented in Table III, SmartCPS-ADAPT again outperforms other advanced models. It achieves 99.85% accuracy, 99.7% precision, 99.8% recall, and an F-score of 99.75%, outperforming existing approaches such as Blending [11] and SMOTE + AE + LSTM + CNN [15]. While blending achieved strong results (99.51% accuracy, 99.07% F-score), SmartCPS-ADAPT consistently improved across all metrics, especially in precision and recall, reflecting its ability to correctly distinguish among multiple complex attack classes, including DoS, DDoS, Mirai, Spoofing, and Web-based attacks. Other methods, such as RE + GA + DNN [13] and XGB [19], fall

Table II. Performance of SmartCPS-ADAPT using 5-fold cross-validation on the CICIoT2023 dataset

Fold	Accuracy	Precision	Recall	F1
1	99.81	99.72	99.79	99.75
2	99.84	99.70	99.83	99.76
3	99.83	99.69	99.81	99.74
4	99.80	99.71	99.78	99.74
5	99.82	99.70	99.80	99.75

Table III. Ablation study evaluating the contribution of individual components in the SmartCPS-ADAPT framework

Model	Accuracy
CNN	97.6
CNN + LSTM	98.8
CNN + BiLSTM	99.1
CNN + BiLSTM + XGB	99.63
SmartCPS-ADAPT	99.85

Table IV. Performance comparison under simulated concept drift conditions

Model	Accuracy after drift
Baseline CNN-BiLSTM	92.4
SmartCPS-ADAPT	98.7

Table V. Comparative study On Ciciot2023 2 class binary-class classification

Ref	Model	Accuracy	Precision	Recall	F-Score
[14]	CKAN	99.22	99.81	99.4	99.6
[16]	LightGBM	100	95.1	98.5	96.6
	XGB	100	98.7	99.8	97.4
	XGB-RF	100	98.7	92.3	95.2
[17]	ST + LSTM	98.78	98.54	98.77	98.56
	SmartCPS-ADAPT	100	99.9	99.95	99.92

significantly short, particularly in precision and recall, which limits their reliability in real-world CPS and IoT scenarios. The performance comparison under simulated concept drift conditions is presented in Table IV, highlighting the robustness of the proposed adaptive mechanism. Table V presents a comparative analysis of binary classification performance against state-of-the-art methods. Table VI summarises the multi-class classification performance, demonstrating the superiority of the proposed framework across all evaluation metrics's all evaluation metrics.

Overall, SmartCPS-ADAPT establishes itself as a robust, scalable, and highly accurate IDS framework capable of outperforming traditional ML, ensemble, and hybrid DL approaches by effectively extracting discriminative features, learning temporal patterns, and adapting to evolving threats.

Although the proposed model achieved extremely high detection accuracy, several techniques were employed to mitigate potential overfitting. These include dropout regularization, batch normalization, and early stopping during training. Furthermore, cross-validation experiments confirm that the model maintains stable performance across multiple data partitions.

V. CONCLUSION

This work addressed the growing security challenges in CPS and IoT–edge–cloud environments, where the increasing scale and complexity of networks expose systems to sophisticated cyberattacks. The study was motivated by the research gap in existing IDS, where many models either fail to generalize across evolving threats, suffer from imbalanced performance between accuracy and recall, or demand high computational costs. The problem addressed in this work is the need to develop a robust DL-based classification framework that accurately detects cyberattacks in CPS using packet-level analysis. The primary objective of this work was to design an adaptable IDS, integrate effective preprocessing, feature extraction, and classification techniques, and incorporate drift detection to maintain long-term robustness. Hence, the proposed SmartCPS-ADAPT methodology successfully fulfilled these objectives. It achieved 100% accuracy, 99.9% precision, 99.95% recall, and 99.92% F-score for binary-class classification and 99.85% accuracy, 99.7% precision, 99.8% recall, and 99.75% F-score for multi-class classification on the CICIoT2023 dataset. These results demonstrate significant improvements compared to state-of-the-art methods, ensuring highly reliable detection of both common and advanced attack types. For future work, SmartCPS-ADAPT can be extended to develop resilient CPS architectures using ML/DL models that ensure continuous operation and adaptability, even under adverse conditions such as large-scale distributed or adaptive adversarial attacks.

