IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 3 October 2025, accepted 3 November 2025, date of publication 10 November 2025,
date of current version 14 November 2025.

Digital Object Identifier 10.1109/ACCESS.2025.3630753

RESEARCH ARTICLE

A Hybrid Lightweight Deep Learning-Based Intrusion Detection Approach in IoT Utilizing Feature Selection & Explainable Artificial Intelligence

AHWAR KHAN[1], MD ASDAQUE HUSSAIN[2], AND FAISAL ANWER[1]
[1]Department of Computer Science, Aligarh Muslim University, Aligarh, Uttar Pradesh 202002, India
[2]Faculty of Computer Studies, Arab Open University, Manama 26196, Bahrain

Corresponding author: Ahwar Khan (khanahwar4@gmail.com)

ABSTRACT Due to the resource constraints of IoT devices, standard cryptographic-based intrusion detection techniques are ineffective in real-world IoT environments. This study presents DL-IID, a lightweight deep learning-based framework for IoT intrusion detection. The model integrates deep neural networks (DNN) with bidirectional long short-term memory (Bi-LSTM) to detect complicated temporal and nonlinear attack patterns. A wrapper-based genetic algorithm (GA) is employed for feature selection, eliminating redundant features and further reducing memory consumption. In addition, dynamic quantization after training reduces the model size to 108.42 KB while retaining high accuracy, which is within the reach of limited IoT nodes. A key aspect of this study is the application of XAI methodology, specifically Local Interpretable Model-Agnostic Explanations (LIME), which ensures transparency in decision-making and enhances trust in the model predictions. The proposed method is compared with four standard datasets: RF fingerprinting (450 IoT devices), CICIDS2017, CICIoMT2024, and UNSW-NB15. Experimental results demonstrate improved performance with 99.84% accuracy, 100% precision, 99.69% recall, and 99.84% F1-score, while significantly reducing model size and computational overhead. Experimental results confirm that DL-IID is a viable and practical solution for intrusion detection in next-generation IoT environments.

INDEX TERMS Deep learning, dynamic quantization, explainable artificial intelligence, genetic algorithm, intrusion detection, Internet of Things.

I. INTRODUCTION

The Internet of Things (IoT) is an interactive paradigm that links everyday objects by allowing easy access and interaction with several devices. The integration of new network technologies with the IoT has considerably expanded the danger landscape for NGN-IoT networks and applications [1]. We must recognize that attackers can easily compromise IoT devices because of their limited resources and inability to execute advanced computational authentication procedures at the endpoint. Conventional security solutions rely on bit-level identifiable information, such as Message Authentication Code (MAC) addresses and network cryptographic keys [2], [3]. However, it is possible to duplicate or alter MAC addresses and generate cryptographic keys [4], [30]. Once attackers have traditional credentials, they can deploy malicious devices in a new-generation network that is already in operation and use them to launch a variety of attacks or create botnets [5].

In authentication-based systems, the access control paradigm grants users network access. Conventional IoT authentication solutions primarily emphasize credentials and cryptographic mechanisms contingent upon established credentials, including usernames and passwords. Conversely, cryptographic techniques facilitate encrypted communication channels through the use of hashing algorithms.

The associate editor coordinating the review of this manuscript and approving it for publication was Tai-hoon Kim.

Both categories of authentication methods encounter significant dangers [7], [8]. Due to resource limitations, IoT devices cannot execute power-intensive encryption algorithms on end nodes. Adversaries can readily incapacitate networks, infiltrate IoT devices, and repurpose them as botnets. In addition, in certain instances, attackers can impersonate users entirely despite traditional authentication methods [9], [10], especially when they have stolen credentials or when authentication relies on certificates. NGN networks possess distinct characteristics that do not entirely accommodate conventional methods for achieving optimal performance in the network. For example, 5G enables real-time applications, ensuring that the time required to authenticate and verify legitimate nodes remains below a certain threshold; otherwise, a much longer delay would occur in determining which nodes are legitimate [6], [11]. Current security methods that rely on traffic data analysis are equally inadequate and insecure for NGNs [12]. Authentication relying on traffic data analysis is inadequate for preventing impersonation attacks. Consequently, authentication of next-generation IoT devices continues to pose challenges.

Deep learning is widely used in the fields of speech and image recognition [13], [14] because of its ability to extract features better than other methods. Recently, researchers have applied deep learning to wireless communication, namely within Non-Orthogonal Multiple Access (NOMA) and hybrid precoding, Multiple-Input Multiple-Output (MIMO), and Internet of Things applications [15], [16], [17], [18]. One of the benefits of deep learning is that it can efficiently and automatically identify high-level representations of features from interconnected complex datasets [19], [20]. Researchers have discussed a detailed model of power amplifiers (PAs) and digital pre-distortion in RF transmitters using neural networks [21], [22]. For passive radar signal recognition, researchers used an Artificial Neural Network (ANN) with a Multilayer Perceptron (MLP) architecture to determine the carefully built parameters for both single and multiple pulses [23]. Researchers utilized a Probabilistic Neural Network (PNN) to predict the amplitude of eight Wi-Fi cards from the fluctuating amplitudes in Wi-Fi waveforms [24] to determine the feature vectors of the desired target manually. Automated feature extraction, the primary advantage of deep learning, has not yet been mentioned in all the research papers mentioned above. Explainable Artificial Intelligence (XAI) could make it easier to train different machine learning models and make people more confident in the results these systems find. The use of XAI in intrusion detection systems enables them to explain their decision-making process in an easily understandable manner.

While machine learning and deep learning offer several advantages in intrusion detection, existing models for IoT intrusion detection face significant challenges. Many ML-based IDS models require high computational resources, making them unsuitable for IoT devices with limited processing power. Traditional models often lack efficient feature selection, leading to redundant computations and suboptimal detection accuracy. These challenges highlight the urgent need for more efficient and effective models, such as the DL-IID model.

This study introduces the DL-IID model, which is an efficient, lightweight, and explainable deep learning-based IoT intrusion detection framework. Specifically designed to enhance IoT security, the DL-IID model addresses the computational, feature selection, and explainability challenges of existing IDS solutions. The main contributions of this study are as follows.

1) A deep learning architecture that combines DNN and BiLSTM to capture temporal dependencies that are present in both forward and backward directions, improving the detection of complex attack patterns in IoT networks.
2) We apply efficient feature selection using a Genetic Algorithm (GA) following the extraction of features to select the most relevant features that reduce computational complexity while preserving the accuracy of detection.
3) Bidirectional feature extraction and GA-based feature selection are two complementary processes, with the first focusing on learning patterns and the latter optimizing the feature set to increase efficiency.
4) It integrates explainable AI (XAI) using Local Interpretable Model-Agnostic Explanations (LIME) to provide transparency in intrusion detection, increasing trust and ease of interpretation.
5) Lightweight model optimization was performed using post-training dynamic quantization, which reduced the model size to 108.42 KB while maintaining a high detection accuracy (99.84%).
6) A comprehensive performance evaluation using an RF fingerprinting dataset demonstrated that DL-IID outperformed existing IDS solutions in terms of accuracy, precision, recall, and F1 score. This thorough evaluation reassures the effectiveness of the DL-IID model in securing IoT networks.

The proposed DL-IID model provides a flexible and effective solution for securing IoT networks. It offers a balance between high accuracy and low computation, making it suitable for implementation in IoT in the real world.

The remainder of this paper is organized as follows. Section II offers a comprehensive summary of previous studies. In Section III, we provide an in-depth explanation of the methodology used in the proposed scheme. The results of the performance evaluation and discussions are outlined in Section IV, whereas Section V provides the conclusion and future scope of this study. Table 1 presents a list of the abbreviations used in this study.

II. LITERATURE SURVEY

Authentication techniques based on RF fingerprinting are a recognized area of study in wireless networks. Furthermore, approaches that utilize deep learning for the verification of node legitimacy have recently gained popularity. The main

TABLE 1. List of abbreviations used in the paper.

BiLSTM	Bidirectional LSTM
CD	Chi-square Distribution
CFO	Carrier Frequency Offset
CNN	Convolutional Neural Network
DL-IID	Deep Learning-based IoT Intrusion Detection
DNN	Deep Neural Network
GA	Genetic Algorithm
KNN	K-Nearest Neighbor
LIME	Local Interpretable Model-Agnostic Explanations
LSTM	Long Short-Term Memory
LSVM	Linear Support Vector Machine
MAC	Message Authentication Code
MD	Mahalanobis Distance
MSCNN	Multisampling Convolutional Neural Network
NGN-IoT	The Next Generation Networks and IoT
NGNs	The Next Generation Networks
PCA	Principal Component Analysis
RF	Radio Frequency
RNN	Recurrent Neural Network
ROI	Region Of Interest
SVM	Support Vector Machine
XAI	Explainable Artificial Intelligence

objective of feature extraction is to derive information from radio frequency signals to create a distinct identity for each device. RF fingerprinting methods utilize a variety of RF features, such as carrier frequency differences (CFDs), carrier frequency offset (CFOs), channel state information (CSI), discrete wavelet transform (DWT), in-phase and quadrature (I/Q) origin offset, magnitude and phase errors, phase shift differences, power amplifier characteristics, power spectral density (PSD), normalized PSD, radio signal strength (RSS), synchronization frame correlation, and instantaneous phase, amplitude, and frequency [33], [34]. Most current studies focus on extracting RF features from the RF transmitter signal. Table 2 presents the literature on some of the machine learning and deep learning techniques used for intrusion detection.

