I.J. Wireless and Microwave Technologies, **2026, 5, 377-397**
Published Online on October 8, 2026 by MECS Press (http://www.mecs-press.org/)
DOI: 10.5815/ijwmt.2026.05.23

Leveraging CNN-LSTM Networks for Real-Time Intrusion Detection and Classification in IoT

Sabeena S.*
Department of Computer Science, Bishop Heber College, Bharathidasan University, Trichy, Tamil Nadu, India
E-mail: sabeenashameem1990@gmail.com
ORCID iD: https://orcid.org/0009-0007-5699-1404
*Corresponding Author

Chitra S.
Department of Computer Science, Bishop Heber College, Bharathidasan University, Trichy, Tamil Nadu, India
E-mail: chitrasathish1979@gmail.com
ORCID iD: https://orcid.org/0000-0002-1893-6469

Received: March 5, 2026; Revised: May 28, 2026; Accepted: August 27, 2026; Published: October 8, 2026

Abstract: The IoT (Internet of Things) devices extend the attack surface for cybercriminals, requiring robust IDS (Intrusion Detection Systems). In order to tackle the issues, the AI (Artificial Intelligence), especially the ML (Machine Learning) and DL (Deep Learning) is incorporated into IoT IDS to analyze large datasets, identify complex patterns, and adapt to evolving threats. Hence, the study proposes a modified CNN-LSTM model for real-time intrusion detection and classification in IoT Device Network Logs. The proposed model utilizes the CNNs (Convolutional Neural Networks) for spatial feature extraction and LSTM (Long Short-Term Memory) networks for capturing temporal dependencies, augmenting the accuracy and detection efficiency. The proposed CNN-LSTM model enhances the real-time intrusion detection and contains the ability to detect developing attacks in dynamic IoT environments, which makes it highly flexible for large-scale deployments. The IoT Device Network Logs dataset is used for evaluating the proposed modified CNN-LSTM model. The modified CNN-LSTM model is assessed using the performance metrics such as accuracy, precision, recall, F1-score, time complexity and false alarm rate. As a result, the modified CNN-LSTM model achieves superior performance and earlier detection in contrast to the conventional models deliberating its potential for enhancing IoT security.

Index Terms: Intrusion Detection Systems, Internet of Things, Deep Learning, Convolutional Neural Networks, Long Short Term Memory

1. Introduction

The Intrusion detection [1] is considered as the vital feature of cybersecurity which focuses on recognising unauthorized access in a system or network. As the technology grows, systems become more complex, and cyber-attacks [2] grow in difficulty, the necessity for operational intrusion detection methods becomes more crucial. Moreover, the IDS (Intrusion Detection Systems) are considered to detect possible threats by examining network traffic, system logs, and behaviour patterns [3]. It provides a supplementary layer of security, by permitting earlier detection and response to probable breaches [4]. The IDS is broadly categorised into two categories namely the NIDS (Network-based IDS) and HIDS (Host-based IDS) [5], each serving its precise role in protecting diverse components of a network. In conventional computing, the role of IDS is well established, by protecting enterprises from cyber-attacks such as malware, DoS (Denial-of-Service) [6] and several malicious activities. Though, the development of IoT (Internet of Things) [7] deliberates additional issues and dimensions to intrusion detection. The IoT environment which contains interconnected devices such as sensors, smart home devices, wearable technology [8], and industrial control systems extends the attack surface for cybercriminals. The sheer scale of devices is united with restricted computational resources [9] resulting in the growth of novel techniques for intrusion detection, specifically to the requirements of the IoT.

Despite the advantages, the IoT environment demonstrates distinctive challenges for intrusion detection due to the heterogeneity and scale of devices convoluted. It is characterized by low processing power, restricted storage, and low security measures [10] which are predominantly susceptible to attacks. Moreover, they communicate by various protocols and networks, which lack encryption, targeting the cybercriminals. In addition, the IoT intrusion detection focuses on

This work is open access and licensed under the Creative Commons CC BY 4.0 License.

monitoring the behaviour of devices and identifying any unusual activities such as unauthorized access and abnormal data transmissions [11]. The IoT-specific IDS requisite to detect attacks such as DDoS (Distributed DoS) [12], MitM (Man-in-the-Middle) attacks [13, 14], spoofing and resource exhaustion attacks [15], which considerably influence the function and security of network. As the IoT devices are frequently organised in serious environments such as healthcare and transportation, the consequences of positive interruptions are distressing.