Table VI. Comparative study On Ciciot2023 8 class multi-class classification

Ref	Model	Accuracy	Precision	Recall	F-Score
[11]	ET	97.07	95.07	94.08	94.57
	RF	95.36	95.36	97.31	96.33
	GB	96.4	94.4	95.39	94.89
	Blending	99.51	98.51	99.63	99.07
[13]	RE + GA + DNN	92.16	58.82	66.67	62.5
	RE + GA + RF	94.81	65.84	77.25	64.96
[15]	SMOTE + AE + STM + CNN	99.15	99.39	99	99.19
[16]	LightGBM	99.1	91	89.7	76.8
	XGB	99.7	98.7	88	90.1
	XGB-RF	99.6	95.6	84.5	88.6
[17]	ST + LSTM	97.44	97.67	97.37	97.76
[19]	XGB	63.6	63.08	63.23	62.91
	SmartCPS-ADAPT	99.85	99.7	99.8	99.75

CONFLICT OF INTEREST STATEMENT

The author(s) declare that they have no conflicts of interest to report regarding the present study.

ACKNOWLEDGMENT

I would like to express our sincere gratitude to all those who have supported and contributed to this research project. Primarily, I extend our heartfelt thanks to our guide for his unwavering guidance, invaluable insights, and encouragement throughout the research process.

REFERENCES

[1] S. Rani *et al.*, "A new generation cyber-physical system: A comprehensive review from a security perspective," *Comput. Secur.*, vol. 148, p. 104095, Jan. 2025, DOI: 10.1016/j.cose.2024.104095.

[2] S. Lee *et al.*, "Cyber-physical artificial intelligence," *ACM Trans. Cyber-Phys. Syst.*, Mar. 2025, DOI: 10.1145/3721437.

[3] S. Suhail *et al.*, "A framework for enhancing cyber incident response with security-enhancing digital twins in cyber-physical systems," *Internet of Things*, p. 101547, Feb. 2025, DOI: 10.1016/j.iot.2025.101547.

[4] U. Ahmed *et al.*, "Signature-based intrusion detection using machine learning and deep learning approaches empowered with fuzzy clustering," *Sci. Rep.*, vol. 15, no. 1, Jan. 2025, DOI: 10.1038/s41598-025-85866-7.

[5] W. Serrano, "CyberAIBot: Artificial intelligence in an intrusion detection system for CyberSecurity in the IoT," *Future Gener. Comput. Syst.*, p. 107543, Oct. 2024, DOI: 10.1016/j.future.2024.107543.

[6] M. Rahman, S. A. Shakil, and M. R. Mustakim, "A survey on intrusion detection system in IoT networks," *Cyber. Secur. Appl.*, p. 100082, Dec. 2024, DOI: 10.1016/j.csa.2024.100082.

[7] A. Alsirhani *et al.*, "Intrusion detection in smart grids using artificial intelligence-based ensemble modelling," *Cluster Comput.*, vol. 28, no. 4, Feb. 2025, DOI: 10.1007/s10586-024-04964-9.

[8] Y. Sun and Z. Wang, "Intrusion detection in IoT and wireless networks using image-based neural network classification," *Appl. Soft Comput.*, vol. 177, pp. 113236–113236, May 2025, DOI: 10.1016/j.asoc.2025.113236.

[9] R. Manivannan and S. Senthilkumar, "Intrusion detection system for network security using novel adaptive recurrent neural network-based fox optimiser concept," *Int. J. Comput. Intelligence Syst.*, vol. 18, no. 1, Feb. 2025, DOI: 10.1007/s44196-025-00767-x.

[10] F. S. Alsubaei, "Smart deep learning model for enhanced IoT intrusion detection," *Sci. Rep.*, vol. 15, no. 1, Jul. 2025, DOI: 10.1038/s41598-025-06363-5.

[11] T.-T.-H. Le *et al.*, "Toward enhanced attack detection and explanation in intrusion detection system-based IoT environment data," *IEEE Access*, vol. 11, pp. 131661–131676, Nov. 2023, DOI: 10.1109/ACCESS.2023.3336678.

[12] U. Sabeel *et al.*, "Incremental adversarial learning for polymorphic attack detection,"*IEEE Trans. Mach. Learn. Commun. Netw*, vol. 2, pp. 869–887, June. 2024, DOI: 10.1109/TMLCN.2024.3418756.

[13] A. H. Farea *et al.*, "AI-powered integrated with encoding mechanism enhancing privacy, security, and performance for IoT ecosystem," *IEEE Access*, vol. 12, pp. 121368–121386, Aug. 2024, DOI: 10.1109/ACCESS.2024.3449630.

[14] M. Abd Elaziz, I. Ahmed Fares, and A. O. Aseeri, "CKAN: Convolutional Kolmogorov–Arnold networks model for intrusion detection in IoT environment," *IEEE Access*, vol. 12, pp. 134837–134851, Sept. 2024, DOI: 10.1109/ACCESS.2024.3462297.