Mirsky et al. [36] introduced the widely recognized Kitsune solution. The suggested method aims to detect attacks autonomously, without assistance from a monitoring system, by utilizing an autoencoding algorithm to learn typical patterns and analyze abnormal conditions. In their study, Chatterjee et al. [37] employed an artificial neural network to create a distinctive signature using radio frequency fingerprinting characteristics such as frequency offset and I-Q imbalance. This study recommends that RF signature features be adjusted and evaluated to reduce the impact of channel conditions. Tu et al. [38] proposed a method that integrates principal component analysis (PCA) to reduce the number of dimensions and SVMs for classification, utilizing four features, and their method achieves a detection accuracy exceeding 95%. To classify ZigBee devices according to features under the area of significance, Yu et al. [27] devised an RF fingerprinting method. To evaluate the performance, the authors conducted tests under both line-of-sight (LOS) and non-line-of-sight (NLOS) situations, using a multisampling convolutional neural network (MSCNN) to extract features and classification. The proposed method attained a peak accuracy of 97% under LOS conditions.

Aghnaiya et al. [31] proposed a radio frequency fingerprinting technique employing variational mode decomposition (VMD) and exploiting Bluetooth transient data to extract high-order statistical (HOS) features. In contrast, this study employs a Linear Support Vector Machine (LSVM) to classify Bluetooth devices. SLoRa [28] is a commonly employed radio frequency fingerprinting technique for IoT devices. This study introduces an authentication method that utilizes RF fingerprinting, which identifies two features: the CFO and link signatures. By including these features, it is possible to improve the efficiency of eliminating attacks that use impersonation techniques. The classification model uses SVM, and adding the CFO and link signatures makes the identification more accurate.

Ezuma et al. [30] demonstrated the use of RF fingerprinting for both Wi-Fi and Bluetooth systems. The goal of this study was to discover ways to recognize and classify uncrewed aerial vehicles (UAVs). This study used a two-step RF fingerprinting technique. First, the researchers used a naive Bayes method based on the Markov model to extract radio frequency signals. The K-Nearest Neighbor (KNN) model was then used to identify them. Experiments involving 15 distinct UAV types were conducted across various SNR levels to evaluate the method using the analysis of five distinct machine learning models. Jian et al. [35] used a lightweight convolutional neural network (CNN) model to increase the efficiency of fingerprints in a classification framework impressively. The classification processes use a region of interest (ROI). Initially, the researchers preprocessed the raw photos before extracting the regions of interest patterns from the images. The ROI pattern used in the analysis provided input data for the neural network classifiers.

Zong et al. [39] developed an approach by modifying the classic CNN model for the frequency fingerprint detection of VGG-16. The results show that the accuracy was stable as the epochs increased, culminating in a final accuracy of 99.7%. Li et al. [40] utilized the robust KNN model to improve recognition performance by selecting the optimal subset from the generated features. The robustness of the model was evaluated across different SNR levels, achieving a maximum accuracy of 97.86%. This robustness provides a solid foundation for further research and applications in the field. Bovenzi et al. [41] proposed H2ID in a separate study to enhance the efficiency of attack detection. It presents a two-stage approach for identifying attacks. The first stage involves anomaly detection using a lightweight Deep Autoencoder solution. By contrast, the second stage focuses on attack classification using an open-set classification method.

Li and Cetin [42] recently integrated a technique based on deep learning within the waveform space with the concept of RF fingerprinting. The device was identified using the waveform pictures obtained from the original sample, and we recommend using dense neural networks for

TABLE 2. Summary of the related machine learning and deep learning methods in the intrusion detection domain.

Year [Ref.]	Model	Overview	Feature Extraction / Selection	Model Optimization
2018 [36]	Autoencoder	Kitsune aims to minimize labeling efforts by utilizing an Autoencoder to distinguish between normal and abnormal patterns.	Damped Incremental Statistics	No
2019 [38]	PCA + SVM	Utilize PCA for dimensionality reduction and SVM to classify RF fingerprinting features.	PCA	No
2019 [27]	MSCNN	Utilize MSCNN to sort ZigBee devices into groups based on features of interest in a certain area.	MSCNN	No
2019 [31]	LSVM	Utilizes RF fingerprinting and LSVM for classification purposes.	Higher Order Statistics	No
2020 [30]	KNN	Identification and categorization of UAVs by RF fingerprinting methods used on wireless communication protocols.	Neighborhood Component Analysis (NCA)	No
2020 [39]	CNN	Adapt the traditional CNN architecture of VG-16 for frequency fingerprint recognition.	VGG-16	No
2020 [40]	KNN	Employed KNN for classification and improved recognition performance by selecting a compatible feature subset.	RELIEF-F, F Score, Laplacian Score	No
2021 [29]	MDA/ML	Utilize the simple Nelder-Mead bandwidth estimator to mitigate noise in Rayleigh fading environments.	No	Nelder-Mead (N-M) Simplex Algorithm
2023 [32]	MD/CD	An efficient authentication mechanism for IoT nodes in 5G networks utilizing radio frequency fingerprinting when combined with Mahalanobis Distance (MD) and Chi-square Distribution (CD) theories.	Base Stations	No
2023 [50]	CNN	A CNN-based intrusion detection framework with feature selection to enhance accuracy and reduce computational complexity in IoT networks.	ReliefF, Generalized Fisher score, Structured Graph Optimization, etc.	No
2024 [51]	M-MultiSVM	A hybrid machine learning framework for intrusion detection, which addresses problems such as class imbalance and high-dimensional feature space.	Modified single-value decomposition (M-SvD)	Mud ring optimization
2024 [52]	CNN	A CNN-based intrusion detection system for wireless sensor networks using the Aegean Wi-Fi Invasion Dataset (AWID).	No	No
2024 [53]	Decision Tree, Random Forest, Extra Trees, XGBoost	A machine learning-based intrusion detection system using Random Oversampling, Stacking Feature Embedding, and PCA.	Stacking of features, PCA	No

classification purposes. The method achieves approximately 99% accuracy in terms of identification. The study evaluated the performance of the method across multiple scenarios with varying SNR values, achieving a maximum accuracy of 95% in the optimal configuration using the simple Nelder-Mead channel estimator. It effectively reduces the impact of noise on radio operations in the presence of Rayleigh fading [29]. Nguyen et al. [32] propose an authentication method based on the Mahalanobis Distance with Chi-squared distribution theory in their paper. This method relies on the distinct RF signatures of IoT devices to distinguish between legitimate and illegitimate nodes. This study is among the first to utilize RF fingerprinting and distance-based techniques for authenticating 5G-IoT nodes. The framework is evaluated on the European Telecommunication Standards Institute (ETSI) open-source network function virtualization (NFV) platform offered by Amazon Web Services (AWS) to replicate real-world deployment scenarios.

Baldini et al. [50] proposed a hybrid approach that combines convolutional neural networks (CNN) and feature selection techniques for intrusion detection on the Internet of Things. Their work focused on improving the detection performance by reducing the model dimensionality and enhancing interpretability. This study assessed several methods for selecting the most relevant attributes, resulting in better classification accuracy and reduced computational

overhead. The experimental results show that the proposed CNN-based framework, combined with optimal feature selection, outperforms traditional machine learning approaches in detection. Recent research conducted by Turukmane and Devendiran [51] proposed the M-MultiSVM, which is a hybrid machine-learning framework designed for intrusion detection that addresses issues such as class imbalance and high-dimensional feature space. The approach includes preprocessing, advanced synthetic minority sampling techniques to mitigate class differences, and modified singular value decomposition (M-SvD) to ensure efficient feature extraction. The selection of features was optimized using an algorithm called the Northern Goshawk, which reduces dimensionality. For classification, a new multilayer SVM that supports mud rings combines SVM and MLP layers optimized by mud ring optimization to increase the accuracy of detection.

Sadia et al. [52] introduced a robust machine learning-based intrusion detection system for wireless sensor networks using the Aegean Wi-Fi Invasion Dataset (AWID). The authors employed a rigorous preprocessing process, including zero-value processing, structural engineering, and dimensional reduction, which reduced 154 elements to 13 critical attributes. They compared deep learning models, including convolutional neural networks, deep neural networks, and LSTMs, for binary and multilayer attack classification. This study underscores the robustness of CNNs in the extraction and recognition of patterns in WSN data, as validated by metrics such as accuracy, recall, and F1 score. Talukder et al. [53] presented a robust and reliable machine-learning approach to network intrusion detection that addressed the challenges of handling large and unbalanced datasets. Their model uses random oversampling to reduce class imbalance, feature stacking (SFE) to extract meta-features, and principal component analysis (PCA) to reduce dimensionality. This study highlights the importance of data mining and preprocessing techniques in improving detection accuracy, providing a robust solution for real-world cybersecurity applications.

The proposed DL-IID model has several unique features. Unlike many previous studies, DL-IID leverages a genetic algorithm to select the best features, thereby reducing the number of unnecessary computations. Post-training dynamic quantization was also applied, shrinking the model size to 108.42 KB while maintaining a high accuracy of 99.84%. This significant improvement over CNN-based and traditional ML approaches makes DL-IID a suitable choice for IoT devices with limited resources. Furthermore, the integration of LIME into the intrusion detection model enhances transparency and trust in the security decisions.

III. METHODOLOGY

Fig. 1 presents the overall framework of the proposed DL-IID method. We covered the dataset, data preprocessing methods, feature selection techniques, use of XAI, and model quantization. Many researchers base existing approaches for intrusion detection on limited machine learning methods, mainly because of the resource constraints that IoT devices experience when deploying intrusion detection systems. This study introduces an IoT intrusion detection system based on deep learning that employs an intrusion detection model by combining a dual hidden-layer artificial (deep) neural network with a bidirectional long short-term memory. The proposed DL-IID model uses GA-derived features for training purposes. Algorithm 1 shows a structured workflow for the DL-IID model, outlining feature selection,

Algorithm 1 Workflow of the DL-IID Model

Load Dataset
- Load RF fingerprinting dataset (450 IoT devices, 100 samples/device).
- Preprocess:
  1) Handle missing values using mean imputation.
  2) Normalize using StandardScaler.
- Apply K-Means clustering ($K = 2$) to generate initial labels.