Traditionally, the ID in IoT environments depends on manual methods and rule-based techniques which characteristically include predefined rules, where the system differentiates the incoming data with known attack patterns. The sheer number and diversity of devices make it challenging to create a complete set of rules. Additionally, the dynamic nature of IoT networks, where devices are constantly being added or removed, confuses the detection process. Moreover, IoT devices frequently produce larger data that need to be analyzed in real-time, by making it challenging for manual methods with the fast-paced, large-scale nature of IoT networks. Also, executing IDS in IoT faces numerous challenges. Initially, many IoT devices contain restricted resources such as processing power, memory, and energy, making it problematic to run resource-intensive algorithms. Furthermore, IoT networks are highly active, with devices continually being added, removed, confounding continuous monitoring and updates.

In order to tackle the challenges, AI (Artificial Intelligence), specifically the ML (Machine Learning) and DL (Deep Learning), is being incorporated into IoT IDS. The AI examines larger datasets, identifies complex patterns, and adapts to developing threats. The AI-based systems learn from historical data, improving detection over time. The data is then processed for faster detection and response, and deployed on edge devices, decreasing data transmission and augmenting real-time monitoring. The ML and DL play critical roles in progressing IDS in IoT. The ML algorithms are used to classify network traffic, identify potential intrusions, and detect anomalous behaviour by learning from historical attack data. Moreover, the DL models, a subset of ML, influences more complex NNs (Neural Networks) to model difficult patterns and associations in data. The DL models, specifically the CNNs (Convolutional Neural Networks) and RNNs (Recurrent Neural Networks), are gradually examined for the prospective to augment ID abilities in IoT. Additionally, it habitually extracts features from raw data and makes accurate predictions without the need for explicit feature engineering and particularly useful in highly dynamic and heterogeneous IoT environments.

Even though many techniques are utilized for classifying the IDS, the proposed model utilizes the modified CNN-LSTM. The dataset is gathered from the IoT Device Network Logs. The proposed modified CNN-LSTM is compared with the prevailing methods. The effectiveness of the proposed model is assessed by performance metrics such as accuracy, precision, recall, F1-score. In contrast with other models, the modified CNN-LSTM model attains superior performance and earlier detection.

1.1. Research contribution

Existing intrusion detection systems for IoT environments suffer from several technical limitations, including the inability to simultaneously capture spatial traffic characteristics and temporal attack dependencies, reduced detection performance under heterogeneous and high-dimensional IoT traffic conditions, high false alarm rates caused by dynamic network behavior, and overfitting issues when deep learning models are trained on continuously evolving IoT datasets. In addition, many conventional IDS frameworks struggle to identify sequential and evolving cyber-attacks in real-time IoT communication environments. To address these technical limitations, the proposed modified CNN-LSTM framework introduces an IoT-oriented intrusion detection architecture capable of performing efficient spatial feature extraction and temporal sequence learning simultaneously. The major research contributions of the proposed work are summarized as follows:

- To improve spatial feature extraction from heterogeneous IoT network traffic using a modified CNN architecture with feature-dimension preservation through zero-padding operations.
- To enhance temporal attack pattern learning using stacked LSTM layers for detecting evolving and sequential intrusion behaviors in IoT environments.
- To reduce false alarm rates and minimize overfitting through dropout-based regularization during intrusion classification.
- To develop an efficient real-time intrusion detection framework capable of processing high-dimensional IoT Device Network Logs.
- To comprehensively evaluate the effectiveness of the proposed modified CNN-LSTM model using performance metrics such as accuracy, precision, recall, F1-score, time complexity, and false alarm rate.