[15] B. Susilo, A. Muis, and R. F. Sari, "Intelligent intrusion detection system against various attacks based on a hybrid Deep Learning Algorithm," *Sensors*, vol. 25, no. 2, pp. 580–580, Jan. 2025, DOI: 10.3390/s25020580.

[16] M. V. C. Aragão *et al.*, "A sample-based, multi-stage machine learning pipeline for scalable IoT threat detection," in *IEEE Embed. Syst. Lett.*, May. 2025, DOI: 10.1109/LES.2025.3567025.

[17] I. A. Fares *et al.*, "Deep transfer learning based on hybrid Swin Transformers With LSTM for intrusion detection systems in IoT environment," *IEEE Open J. Commun. Soc.*, vol. 6, pp. 4342–4365, May. 2025, DOI: 10.1109/OJCOMS.2025.3569301.

[18] M.-R. Fida, A. H. Ahmed, and A. S. Arsalaan, "IoTShield: Defending IoT systems against prevalent attacks using programmable networks," *IEEE Access*, vol. 13, pp. 136446–136457, July. 2025, DOI: 10.1109/ACCESS.2025.3594580.

[19] S. Alahmari and Noura Aleisa, "Enhancing the CIC IoT dataset 2023 for improved attack detection through GANs augmentation and federated learning," *J. Comput. Sci.*, vol. 21, no. 7, pp. 1688–1704, Jul. 2025, DOI: 10.3844/jcssp.2025.1688.1704.

[20] Y. Zhao *et al.*, "TFedSec-HI: Transformer-driven federated security for IoT-enabled healthcare industry 5.0 on Non-IID data," *IEEE Internet of Things J.*, Sept. 2025, DOI: 10.1109/JIOT.2025.3605756.

[21] "IoT Dataset 2023 | Datasets | Research | Canadian Institute for Cybersecurity | UNB, www.unb.ca.https://www.unb.ca/cic/datasets/iotdataset-2023.html

[22] T. Chen and C. Guestrin, "XGBoost: a Scalable Tree Boosting System," *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '16*, vol. 1, no. 1, pp. 785–794, Aug. 2016, DOI: 10.1145/2939672.2939785.

[23] I. Dutt, S. Borah and I. K. Maitra, "Immune system based intrusion detection system (IS-IDS): A proposed model," *IEEE Access*, vol. 8, pp. 34929–34941, 2020, DOI: 10.1109/ACCESS.2020.2973608.

[24] C. Blum, J. A. Lozano, and P. P. Davidson, "An artificial bioindicator system for network intrusion detection," *Artif. Life*, vol. 21, no. 2, pp. 93–118, May 2015, DOI: 10.1162/ARTL_a_00162.

[25] H. Im and S. Lee, "TinyML-based intrusion detection system for in-vehicle network using convolutional neural network on embedded devices," *IEEE Embedd. Syst. Lett.*, vol. 17, no. 2, pp. 67–70, April 2025, DOI: 10.1109/LES.2024.3475470.

[26] J.-H. Cho, I.-R. Chen, and P.-G. Feng, "Effect of intrusion detection on reliability of mission-oriented mobile group systems in mobile Ad Hoc networks," *IEEE Trans. Reliab.*, vol. 59, no. 1, pp. 231–241, March 2010, DOI: 10.1109/TR.2010.2040534.

[27] W. Hu, W. Hu, and S. Maybank, "AdaBoost-based algorithm for network intrusion detection," *IEEE Trans. Syst. Man Cybern.B (Cybern)*, vol. 38, no. 2, pp. 577–583, April 2008, DOI: 10.1109/TSMCB.2007.914695.

[28] B. J. Bejoy and S. Janakiraman, "Enhanced AIS-based intrusion detection system using Natural Killer Cells," *J. Cyber Secur. Mobility*, vol. 9, no. 4, pp. 515–534, October 2020, DOI: 10.13052/jcsm2245-1439.942.

[29] J. Zuniga-Mejia *et al.*, "A linear systems perspective on intrusion detection for routing in reconfigurable wireless networks," *IEEE Access*, vol. 7, pp. 60486–60500, 2019, DOI: 10.1109/ACCESS.2019.2915936.

[30] R. H. Altaie and H. K. Hoomand, "An intrusion detection system using a hybrid lightweight deep learning algorithm", *Eng. Technol. Appl. Sci. Res.*, vol. 14, no. 5, pp. 16740–16743, Oct. 2024, DOI: 10.48084/etasr.7657.