Feature Selection using Genetic Algorithm (GA)
- Initialization:
  1) Population size: 20 chromosomes (binary encoding).
  2) Each chromosome: 7-bit string (1 = feature included, 0 = excluded).
- Fitness Evaluation:
  1) Train the DLB model on selected features.
  2) Fitness score $=$ (1 $-$ classification error) $+$ (1 $-$ $\frac{\text{selected features}}{\text{total features}}$).
- Selection: Tournament selection (size 2).
- Crossover: Arithmetic crossover (probability = 0.8).
- Mutation: Uniform mutation (probability = 0.05).
- Stopping Criteria: 40 generations.
- Output: Optimal feature subset.

Train DL-IID Model
- Split data: 80% training, 20% testing.
- Define DNN–BiLSTM architecture.
- Train the combined DNN–BiLSTM architecture using the selected features.

Post-Training Dynamic Quantization
- Convert weights from `float32` to `int8`.
- Retain activation precision dynamically during inference.

Evaluate
- Metrics: Accuracy, Precision, Recall, F1-score, RMSE, MAPE.
- Compare with baseline models.

Apply Explainable AI (LIME)
- For each test sample:
  1) Generate perturbed instances around the sample.
  2) Train a local surrogate model.
  3) Extract feature importance weights.

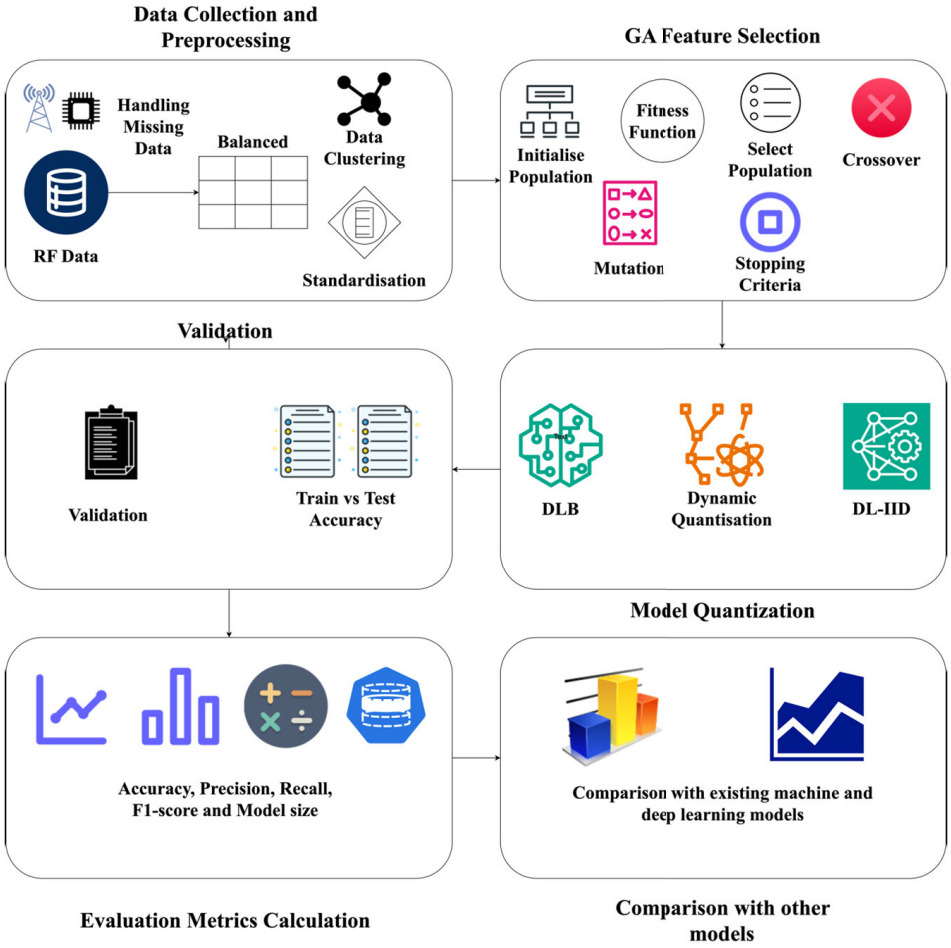


FIGURE 1. Overall framework of the proposed detection scheme.

TABLE 3. Summary of the datasets used.

Dataset	Number of Instances	Number of Features	Number of Classes	Train Size	Test Size
RF Fingerprint	45,000	7	2 (legitimate, malicious)	36,000	9,000
CICIDS2017	2,830,743	85	Multiple attack types + benign	2,264,594	566,149
CICIoMT2024	400,000+	45	18 attack types + benign	320,000	80,000
UNSW-NB15	2,540,044	44	9 attack categories + benign	2,032,035	508,009

model training, quantization, and integration of the LIME. The suggested model, with its high detection accuracy, provides a reassuring solution while maintaining minimal computational complexity.

A. DETAILS OF THE DATASETS USED

The datasets used in this study are widely recognized public datasets for performing intrusion detection experiments. Table 3 provides the summary of the datasets used in this study. Below, we describe the four datasets used in our intrusion detection model.

1) RADIO FREQUENCY (RF) FINGERPRINTING DATASET

The dataset was generated by the authors [32] using the Wireless Waveform Generator toolbox in MATLAB. The researchers utilized 450 IoT devices to generate a dataset with varying RF characteristics in terms of frequency, amplitude, and phase, thereby replicating the nonidealities of RF characteristics. Each device generated 100 RF signal data points, utilizing RF features such as Carrier Frequency Offset (CFO), Amplitude Mismatch, Phase Offset, and other RF parameters, capturing device uniqueness in the experiments. While the dataset effectively simulates IoT device nonidealities, its primary limitation is the controlled noise level. In real-world IoT environments, RF signals may experience more dynamic variations, including interference from other networks, environmental noise affecting signal integrity, and real-time adversarial attacks.

2) CICIDS2017 DATASET

The CICIDS2017 dataset is a valuable resource because it contains benign and frequent attacks that closely resemble

the real-world data. It also contains network traffic analysis results from CICFlowMeter, a tool for the accurate measurement of network traffic with timestamps, source and destination IP addresses, ports, protocols, and attacks. The data retention period started on the morning of July 3, 2017, and lasted for five days, ending on the evening of July 7, 2017. Monday is a typical day and includes only light traffic. The attacks carried out include Brute Force FTP, Brute Force SSH, DoS, Heartbleed, Web-based attacks, infiltration, botnets, and DDoS. On Tuesday, Wednesday, Thursday, and Friday, the authorities executed them in the morning and afternoon [47].

3) CIC IoMT 2024 DATASET

This dataset is of particular interest because it serves as a practical benchmark for the safety of Internet-connected medical devices, that is, the IoT. It contains 18 different cyber-attacks targeting 40 OTMs, providing a variety of protocols commonly used by medical devices, such as Wi-Fi, MQTT, and Bluetooth. The data collection process, which involved network tapping to capture traffic between the switch and IoMT-enabled devices for Wi-Fi and MQTT, helped create datasets for security and profiling. The use of the combination of a malicious PC and a smartphone to capture malicious and benign data for Bluetooth Low Energy (BLE) enabled devices further increases the usability and relevance of the dataset [48].

4) CIC UNSW-NB15 DATASET

The UNSW-NB15 dataset is a comprehensive resource that uses IXIA PerfectStorm to generate a dataset for producing modern normal and abnormal network traffic. Their dataset includes nine categories of attacks and benign traffic. They captured 100GB of network traffic over two days using Argus and Bro-IDS tools to extract information from the intercepted traffic. They extracted 47 categories of features, including Basic, Content, Time, and other features. The system compares the extracted flows with the logs in the ground truth list according to the source IP, target IP, source port, target port, and protocol. If any flow in the ground truth list matched the log, the system flagged it under the attack category. If more than one log in the ground truth list matches the flow, the time stamps are compared, and the log that matches the flow time stamp is selected. If the timestamps cannot be compared, the flow is aborted. After flagging all malicious flows, researchers flagged all other flows as benign, providing a comprehensive view of the network traffic [49].

B. DATA PREPROCESSING AND DATA SPLITTING

Dataset preparation is a crucial step in implementing the proposed deep learning model. Each row in the dataset represents an IoT device. The cleaning process, which involves replacing missing values with the mean of the respective data instances, is a key component of this preparation. The use of the StandardScaler method to normalize the data ensured that the data points were on a balanced scale. Because the raw RF fingerprinting dataset is not predefined with labels, K-means clustering was used to categorize the data points into two groups: legitimate IoT devices (Class 0) and malicious IoT devices (Class 1). Following K-means clustering and feature selection, we converted the dataset into NumPy arrays for efficient processing.

We divided the dataset into training (80%) and testing (20%) sets. Subsequently, we allocated 80% of the initial 80% subset for training purposes, while choosing the remaining 20% for validation. A 20% validation set was employed after each training epoch to identify the optimal model performance and ensure the model accuracy. The development process includes an essential component that ensures the model's efficiency in identifying and predicting previously unseen data.

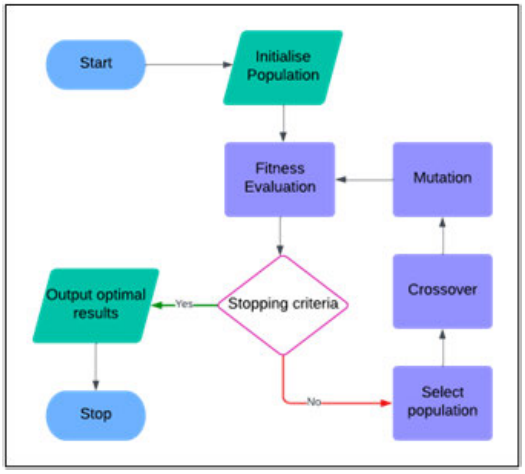


FIGURE 2. A general workflow diagram of the Genetic Algorithm for feature selection.