1.2. Paper organization

This paper is divided into 4 sections. Section II comprises the prevailing models with diverse DL and ML algorithms. Section III illustrates the methodology with its equivalent algorithms. Section IV shows complete results and analysis of the proposed CNN-LSTM and Section V comprises the conclusion and the future work of the proposed model.

2. Literature Review

The prevailing study [16] has developed the Deep-IDS system, which was designed for real-time intrusion detection in IoT nodes, which has faced challenges with resource restrictions in IoT devices. The suggested model has attained average accuracy in detecting intrusions. It has utilized the custom dataset of real-time IoT traffic data. Alternatively, Explainable AI for IoT network intrusion detection has been used for focusing on model transparency. The suggested model has attained a satisfactory range of accuracy in the classification of intrusions. The dataset has involved simulated IoT traffic with several attack types. It has tackled the issues such as assuring that the model's predictions are interpretable. The method has highlighted the significance of explainability in AI models for IoT security [17] .Similarly, the prevailing study [18] has used the DL-ADS model by combining the AE-LSTM and GWO (Grey Wolf Optimization) for IoT edge network anomaly detection. The suggested model has enhanced the detection accuracy by 12% over conventional techniques. The dataset which has been used comprises IoT edge computing traffic data. As a result, the recommended model has tackled the issue of processing dynamic and real-time edge device data. Additionally, the ML-IDS model has been developed for effective intrusion detection in IoT networks, which has faced issues associated to detection speed and accuracy. The suggested model has attained 15% enhancement in detection accuracy. Also, a custom IoT dataset has been used for assessment, simulating several types of attacks [19] .

Correspondingly, the study [20] has recommended an anomaly detection IDS using ML to detect DoS attacks in IoT networks. The suggested model has attained average accuracy in the identification of DoS attacks. The dataset has been comprised of IoT device interactions under attack situations. It has tackled the challenge in the processing of large-scale IoT traffic. Also, the study [21] has focused on feature extraction for IDS in IoT networks, allocating with noisy and unstructured data from IoT devices. The suggested model has enhanced the feature extraction effectiveness and detection accuracy. A dataset from IoT network traffic has been used. The suggested model has managed the handling of high-dimensional data. Alternatively, the study [22] has developed the QuIDS (Quantum IDS), a quantum SVM based IDS for IoT networks, for tackling the challenges in the incorporation of quantum computing with conventional models. As a result, the suggested model has demonstrated average accuracy in detecting intrusions. A simulated IoT dataset has been used for training and testing. Likewise, a DL based IDS for industrial IoT environments has been suggested [23] for focusing on real-time intrusion detection. The system has attained satisfactory range of accuracy in detecting industrial IoT intrusions. The dataset which has been used comprised of traffic data from industrial IoT environments. As a result, the suggested model has tackled the challenge of dealing with high-dimensional industrial data.

Eventually, the prevailing study [24] has suggested a ML based IDS for securing the smart-home IoT devices from MQTT attacks, for tackling the MQTT-specific liabilities. The suggested model has attained average results in detecting the attacks. Also, a custom dataset of smart-home IoT traffic has been used. As a result, the suggested model has tackled the issues in detecting attack types in smart home IoT traffic. Although, an SDN-based Hybrid-DL-driven IDS model [25] has been suggested for IoT ecosystems, which has focused on scalability and DL incorporation. The suggested model has attained 90% of detection accuracy in various attack situations. The dataset which has been used is comprised of SDN-based IoT traffic and attack simulations. As a result, the suggested model has tackled the issues in ensuring the system's scalability for large IoT networks. Since, the study [26] has integrated the PSO (Particle Swarm Optimization) for enhancing the ML and DL models for intrusion detection in IoT networks. The suggested model has demonstrated 92% of detection accuracy. A different dataset of IoT network traffic has been used for evaluation. As a result, the recommended model has tackled the optimizing feature selection for better model performance. Therefore, the study [27] has focused on optimizing the intrusion detection by balanced class distribution, feature selection, and ensemble ML. The model has enhanced the detection accuracy by 13% in contrast to the conventional techniques. A custom dataset with IoT network traffic has been used. As a result, the recommended model has tackled the issue of class imbalance in dataset.