C. FEATURE SELECTION USING GENETIC ALGORITHM

The initial dataset contains features that add complexity to the intrusion detection process, thereby making the detection method more challenging. To enhance the performance of the intrusion detection approach, we employed a feature selection technique using the GA algorithm. The GA is a heuristic-based search approach inspired by the natural evolution of Charles Darwin. It uses data structures with chromosomes that employ recursive combinations of searching techniques. This natural-selection-based algorithm is a key part of our feature selection process [43]. Fig. 2 presents the fundamental workflow diagram used to design the GA.

We discuss the key components of GA-based feature selection below.

- Chromosome Encoding: A binary string represents each chromosome, where 1 denotes the inclusion of a feature and 0 denotes the exclusion of a feature.
- Fitness Function: A DLB-based classification error function examines feature subsets. The goal is to reduce

TABLE 4. Parameters used in the genetic algorithm for feature selection.

Parameter	Value
Population size	20
Number of features	7
Selection mechanism	Tournament selection
Crossover type	Arithmetic
Crossover probability	0.8
Mutation type	Uniform
Mutation probability	0.05

both the classification error and the total number of features.

- GA parameters: Table 4 lists the parameters used by the GA.

Below, we discuss the procedure for feature selection using the GA.

(I) Initial population: A random binary matrix represents the inclusion or exclusion of features. This ensures sufficient diversity across chromosomes to explore the search space.

(II) Fitness evaluation: The step that separates the best from the rest of the population. DLB was used to identify subsets of features and calculate the classification errors. Fewer features and lower errors resulted in the best fitness scores, paving the way for high-quality feature selection.

(III) Selection: Tournament selection ensures that only the best chromosomes move forward. The selectors then compare these chromosomes and choose the one with the best fitness.

(IV) Crossover: Binary XOR combines the two parent chromosomes to produce the offspring.

(V) Mutation alters bits in chromosomes with a low probability, playing a key role in promoting diversity and maintaining the genetic diversity of the population.

(VI) The new generation is formed by merging elite, crossover, and mutation offspring. This process was repeated for 40 generations or until the GA converged.

(VII) Stopping criteria: The checkpoint that signals the end of the journey. The GA process stops if fitness improvements are minimal over 80 generations, ensuring that we do not continue unnecessarily.

The feature selection process of the Genetic Algorithm (GA) underscores the efficiency of our model. After creating each chromosome by randomly selecting genes (features), we formed a new dataset using only the selected genes for classification. Once the GA converged, it considered only the features represented by the optimal chromosome for the dataset. The GA selected three features, represented as binary 1s, with the positional indices of 1s being 0 and 4.

The GA-based feature selection used in this study employs a wrapper methodology, where the performance of the DNN-BiLSTM model determines the fitness of feature subsets. In contrast to filter methods, which utilize statistical values, the wrapper method ensures that features responsible for improved classification accuracy are retained, thereby achieving both efficiency and robustness.

D. MODEL SELECTION

This study designs an intrusion detection model based on DLB to enhance the capabilities of BiLSTM in extracting nonlinear features and preserving its inherent bidirectional long-distance dependency (DL) characteristics. The use of deep neural networks to uncover hidden information within features and surpass traditional machine learning models is a unique approach to this problem. The DNN design improves the deep nonlinear feature extraction capabilities of BiLSTM. The combination of various network architectures typically increases the number of parameters within the original model, necessitating additional computational resources to train the model. A visual representation of the DNN-BiLSTM model architecture is shown in Fig. 3, which illustrates the data flow through the model, including the input, hidden, BiLSTM, and output layers. When developing an IoT intrusion detection model, it is crucial to consider the model size, real-time performance, and methods for enhancing detection performance. These considerations are key to the model's successful implementation.

The activation functions in Deep Neural Networks (DNNs) facilitate the learning process of the model. These functions allow for a variety of nonlinear transformations on the network data, helping the network learn to perform complex tasks. The most commonly used activation functions in neural networks are sigmoid, softmax, tanh, and ReLU. Deep neural networks extensively utilize the ReLU function, which outputs the same value for positive inputs and zero for negative inputs, facilitating rapid computation. These activation functions generate nonlinear properties by transforming the input into a nonlinear domain. The deep neural networks enable the system to acquire enhanced properties and functionalities through the combination of several nonlinear transformations.

The Long Short-Term Memory (LSTM) network is a specialized type of recurrent neural network (RNN) designed to overcome the gradient vanishing problem often encountered with long-term dependencies. Two key components of LSTM address this challenge: cell storage and cell state management. These components allow the network to independently determine which information to retain and which to discard [25]. Bidirectional Long Short-Term Memory (BiLSTM) is an extension of the standard LSTM model that enhances sequential feature extraction by processing data in both forward and backward directions. Given an input sequence $X = x1, x2, \ldots, x_T$, the equations below describe the computational units involved in the BiLSTM for updating each step as follows:

(I) Forward LSTM computations:

$$\overrightarrow{i_t} = \sigma\left(W_i x_t + U_i \overrightarrow{h_{t-1}} + b_i\right) \quad (1)$$

$$\overrightarrow{f_t} = \sigma\left(W_f x_t + U_f \overrightarrow{h_{t-1}} + b_f\right) \quad (2)$$

TABLE 5. Notations for BiLSTM equations.

Symbol	Description
$i_t^{\rightarrow}, i_t^{\leftarrow}$	Input gate (forward/backward direction)
$f_t^{\rightarrow}, f_t^{\leftarrow}$	Forget gate (forward/backward direction)
$o_t^{\rightarrow}, o_t^{\leftarrow}$	Output gate (forward/backward direction)
$c_t^{\rightarrow}, c_t^{\leftarrow}$	Cell state (forward/backward direction)
$h_t^{\rightarrow}, h_t^{\leftarrow}$	Cell hidden state (forward/backward direction)
$\odot$	Element-wise multiplication
W, U	Weight matrices for input and hidden states
σ	Sigmoid activation function

$$o_t^{\rightarrow} = \sigma\left(W_o x_t + U_o h_{t-1}^{\rightarrow} + b_o\right) \quad (3)$$
$$\tilde{c}_t = \tanh\left(W_c x_t + U_c h_{t-1}^{\rightarrow} + b_c\right) \quad (4)$$
$$c_t^{\rightarrow} = f_t^{\rightarrow} \odot c_{t-1}^{\rightarrow} + i_t^{\rightarrow} \odot \tilde{c}_t \quad (5)$$
$$h_t^{\rightarrow} = o_t^{\rightarrow} \odot \tanh\left(c_t^{\rightarrow}\right) \quad (6)$$

(II) Backward LSTM computations:

$$i_t^{\leftarrow} = \sigma\left(W_i x_t + U_i h_{t+1}^{\leftarrow} + b_i\right) \quad (7)$$
$$f_t^{\leftarrow} = \sigma\left(W_f x_t + U_f h_{t+1}^{\leftarrow} + b_f\right) \quad (8)$$
$$o_t^{\leftarrow} = \sigma\left(W_o x_t + U_o h_{t+1}^{\leftarrow} + b_o\right) \quad (9)$$
$$\tilde{c}_t = \tanh\left(W_c x_t + U_c h_{t+1}^{\leftarrow} + b_c\right) \quad (10)$$
$$c_t^{\leftarrow} = f_t^{\leftarrow} \odot c_{t+1}^{\leftarrow} + i_t^{\leftarrow} \odot \tilde{c}_t \quad (11)$$
$$h_t^{\leftarrow} = o_t^{\leftarrow} \odot \tanh\left(c_t^{\leftarrow}\right) \quad (12)$$

The final output of the BiLSTM is the concatenation of both hidden states:

$$h_t = \left[h_t^{\rightarrow}; h_t^{\leftarrow}\right] \quad (13)$$

where the output gate determines whether the model may keep the previously learned information c_{t-1}, x_t indicates the current input, and h_{t-1} and h_{t+1} indicate the output from the final hidden layers. Table 5 lists the symbols used in the BiLSTM equations. The tanh function transforms any numerical input into a range between -1 and 1. To preserve the information for the current time step, the incoming signals sequentially pass through the gate cells.

The bidirectional LSTM constructs two layers of the LSTM, and these layers work together to compute the hidden parameters in opposing directions. This bidirectional mechanism allows the network to capture long-range dependencies before and after each data point, thereby improving the accuracy of intrusion detection [26]. Tables 6 and 7 present the details of the model layers and parameters used during training, respectively.

E. LOCAL INTERPRETABLE MODEL-AGNOSTIC EXPLANATIONS (LIME)

The LIME technique in XAI is employed to emphasize the dataset features that are significant for training the proposed DL-IID. This enables the model to achieve the intended results [44]. It describes the predictions or detections generated by an ML or DL model by comparing them locally with a model that is easier to understand [45]. Pantazatos et al. [46] say that LIME is helpful because it is model-agnostic, which means it gives clear and easy-to-understand information on the importance of features. Security analysts can understand why different machine learning models make the predictions they do.

F. MODEL QUANTIZATION

Model quantization is a technique aimed at optimizing a model by decreasing the bit size of its variables from the standard 32-bit floating-point representation to a more limited 8-bit one. This method uses an algorithm with a reduced bit size rather than the previously used complete precision techniques. Model quantization reduces the consumption of computational resources and enhances the inference speed of the model. Model quantization generally employs a small 8-bit size to represent model parameters and activation functions. Consequently, we selected dynamic quantization for post-training as our model quantization method. In the post-training dynamic quantization process, weights are converted to int8, as is the case with all quantization methods.

Additionally, we dynamically convert activations to int8 and achieve efficient quantization by utilizing matrix multiplication and convolution during computation. In the research context described in this study, post-training dynamic quantization is the best solution because it works independently of the dataset's training process, reduces the model size, and maintains accuracy within acceptable limits. This makes deployment easier in IoT environments with limited resources.

IV. RESULTS AND DISCUSSION

In this section, we conduct experiments on the proposed IoT model to demonstrate its superior detection performance. We have proposed the DLB model with the GA feature selection method due to its complex structure, which protects against attacks on deep learning models, unlike DNN and BiLSTM, which have simple model structures and are susceptible to adversarial attacks. We offer a concise summary of the experimental setup and the metrics used for evaluation.

1) We performed all evaluations on a Google Colab Platform with 12.7 GB of RAM and 107.7 GB of disk space, using Python 3.10 and the Pytorch framework.
2) The study evaluates the model by measuring accuracy, precision, recall, and F1-score due to the complexity of intrusion detection in the IoT environment. We outline the evaluation metrics and their corresponding calculation formulas below.
   a) **Accuracy**: It measures the proportion of correct predictions.

$$\text{Accuracy} = \frac{T_p + T_n}{T_p + T_n + F_p + F_n} \quad (14)$$

   b) **Precision**: It refers to the ability to identify intrusion instances correctly.

$$\text{Precision} = \frac{T_p}{T_p + F_p} \quad (15)$$

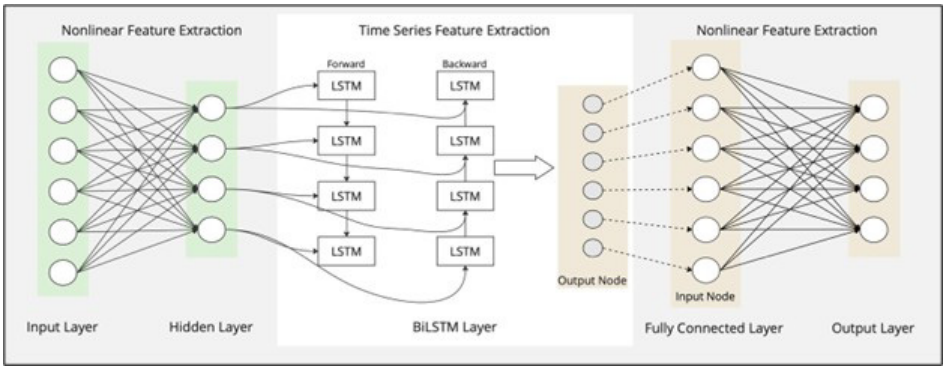


FIGURE 3. The visual representation of the DNN-BiLSTM Model Architecture.

TABLE 6. Layer details of the proposed model architecture.

Layer No.	Layer Type	Input Shape	Output Shape	Activation Function
1	Fully Connected (FC)	(batch_size, input_size)	(batch_size, 128)	ReLU
2	Fully Connected (FC)	(batch_size, 128)	(batch_size, 64)	ReLU
3	Reshape (Unsqueeze)	(batch_size, 64)	(batch_size, 1, 64)	-
4	BiLSTM	(batch_size, 1, 64)	(batch_size, 1, hidden_size*2)	-
5	Fully Connected (FC)	(batch_size, hidden_size*2)	(batch_size, output_size)	Softmax

TABLE 7. Parameters used during the training of the proposed model.

Parameter	Value
Optimizer	Adam
Learning Rate	0.0001
Input Size	7
Batch Size	128
Hidden Size	64
Output Size	2
Epochs	50
Loss Function	Cross-Entropy Loss

c) **Recall**: It refers to the ability to detect intrusion instances.

$$\text{Recall} = \frac{T_p}{T_p + F_n} \tag{16}$$

d) **F1-Score**: It calculates the harmonic mean of precision and recall.

$$\text{F1-Score} = \frac{2T_p}{2T_p + F_p + F_n} \tag{17}$$

e) **False Alarm Rate (FAR)**: It quantifies the probability of incorrectly classifying legitimate IoT traffic as malicious.

$$\text{FAR} = \frac{F_p}{F_p + T_n} \tag{18}$$

where T_p is true positive, T_n is true negative, F_p is false positive, and F_n is false negative.

We can describe DLB + GA as a type of model that combines the DNN architecture with BiLSTM. The design works with features that the genetic algorithm selects to enhance its performance by focusing on the most relevant data attributes. This model is more robust but computationally intensive due to its complex architecture. On the other hand, DL-IID + GA is a lightweight version of DLB + GA. It similarly uses features selected by the genetic algorithm; however, it adopts a simplified architecture and employs dynamic quantization after the training process. This design enables DL-IID + GA to have reduced memory and reduced computation footprint, thus making it more suitable for resource-limited areas where effectiveness matters.

We evaluate the proposed model against the base deep learning model, using features extracted from GA and metrics such as accuracy, precision, recall, F1-score, model size, and error metrics like root mean squared error (RMSE), mean absolute percentage error (MAPE), and others. Tables 8 and 9 present the outcomes of the experiments. The proposed method achieves a 99.84% accuracy, which is equivalent to the DLB + GA method and slightly higher than the DLB method (99.60%). In terms of precision, both DLB + GA and the proposed method achieve a perfect 100%, surpassing DLB (99.33%). The recall score of the proposed method (99.69%) is slightly higher than DLB + GA (99.67%). However, it is marginally lower than DLB (99.87%). Similarly, the F1-score for the proposed method and DLB + GA remains at 99.84%, outperforming DLB (99.60%). One of the most notable advantages of the proposed method is its significantly reduced model size (108.42 KB), making it more efficient compared to DLB (302.14 KB) and DLB + GA (299.66 KB). The results indicate that the proposed DL-IID + GA model maintains high classification performance while reducing computational complexity and storage requirements, making it a more efficient and scalable solution for intrusion detection.

TABLE 8. Performance evaluation of our proposed scheme compared to the baseline model with and without GA-based feature selection.

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	FAR	Model Size (KB)
DLB	99.60	99.33	99.87	99.60	0.0066	302.14
DLB + GA	99.84	100.0	99.67	99.84	0.0000	299.66
Proposed method (**DL-IID + GA**)	99.84	100.0	99.69	99.84	0.0000	108.42

TABLE 9. Error metrics for evaluation of DL-IID to baseline model with and without GA-based feature selection.

Model	Mean Absolute Error (MAE)	Mean Squared Error (MSE)	Root Mean Squared Error (RMSE)	Mean Absolute Percentage Error (MAPE)
DLB	0.0040	0.0040	0.0632	0.13%
DLB + GA	0.0016	0.0016	0.0404	0.33%
Proposed method (**DL-IID + GA**)	0.0016	0.0016	0.0394	0.31%

Below, we outline the error metrics and their corresponding calculation formulas.

1) Mean Absolute Error (MAE): It indicates whether the model overestimates or underestimates values.

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \bar{y}_i\right) \tag{19}$$

2) Mean Squared Error (MSE): It measures the average squared error of predictions.

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \bar{y}_i\right)^2 \tag{20}$$

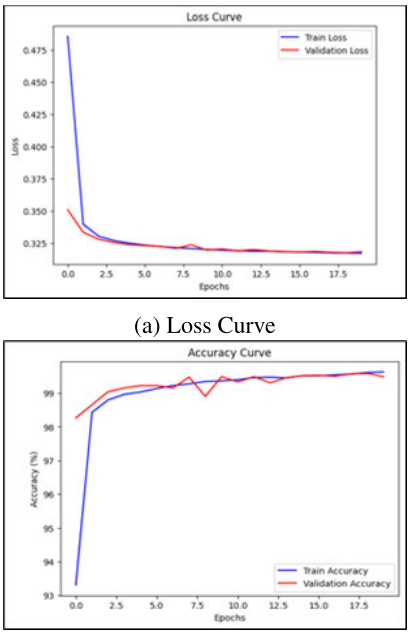
3) Root Mean Squared Error (RMSE): It shows how much the predictions deviate from actual values in absolute terms.

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(y_i - \bar{y}_i\right)^2} \tag{21}$$

4) Mean Absolute Percentage Error (MAPE): It measures the percentage deviation of predictions from actual values.

$$\text{MAPE} = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \bar{y}_i}{y_i}\right| \times 100 \tag{22}$$

Table 9 presents additional error metrics to evaluate the DL-IID model in comparison to DLB and DLB + GA. The metrics include MAE, MSE, RMSE, and MAPE. The proposed method (DL-IID + GA) achieves an MAE and MSE of 0.0016, which is comparable to DLB + GA and outperforms DLB, which has higher errors (0.0040). In terms of RMSE, the proposed method has a value of 0.0394, which is slightly lower than DLB + GA (0.0404) and significantly lower than DLB (0.0632), which indicates a lower prediction error. The MAPE for the proposed method is 0.31%, showing a minor improvement over DLB + GA (0.33%) but slightly higher than DLB (0.13%). These results show the efficiency of DL-IID + GA in minimizing error rates while maintaining high accuracy, demonstrating its reliability as an intrusion detection model.



(a) Loss Curve

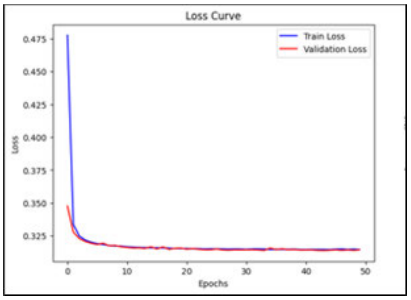
(b) Accuracy Curve

FIGURE 4. Training loss and accuracy curves for DLB Model without GA-based feature selection.

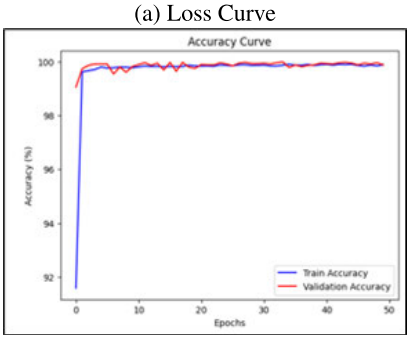
Moreover, Table 10 gives a comparative analysis of the proposed DL-IID model with commonly used machine learning and deep learning-based intrusion detection methods. The DL-IID model surpasses existing IDS solutions in a variety of performance aspects. Compared to CNN [27] (99.60%), KNN [30] (99.40%), and LSVM [28] (99.86%), DL-IID achieves a superior accuracy of 99.84%. In addition, error metrics support the effectiveness of the model as it has the lowest root mean squared error (RMSE) at 0.0394 and a mean absolute percentage error (MAPE) of only 0.31%, which significantly reduces misclassification rates. Figs. 4 and 5 show the graphs of loss and accuracy during the training of

TABLE 10. Classification metrics of the proposed method in comparison with the existing works.