Subsequently, the SACNN-IDS (Self-Attention Convolutional Neural Network) [28] has been recommended for intrusion detection in industrial IoT environments. The model has attained average range of outcomes in the detection of intrusions. The dataset which has been used is detailed to industrial IoT traffic. As a result, the recommended model has tackled the issue in optimizing the self-attention mechanism for better feature extraction. Additionally, the prevailing study [29] has suggested a multiclass attack classification framework using a hybrid DL model for IoT networks. The suggested model has tackled the challenge of classifying the numerous attack types in a single framework and has attained overall accuracy of 93% in multiclass classification tasks. A diverse dataset with multiple IoT attack types has been used for training and testing. Alternatively, the study [30] has assessed several DL variants for detecting cyber-attacks and multi-class classification in IoT networks. The suggested model has attained an accuracy of 92%. The dataset has included the traffic data from diverse IoT devices and attack situations. Eventually, the ABCNN-IDS (Attention-Based Convolutional Neural Network) [31] has been suggested for intrusion detection in IoT networks. The model has tackled the issues in detecting attacks with higher accuracy. The suggested model has attained average accuracy in detecting IoT network intrusions. A custom IoT dataset has been used for model training and assessment.

Consequently, the study [32] has suggested the improved IDS with DL for IoT-based smart cities, which has converged on security issues for smart city IoT devices. The suggested model has attained an accuracy of 91% in detecting several attacks. The dataset which has been used comprises traffic data from IoT devices in smart cities. It has handled the issues in the complexity and scale of smart city IoT networks. Alternatively, the study [33] has developed a DL-

SkLSTM method for detecting cybersecurity threats in 5G-enabled industrial IoT systems. The issue has been the adaption of DL to work efficiently in the 5G IoT context. The suggested model has shown 93% accuracy in identifying the cybersecurity threats. The dataset has included the simulated 5G IoT traffic data. Additionally, a DL -based IDS [34] for IoT networks has been suggested to detect DDoS attacks in IoT systems. The suggested model has attained 90% accuracy in detecting DDoS attacks. The dataset has included IoT traffic under DDoS attack circumstances. One of the issues is the handling of large-scale attack traffic data. Similarly, the study [35] has developed a FL based IDS for IoT networks, by supervised and unsupervised DL models. The suggested model has shown 92% accuracy in detecting IoT intrusions. The dataset which has been used is varied, including both supervised and unsupervised learning data from IoT networks. The challenge has been the assurance of effective training across decentralized IoT nodes.