Study	Methodology	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	Model Size (KB)
Wang et al. [28]	SVM-based IoT IDS	99.86	99.91	99.82	99.86	880.97
Ezuma et al. [30]	KNN for RF fingerprinting	99.40	99.84	98.94	99.39	2709.34
Yu et al. [27]	CNN-based RF fingerprinting	99.60	99.42	99.78	99.60	189.89
Proposed DL-IID Model	DNN-BiLSTM-Quantization-based IoT IDS	99.84	**100.0**	99.69	99.84	**108.42**



(a) Loss Curve



(b) Accuracy Curve

FIGURE 5. Training Loss and accuracy curves for DLB model with GA-based feature selection.

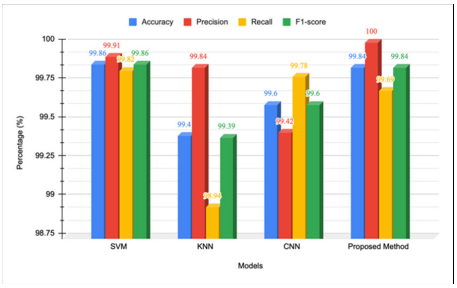


FIGURE 6. Performance evaluation of our proposed scheme compared to existing works.

the baseline deep learning model with and without selected features.

However, the comparative works in Table 10 focus on broader network intrusion detection, while the proposed model targets resource-constrained IoT environments. Nevertheless, the results provide a baseline for deep learning approaches applied to IoT contexts.

The proposed DL-IID model achieves an overall performance evaluation index above 99.5%, as shown in Table 10 and Fig. 6, compared to machine and deep learning models used in other existing works, including SVM-based IoT IDS [28], KNN for RF fingerprinting [30], and CNN-based RF fingerprinting [27]. The DLB model primarily employed the features selected using the GA technique. The DNN model architecture in DL-IID facilitates more efficient feature extraction that may fix the drawbacks of the BiLSTM model, thereby enhancing classification detection capabilities beyond those of the original local techniques while utilizing minimal computational resources. The proposed model achieves a precision of 100%, outperforming all other methods while also maintaining a high accuracy (99.84%), recall (99.69%), and F1 score (99.84%). Compared to previous models, the proposed method significantly reduces the model size (108.42 KB), making it more lightweight than the CNN-based model (189.89 KB), SVM-based (880.97 KB), and KNN-based (2820.48 KB) models. The results show the superiority of the proposed DL-IID model in terms of precision, overall classification performance, and computational efficiency, making it a strong candidate for IoT-based intrusion detection.

Table 11 presents a comparative performance analysis of the binary classification of the proposed DL-IID model across four different datasets: RF Fingerprint, CICIDS2017, CICIOMT2024, and UNSW-NB15. The evaluation metrics considered include accuracy, precision, recall, F1-score, RMSE, and MAPE, providing a comprehensive evaluation of the model's effectiveness. Although CICIDS2017 and UNSW-NB15 are not IoT-specific, they are standard benchmarks widely used for evaluating IoT intrusion detection models due to their coverage of diverse modern attacks and public accessibility.

The performance of the DL-IID model is impressive, achieving exceptional accuracy across all datasets. Its robustness in intrusion detection is evident, with the highest accuracy (100%) seen on the CICIDS2017 dataset, where the GA selected 49 out of 85 features. Similarly, the classification performance on the UNSW-NB15 dataset is outstanding, with an accuracy of 99.99%, precision, recall, and F1-score all at 99.99%, and a very low RMSE (0.0087) and MAPE (0.01%), indicating minimal error rates.

TABLE 11. Results of binary classification comparison of the proposed DL-IID model on different datasets.

Dataset	Total number of features	Features Selected	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	RMSE	MAPE (%)
RF Fingerprint	7	2	99.84	100.0	99.69	99.84	0.0394	0.31
CICIDS2017	85	49	100.0	100.0	100.0	100.0	0.0038	0.00
CICIoMT2024	45	26	97.66	97.66	100.0	98.81	0.1531	0.00
UNSW-NB15	44	20	99.99	99.99	99.99	99.99	0.0087	0.01

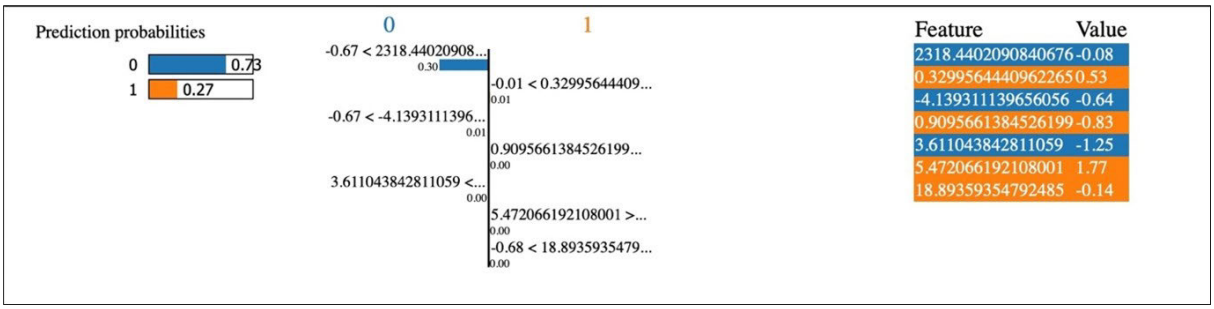


FIGURE 7. LIME results of DLB without GA feature selection.

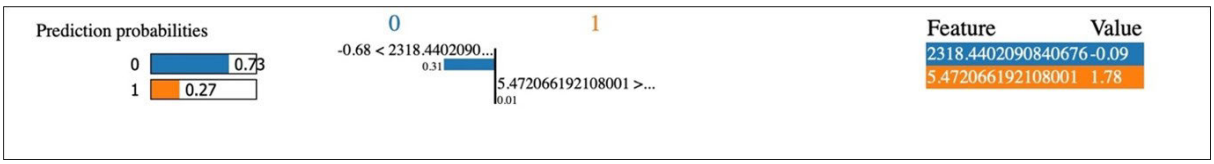


FIGURE 8. LIME results of DLB with GA feature selection.

For the RF fingerprint dataset, the model achieved 99.84% accuracy while using only two features out of 7, highlighting GA's efficiency in feature selection. The recall value of 99.69% suggests strong detection capabilities, with an RMSE of 0.0394 and MAPE of 0.31%, more than other datasets. For the CICIOMT2024 dataset, the model maintained an accuracy of 97.66%, with a recall of 100%, suggesting high sensitivity in identifying malicious activities, although with a slightly elevated RMSE (0.1531). Overall, the results show that the DL-IID model consistently outperforms traditional IDS approaches, with nearly perfect detection accuracy and minimal errors across various datasets. It is essential to note that model training occurs offline on robust systems, while the quantized inference model will be executed on IoT devices or at edge nodes. It is done to ensure feasibility in resource-challenged environments by delegating computationally intensive tasks to the cloud/edge, and IoT devices run only lightweight inference.

To conduct an analysis, we selected the tenth data point from the test set because we want to understand this data point as the first step in the operation of the LIME model. The LIME model produces a new local dataset through the modification of the tenth data point. After that, we proceeded to deploy the LIME model on the DLB and then set the number of features that corresponded with the dimensions of our training data. After that, we used DLB to generate predictions for the disturbed data instances, and then we trained a local interpretable model on those instances using the predictions provided by DLB. As seen in Fig. 7, LIME produces three different results:

- The first result shows the predicted probability assigned to each class label by the original model for the test data point.
- The second result demonstrates the optimal properties that allow the local interpretable model to produce results for changed cases.
- The third result shows a table showing the actual values for the elements.

Fig. 7 confirms that DLB accurately classified the specified test data point as 0, as confirmed by the output of the local model. In the end, we implement the LIME model on DLB using the features extracted from the GA method. Fig. 8 illustrates how Carrier Frequency Offset and Amplitude Mismatch are among the most effective features for classifying legitimate and malicious devices. A CFO deviation can increase the likelihood of a malicious classification. Likewise, Amplitude Mismatch can lead to false positives, underscoring the need to make changes for noisy conditions. While artificial intelligence-based intrusion detection systems (IDS) significantly increase security, they also pose potential security risks and challenges that we must address. An important risk is adversarial attacks, where attackers deceive deep learning models using malicious inputs. Adversarial attacks involve subtle changes in the

input data that cause the model to incorrectly classify threats, while poisoning attacks involve the introduction of manipulated data into the training process, which degrades the performance of the model. Credibility and explainability are issues, given that many deep learning models, such as BiLSTM, operate as non-interpretable model systems that make it challenging for security analysts to understand and verify their decisions. We selected LIME for its agnosticism to models and low computational costs that align with IoT environment demands, but it has some limitations when dealing with complicated nonlinear relations in a dataset. In addition, privacy concerns arise during training on sensitive IoT data, as models may inadvertently remember and disclose confidential information. The urgency of the situation is apparent, and future work is needed to explore adversarial training, different privacy techniques, and robust modeling.

V. CONCLUSION AND FUTURE SCOPE

This research outlines the methodological basis and evaluates the performance of the intrusion detection method based on the radio frequency fingerprinting capabilities present in IoT devices. The DL-IID model has several significant advantages over conventional IDS methods. It incorporates DNN and BiLSTM, which permit bidirectional feature extraction, leading to a higher degree of detection accuracy (99.84%) with very low false alarms when compared to traditional CNN and SVM-based models. In addition, it uses the wrapper-based genetic algorithm effectively, which reduces the size of features, thereby increasing the efficiency of computation. Utilizing the XAI technique (LIME) guarantees better model interpretation, which increases confidence in the process of making decisions. Additionally, dynamic quantization reduces the model size (108.42 KB), making it particularly suitable for IoT networks in smart homes, industrial IoT (IIoT), and Internet of Medical Things (IoMT), where devices operate with constrained resources without significant performance degradation.

However, despite these benefits, there are some limitations. Although efficient in reducing the complexity of models, the quantization process could result in some slight errors in accuracy. In addition, the dataset used in this study, although extensive, does not fully replicate the changing and dynamic reality of IoT situations, as variables such as network traffic congestion, adversarial interference, and protocol-related variability could impact performance. Furthermore, even though LIME enhances explainability, it cannot handle complex, large-scale interactions between features, which may make it less effective in the deep feature extraction task. However, future research needs to concentrate on strengthening the model's resilience against adversarial attacks. It includes examining methods to protect privacy in training and evaluating the model on larger, more diverse datasets to increase generalization. In addition, researchers ought to explore other explainability techniques like SHAP or adversarial interpretability to gain deeper insight into the behavior of models. In addition, federated learning can be used to develop security models for intrusion detection in distributed IoT environments without exposing the data in its raw form, thus ensuring privacy and security.

STATEMENTS AND DECLARATIONS

Availability of data and material: The datasets utilized in this paper for training, testing, and validating the experiment can be freely accessed through the links on the internet: RF fingerprinting dataset https://github.com/ndducnha/mahalanobis_dataset, CICIDS2017 dataset https://www.kaggle.com/datasets/chethuhn/network-intrusion-dataset, CI CIoMT2024 dataset https://www.kaggle.com/datasets/erengencturk/svagdataset, and UNSW-NB15 dataset https://www.kaggle.com/datasets/mrwellsdavid/unsw-nb15?select=UNSW_NB15_testing-set.csv. The source code for the experiments can be accessed through the Github repository: https://github.com/ahwarkhan/DL-IID.

Competing interest: The authors declare that they have no competing financial or non-financial interests that could have appeared to influence the work reported in this paper.

Author's contributions: Author 1 has experimented with and written the complete manuscript. Author 3 has reviewed the manuscript and suggested modifications and changes for further improvements.

Funding: We gratefully acknowledge Md Asdaque Hussain, Faculty of Computer Studies, Arab Open University, for funding the publication of this research.

ACKNOWLEDGEMENT

The completion of this research paper would not have been possible without the support and guidance of Dr. Faisal Anwer, Department of Computer Science, Aligarh Muslim University. His dedication and overwhelming attitude towards helping his students are solely responsible for completing the research paper. The encouragement and insightful feedback were instrumental in accomplishing this task.

REFERENCES

[1] T. Taleb, K. Samdanis, B. Mada, H. Flinck, S. Dutta, and D. Sabella, "On multi-access edge computing: A survey of the emerging 5G network edge cloud architecture and orchestration," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 3, pp. 1657–1681, 3rd Quart., 2017, doi: 10.1109/COMST.2017.2705720.

[2] J. Wright, "KillerBee: Practical ZigBee exploitation framework, inguardians (presentation/technical report)," ToorCon Presentation, Oct. 2009.

[3] K. Zhang, X. Liang, R. Lu, and X. Shen, "Sybil attacks and their defenses in the Internet of Things," *IEEE Internet Things J.*, vol. 1, no. 5, pp. 372–383, Oct. 2014, doi: 10.1109/JIOT.2014.2344013.

[4] G. Dini and M. Tiloca, "Considerations on security in ZigBee networks," in *Proc. IEEE Int. Conf. Sensor Netw., Ubiquitous, Trustworthy Comput.*, Jun. 2010, pp. 58–65, doi: 10.1109/SUTC.2010.15.

[5] P. Gope, O. Millwood, and B. Sikdar, "A scalable protocol level approach to prevent machine learning attacks on physically unclonable function based authentication mechanisms for Internet of Medical Things," *IEEE Trans. Ind. Informat.*, vol. 18, no. 3, pp. 1971–1980, Mar. 2022, doi: 10.1109/TII.2021.3096048.

[6] K. Sood, S. Yu, D. D. N. Nguyen, Y. Xiang, B. Feng, and X. Zhang, "A tutorial on next generation heterogeneous IoT networks and node authentication," *IEEE Internet Things Mag.*, vol. 4, no. 4, pp. 120–126, Dec. 2021, doi: 10.1109/IOTM.001.2100115.

[7] V. Hassija, V. Chamola, V. Saxena, D. Jain, P. Goyal, and B. Sikdar, "A survey on IoT security: Application areas, security threats, and solution architectures," *IEEE Access*, vol. 7, pp. 82721–82743, 2019, doi: 10.1109/ACCESS.2019.2948173.

[8] F. Hussain, R. Hussain, S. A. Hassan, and E. Hossain, "Machine learning in IoT security: Current solutions and future challenges," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 3, pp. 1686–1721, 3rd Quart., 2020, doi: 10.1109/COMST.2020.2986444.

[9] B. Bera, A. K. Das, S. Garg, M. J. Piran, and M. S. Hossain, "Access control protocol for battlefield surveillance in drone-assisted IoT environment," *IEEE Internet Things J.*, vol. 9, no. 4, pp. 2708–2721, Feb. 2022, doi: 10.1109/JIOT.2020.3049003.

[10] A. K. Das, B. Bera, M. Wazid, S. S. Jamal, and Y. Park, "On the security of a secure and lightweight authentication scheme for next generation IoT infrastructure," *IEEE Access*, vol. 9, pp. 71856–71867, 2021, doi: 10.1109/ACCESS.2021.3079312.

[11] T. Alladi, V. Venkatesh, V. Chamola, and N. Chaturvedi, "Drone-MAP: A novel authentication scheme for drone-assisted 5G networks," in *Proc. IEEE Conf. Comput. Commun. Workshops (INFOCOM WKSHPS)*, May 2021, pp. 1–6, doi: 10.1109/INFOCOMWKSHPS51825.2021.9484594.

[12] N. Miguelez-Gomez and E. A. Rojas-Nastrucci, "Antenna additively manufactured engineered fingerprinting for physical-layer security enhancement for wireless communications," *IEEE Open J. Antennas Propag.*, vol. 3, pp. 637–651, 2022, doi: 10.1109/OJAP.2022.3181325.

[13] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Commun. ACM*, vol. 60, no. 6, pp. 84–90, May 2017, doi: 10.1145/3065386.

[14] R. Collobert and J. Weston, "A unified architecture for natural language processing," in *Proc. 25th Int. Conf. Mach. Learn.*, 2008, pp. 160–167, doi: 10.1145/1390156.1390177.

[15] G. Gui, H. Huang, Y. Song, and H. Sari, "Deep learning for an effective nonorthogonal multiple access scheme," *IEEE Trans. Veh. Technol.*, vol. 67, no. 9, pp. 8440–8450, Sep. 2018, doi: 10.1109/TVT.2018.2848294.

[16] H. Huang, Y. Song, J. Yang, G. Gui, and F. Adachi, "Deep-learning-based millimeter-wave massive MIMO for hybrid precoding," *IEEE Trans. Veh. Technol.*, vol. 68, no. 3, pp. 3027–3032, Mar. 2019, doi: 10.1109/TVT.2019.2893928.

[17] H. Huang, J. Yang, H. Huang, Y. Song, and G. Gui, "Deep learning for super-resolution channel estimation and DOA estimation based massive MIMO system," *IEEE Trans. Veh. Technol.*, vol. 67, no. 9, pp. 8549–8560, Sep. 2018, doi: 10.1109/TVT.2018.2851783.

[18] X. Sun, G. Gui, Y. Li, R. P. Liu, and Y. An, "ResInNet: A novel deep neural network with feature reuse for Internet of Things," *IEEE Internet Things J.*, vol. 6, no. 1, pp. 679–691, Feb. 2019, doi: 10.1109/JIOT.2018.2853663.

[19] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, May 2015, doi: 10.1038/nature14539.

[20] C. Zhang, P. Patras, and H. Haddadi, "Deep learning in mobile and wireless networking: A survey," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 3, pp. 2224–2287, 3rd Quart., 2019, doi: 10.1109/COMST.2019.2904897.

[21] N. Benvenuto, F. Piazza, and A. Uncini, "A neural network approach to data predistortion with memory in digital radio systems," in *Proc. IEEE Int. Conf. Commun.*, May 1993, pp. 232–236, doi: 10.1109/ICC.1993.397263.

[22] F. Mkadem and S. Boumaiza, "Physically inspired neural network model for RF power amplifier behavioral modeling and digital predistortion," *IEEE Trans. Microw. Theory Techn.*, vol. 59, no. 4, pp. 913–923, Apr. 2011, doi: 10.1109/TMTT.2010.2098041.

[23] G. B. Willson, "Radar classification using a neural network," *Proc. SPIE*, vol. 1294, pp. 200–210, Aug. 1990, doi: 10.1117/12.21170.

[24] O. Ureten and N. Serinken, "Wireless security through RF fingerprinting," *Can. J. Electr. Comput. Eng.*, vol. 32, no. 1, pp. 27–33, 2007, doi: 10.1109/cjece.2007.364330.

[25] Y. Fu, Y. Du, Z. Cao, Q. Li, and W. Xiang, "A deep learning model for network intrusion detection with imbalanced data," *Electronics*, vol. 11, no. 6, p. 898, Mar. 2022, doi: 10.3390/electronics11060898.

[26] C. Cai, Y. Tao, T. Zhu, and Z. Deng, "Short-term load forecasting based on deep learning bidirectional LSTM neural network," *Appl. Sci.*, vol. 11, no. 17, p. 8129, Sep. 2021, doi: 10.3390/app11178129.