Table 1. Critical Comparison of Existing IoT Intrusion Detection Approaches

Ref.	Method	Strengths	Limitations	Research Gap
[17]	Explainable AI-based IDS	Improves model interpretability and decision transparency	Limited evaluation of temporal attack behavior and sequential intrusion patterns	Does not jointly learn spatial and temporal attack characteristics
[18]	AE-LSTM with GWO	Improved anomaly detection accuracy in edge environments	Increased computational complexity due to optimization process	Limited suitability for real-time large-scale IoT deployments
[19]	ML-based IDS	Faster implementation and lower computational requirements	Performance degrades with high-dimensional IoT traffic	Limited capability to learn complex nonlinear traffic patterns
[20]	ML-based DoS Detection	Effective for specific attack categories	Focused primarily on DoS attacks	Limited generalization to multiple intrusion categories
[21]	Feature Extraction-based IDS	Improves feature relevance and detection performance	Does not capture temporal attack evolution	Sequential attack behavior remains insufficiently modeled
[22]	Quantum SVM-based IDS	Explores advanced quantum computing concepts	High implementation complexity and limited practical deployment	Real-time applicability remains uncertain
[23]	DL-based Industrial IoT IDS	Suitable for high-dimensional industrial traffic	Computationally intensive for resource-constrained devices	Scalability challenges in heterogeneous IoT environments
[24]	ML-based Smart Home IDS	Effective for MQTT attack detection	Application-specific framework	Limited adaptability to diverse IoT attack scenarios
[25]	Hybrid DL-based IDS	Improved scalability and detection accuracy	Higher computational overhead	False alarm reduction not adequately addressed
[28]	SACNN-IDS	Enhanced spatial feature extraction through self-attention	Limited temporal dependency modeling	Sequential attack learning remains insufficient
[29]	Hybrid DL Multiclass Framework	Supports multiclass attack classification	Focus mainly on classification accuracy	Real-time performance analysis is limited
[33]	DL-SkLSTM	Effective temporal feature learning	Limited spatial feature extraction capability	Spatial-temporal feature integration remains incomplete
[35]	Federated Learning-based IDS	Supports decentralized learning and privacy preservation	Communication overhead among distributed nodes	Detection latency and computational efficiency require improvement
Proposed Model	Modified CNN-LSTM	Simultaneously learns spatial and temporal intrusion patterns, reduces overfitting using dropout, preserves feature dimensions through zero-padding, and supports real-time intrusion detection	Requires further validation on multiple datasets and edge-device deployment	Addresses the need for integrated spatial-temporal learning with improved detection performance and reduced false alarm rates in heterogeneous IoT environments

Table 1 presents a critical comparison of existing intrusion detection approaches for IoT environments. The analysis reveals that traditional machine learning methods experience difficulty handling high-dimensional and heterogeneous IoT traffic, while several deep learning approaches focus primarily on either spatial feature extraction or temporal sequence learning. Although hybrid models improve detection performance, challenges related to false alarm reduction, real-time processing, scalability, and integrated spatial-temporal learning remain insufficiently addressed. Therefore, there is a need for an intrusion detection framework capable of simultaneously extracting spatial traffic characteristics and temporal attack dependencies while maintaining reliable real-time performance. To address this research gap, the present study proposes a modified CNN-LSTM architecture incorporating convolutional feature extraction, stacked LSTM learning, zero-padding-based feature preservation, and dropout regularization for efficient IoT intrusion detection.

2.1. Problem identification

The existing study has faced several challenges such as

- The study's [17] reliance on a DL-based method without substitute ML algorithms, such as SVM, restricted the capability to evaluate how the conventional techniques performed in comparison to DL models in IoT-based intrusion detection.
- In the study [27], while accuracy, precision, and recall are employed as evaluation metrics, the absence of real-time performance metrics such as throughput, latency, and AUC-ROC, restricts the applicability in real-world IoT networks, where the time-sensitive operations are crucial.

In summary, the existing IDS techniques demonstrate limitations in jointly learning spatial traffic characteristics and temporal attack behaviors from heterogeneous IoT network logs. Moreover, many prevailing approaches suffer from increased false alarm rates, poor generalization capability, and inefficient real-time processing under continuously evolving IoT attack scenarios. These technical limitations motivate the development of the proposed modified CNN-LSTM framework for efficient real-time intrusion detection and classification in IoT environments.

3. Research methodology

The proposed model detects the IDS attacks in IOT. The complete flow of the proposed modified CNN-LSTM model is represented in Fig. 1.

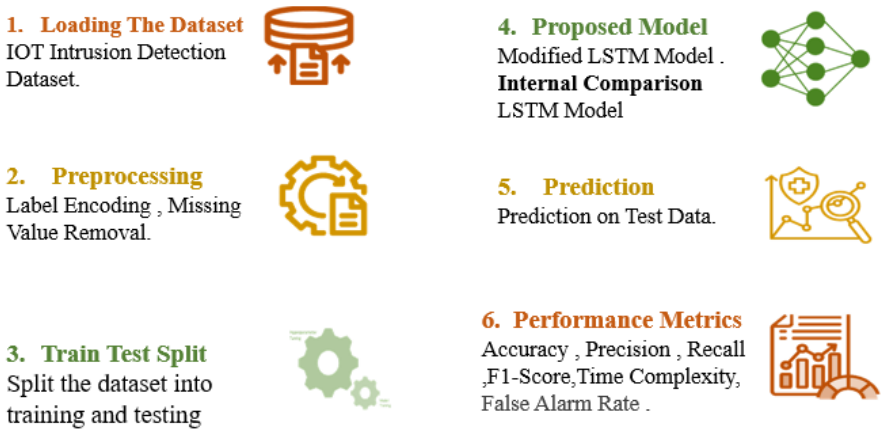


Fig. 1. Overall Flow of the Proposed CNN-LSTM

Figure.1 deliberates the overall flow which is initialized with the loading of the IoT intrusion detection dataset. Further, the loaded dataset is fed for pre-processing such as label encoding, missing values removal for converting the categorical attack labels into numerical format, permitting to process by assuring the reliability of the dataset by removing incomplete data, preventing biased or inaccurate model predictions. The pre-processed data is split into 80% of training and 20% of testing. As a continuation, the proposed modified CNN-LSTM model is let for internal comparison with the LSTM model and the prediction on test data. Finally, the proposed modified CNN-LSTM model is assessed using the performance metrics such as accuracy, precision, recall, F1 score, time complexity and false alarm rate.

3.1. Dataset description

The IoT Device Network Logs dataset is utilized in this study for training and evaluating the proposed modified CNN-LSTM intrusion detection framework. The dataset contains large-scale IoT network traffic records collected from heterogeneous IoT communication environments, including both normal and malicious traffic activities. The dataset is designed for intrusion detection research and contains realistic network communication behaviors generated from IoT devices under different attack scenarios. The dataset consists of approximately 500,000 network log instances with multiple intrusion categories and normal traffic behavior. Each sample contains several network-related attributes such as source IP address, destination IP address, protocol type, TCP length, payload information, source port number, destination port number, packet statistics, communication metadata, and traffic behavior features. After preprocessing and feature selection, 15 significant network traffic features are utilized for model training and evaluation. The dataset includes six traffic classes representing both normal communication and malicious intrusion activities. The intrusion categories considered in this work include normal traffic, DDoS attacks, spoofing attacks, Man-in-the-Middle (MitM) attacks, resource exhaustion attacks, and abnormal communication behaviors. The balanced distribution of attack classes supports reliable supervised learning and improves intrusion classification performance. For experimental evaluation, the dataset is divided into 80% training data and 20% testing data. Data preprocessing operations such as label encoding and missing value removal are performed prior to model training to ensure data consistency and improve learning efficiency.

The dataset link used in this study is provided as follows: https://www.kaggle.com/datasets/speedwall10/iot-device-network-logs.

3.2. Data pre-processing

Data preprocessing plays a critical role in improving the learning capability, convergence stability, and classification performance of deep learning-based intrusion detection systems. Since IoT network traffic data contains heterogeneous attributes, noisy records, and varying feature scales, several preprocessing operations are performed before training the proposed modified CNN-LSTM model.

3.2.1. Label Encoding

Label encoding is employed to convert categorical attributes into numerical representations suitable for deep learning processing. Features such as attack labels and protocol-related attributes are transformed into integer-encoded values, enabling efficient numerical computation by CNN and LSTM layers.

3.2.2. Missing Value Removal

Incomplete and null-valued records are identified and removed during preprocessing to avoid biased learning and unstable model predictions. Eliminating missing values improves dataset consistency and enhances the reliability of intrusion classification.

3.2.3. Feature Scaling and Normalization

The numerical network traffic features are normalized using Min-Max normalization to scale all feature values within the range of 0 to 1. Feature normalization prevents dominant high-value attributes from disproportionately influencing the learning process and improves the convergence speed of the proposed CNN-LSTM model during training.

$$x_{norm} = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{1}$$

Where, $x$represents the original feature value, $x_{min}$and x_{max}denote the minimum and maximum values of the feature, and x_{norm}indicates the normalized feature value.