[27] J. Yu, A. Hu, G. Li, and L. Peng, "A robust RF fingerprinting approach using multisampling convolutional neural network," *IEEE Internet Things J.*, vol. 6, no. 4, pp. 6786–6799, Aug. 2019, doi: 10.1109/JIOT.2019.2911347.

[28] X. Wang, L. Kong, Z. Wu, L. Cheng, C. Xu, and G. Chen, "SLoRa," in *Proc. 18th Conf. Embedded Networked Sensor Syst.*, Nov. 2020, pp. 258–270, doi: 10.1145/3384419.3430770.

[29] M. Fadul, D. Reising, T. D. Loveless, and A. Ofoli, "Nelder-mead simplex channel estimation for the RF-DNA fingerprinting of OFDM transmitters under Rayleigh fading conditions," *IEEE Trans. Inf. Forensics Security*, vol. 16, pp. 2381–2396, 2021, doi: 10.1109/TIFS.2021.3054524.

[30] M. Ezuma, F. Erden, C. Kumar Anjinappa, O. Ozdemir, and I. Guvenc, "Detection and classification of UAVs using RF fingerprints in the presence of Wi-Fi and Bluetooth interference," *IEEE Open J. Commun. Soc.*, vol. 1, pp. 60–76, 2020, doi: 10.1109/OJCOMS.2019.2955889.

[31] A. Aghnaiya, A. M. Ali, and A. Kara, "Variational mode decomposition-based radio frequency fingerprinting of Bluetooth devices," *IEEE Access*, vol. 7, pp. 144054–144058, 2019, doi: 10.1109/ACCESS.2019.2945121.

[32] D. D. N. Nguyen, K. Sood, Y. Xiang, L. Gao, L. Chi, and S. Yu, "Toward IoT node authentication mechanism in next generation networks," *IEEE Internet Things J.*, vol. 10, no. 15, pp. 13333–13341, Aug. 2023, doi: 10.1109/JIOT.2023.3262822.

[33] N. Soltanieh, Y. Norouzi, Y. Yang, and N. C. Karmakar, "A review of radio frequency fingerprinting techniques," *IEEE J. Radio Freq. Identificat.*, vol. 4, no. 3, pp. 222–233, Sep. 2020, doi: 10.1109/JRFID.2020.2968369.

[34] X. Guo, Z. Zhang, and J. Chang, "Survey of mobile device authentication methods based on RF fingerprint," in *Proc. IEEE Conf. Comput. Commun. Workshops (INFOCOM WKSHPS)*, Apr. 2019, pp. 1–6, doi: 10.1109/INFOCOMWKSHPS47286.2019.9093755.

[35] W. Jian, Y. Zhou, and H. Liu, "Lightweight convolutional neural network based on singularity ROI for fingerprint classification," *IEEE Access*, vol. 8, pp. 54554–54563, 2020, doi: 10.1109/ACCESS.2020.2981515.

[36] Y. Mirsky, T. Doitshman, Y. Elovici, and A. Shabtai, "Kitsune: An ensemble of autoencoders for online network intrusion detection," in *Proc. NDSS*, 2018, pp. 1–12, doi: 10.14722/ndss.2018.23204.

[37] B. Chatterjee, D. Das, S. Maity, and S. Sen, "RF-PUF: Enhancing IoT security through authentication of wireless nodes using in-situ machine learning," *IEEE Internet Things J.*, vol. 6, no. 1, pp. 388–398, Feb. 2019, doi: 10.1109/JIOT.2018.2849324.

[38] Y. Tu, Z. Zhang, Y. Li, C. Wang, and Y. Xiao, "Research on the Internet of Things device recognition based on RF-fingerprinting," *IEEE Access*, vol. 7, pp. 37426–37431, 2019, doi: 10.1109/ACCESS.2019.2904657.

[39] L. Zong, C. Xu, and H. Yuan, "A RF fingerprint recognition method based on deeply convolutional neural network," in *Proc. IEEE 5th Inf. Technol. Mechatronics Eng. Conf. (ITOEC)*, Jun. 2020, pp. 1778–1781, doi: 10.1109/ITOEC49072.2020.9141877.

[40] Y. Li, Y. Lin, Z. Dou, and Y. Chen, "Research on RF fingerprint feature selection method," in *Proc. IEEE 91st Veh. Technol. Conf. (VTC-Spring)*, May 2020, pp. 1–5, doi: 10.1109/VTC2020-SPRING48590.2020.9129105.

[41] G. Bovenzi, G. Aceto, D. Ciuonzo, V. Persico, and A. Pescape, "A hierarchical hybrid intrusion detection approach in IoT scenarios," in *Proc. IEEE Global Commun. Conf.*, Dec. 2020, pp. 1–7, doi: 10.1109/GLOBECOM42002.2020.9348167.

[42] B. Li and E. Cetin, "Waveform domain deep learning approach for RF fingerprinting," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2021, pp. 1–5, doi: 10.1109/ISCAS51556.2021.9401486.

[43] P. K. D. Pramanik, S. Pal, M. Mukhopadhyay, and S. P. Singh, "Big data classification: Techniques and tools," in *Applications of Big Data in Healthcare*. Amsterdam, The Netherlands: Elsevier, 2021, pp. 1–43, doi: 10.1016/b978-0-12-820203-6.00002-3.

[44] M. T. Ribeiro, S. Singh, and C. Guestrin, "'Why should i trust you?': Explaining the predictions of any classifier," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2016, pp. 1135–1144, doi: 10.1145/2939672.2939778.
[45] S. Patil, V. Varadarajan, S. M. Mazhar, A. Sahibzada, N. Ahmed, O. Sinha, S. Kumar, K. Shaw, and K. Kotecha, "Explainable artificial intelligence for intrusion detection system," *Electronics*, vol. 11, no. 19, p. 3079, Sep. 2022, doi: 10.3390/electronics11193079.
[46] D. Pantazatos, A. Trilivas, K. Meli, D. Kotsifakos, and C. Douligeris, "Machine learning and explainable artificial intelligence in education and training-status and trends," in *Proc. Int. Wireless Internet Conf.*, in Lecture Notes of Institute Computer Sciences, Social Informatics Telecommunications Engineering, Cham, Switzerland. Springer, 2024, pp. 110–122, doi: 10.1007/978-3-031-58053-6_8.
[47] I. Sharafaldin, A. Habibi Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," in *Proc. 4th Int. Conf. Inf. Syst. Secur. Privacy*, 2018, pp. 108–116, doi: 10.5220/0006639801080116.
[48] S. Dadkhah, E. C. P. Neto, R. Ferreira, R. C. Molokwu, S. Sadeghi, and A. A. Ghorbani, "CICIoMT2024: A benchmark dataset for multi-protocol security assessment in IoMT," *Internet Things*, vol. 28, Dec. 2024, Art. no. 101351, doi: 10.1016/j.iot.2024.101351.
[49] N. Moustafa and J. Slay, "UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)," in *Proc. Mil. Commun. Inf. Syst. Conf. (MilCIS)*, Nov. 2015, pp. 1–6, doi: 10.1109/MILCIS.2015.7348942.
[50] G. Baldini, I. Amerini, F. Dimc, and F. Bonavitacola, "Convolutional neural networks combined with feature selection for radio-frequency fingerprinting," *Comput. Intell.*, vol. 39, no. 5, pp. 734–758, Jul. 2023, doi: 10.1111/coin.12592.
[51] A. V. Turukmane and R. Devendiran, "M-MultiSVM: An efficient feature selection assisted network intrusion detection system using machine learning," *Comput. Secur.*, vol. 137, Feb. 2024, Art. no. 103587, doi: 10.1016/j.cose.2023.103587.
[52] H. Sadia, S. Farhan, Y. U. Haq, R. Sana, T. Mahmood, S. A. O. Bahaj, and A. R. Khan, "Intrusion detection system for wireless sensor networks: A machine learning based approach," *IEEE Access*, vol. 12, pp. 52565–52582, 2024, doi: 10.1109/ACCESS.2024.3380014.
[53] M. A. Talukder, M. M. Islam, M. A. Uddin, K. F. Hasan, S. Sharmin, S. A. Alyami, and M. A. Moni, "Machine learning-based network intrusion detection for big and imbalanced data using oversampling, stacking feature embedding and feature extraction," *J. Big Data*, vol. 11, pp. 1–44, Feb. 2024, doi: 10.1186/s40537-024-00886-w.

AHWAR KHAN is currently pursuing the master's degree with the Department of Computer Science, Aligarh Muslim University (AMU). His recent experiences involve working as a Research Assistant with the Department of Computer Science, AMU. He has experience in the field of artificial intelligence, machine learning, and cybersecurity. He has multiple scientific articles published in national/international conferences and journals. He has contributed to several research projects, where he worked on projects pertaining to information security, lightweight cryptography, post-quantum cryptography, and deep learning.

MD ASDAQUE HUSSAIN received the Ph.D. degree in information and communication engineering from South Korea. He is currently an Associate Professor with the Faculty of Computer Studies, Arab Open University (AOU), Bahrain. He have stayed with Purdue University, USA, for collaborative research in SHM, visited AIT, Thailand, for academic collaboration and taught in African nation under UNDP program. His primary research interest lies in wireless sensor/adhoc networks, digital forensics, and structural health monitoring.

FAISAL ANWER received the master's degree in computer application and the Ph.D. degree in information security from Jamia Millia Islamia, New Delhi. He is currently an Assistant Professor with the Department of Computer Science, Aligarh Muslim University (AMU), Aligarh. Prior to joining AMU, he was a Senior Software Engineer with the Computer Science Corporation (CSC), Noida. He has also with CSC, U.K., from 2009 to 2010. He has published several research papers in international/national conferences and journals. His research interests include software security, cryptography, and program robustness.