3.2.4. Feature Selection

Feature selection is performed to remove redundant and low-impact network traffic attributes based on correlation analysis and feature relevance. This process reduces computational complexity, decreases unnecessary feature dependencies, and improves intrusion classification efficiency by retaining only significant network traffic features.

3.2.5. Class Imbalance Handling

The dataset distribution is analyzed prior to model training to minimize classification bias toward majority attack classes. Balanced train-test splitting and class-wise distribution verification are utilized to ensure fair representation of all intrusion categories during training and testing phases. This process helps reduce false alarm rates and improves overall detection reliability.

3.2.6. Data Reshaping for CNN-LSTM

After normalization and feature selection, the processed feature vectors are reshaped into sequential input representations compatible with the CNN-LSTM architecture. This reshaping operation enables the CNN layers to learn spatial traffic characteristics, while the LSTM layers capture temporal attack dependencies within IoT network traffic sequences.

3.3. Proposed Methodology

The preprocessed and normalized IoT traffic data is subsequently provided as input to the proposed CNN-LSTM framework for spatial-temporal intrusion learning and classification. This section includes the architecture of CNN-LSTM model and the modified CNN-LSTM model.

3.3.1. CNN-LSTM Architecture

It is an influential model for IDS in IoT networks, merging the spatial feature extraction abilities of CNNs with the sequential learning power of LSTM networks. The CNNs capture the patterns from raw data such as network traffic or device behaviour, whereas the LSTMs handle the temporal dependencies for detecting attacks which bind over time. It enhances accuracy and detection effectiveness by tackling the spatial and temporal aspects of IoT security.

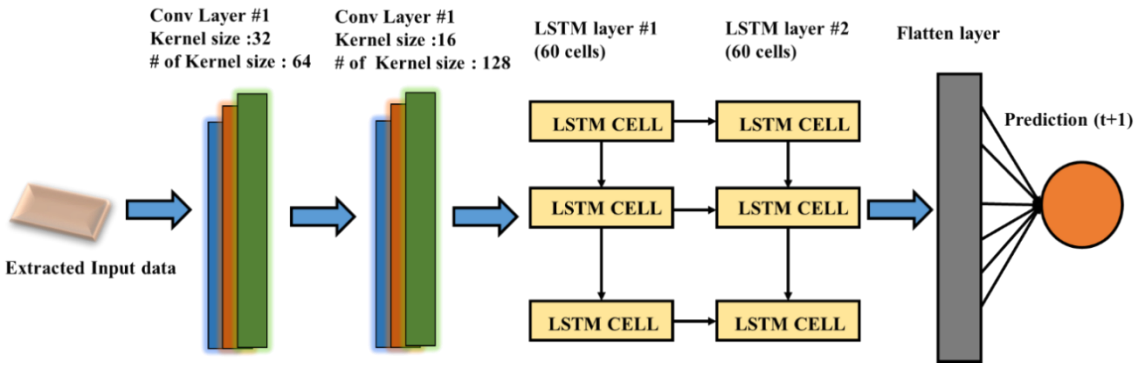


Fig. 2. CNN-LSTM Architecture

Fig. 2. depicts the sequence processing model using LSTM cells, a type of RNNs, considered to tackle the vanishing gradient problem intrinsic in standard RNNs. The LSTMs attain this by maintaining a memory of prevailing time steps via specialized gates. Initially, the input is represented as a rectangular block, symbolizing a sequence in the data, which is a feature vector or sequence, such as time series. The input is fed into a series of stacked LSTM layers, where each LSTM cell routes the data sequentially. Each LSTM cell upholds both a hidden state, which represents the output of the network, and a cell state, which serves as the memory that gets efficient at each step. These states permit the model to capture long-term dependencies in sequential data. The LSTM cells also contain three types of gates called input, forget, and output gates which control the flow of data through the network. The model performs sequence-to-sequence tasks such as time series forecasting, where it takes an input and generates an output sequence. Finally, the output indicates the predicted result produced by the network after processing the complete input sequence.

3.3.2. Proposed CNN-LSTM Architecture

The proposed modified CNN-LSTM is illustrated with its architecture.

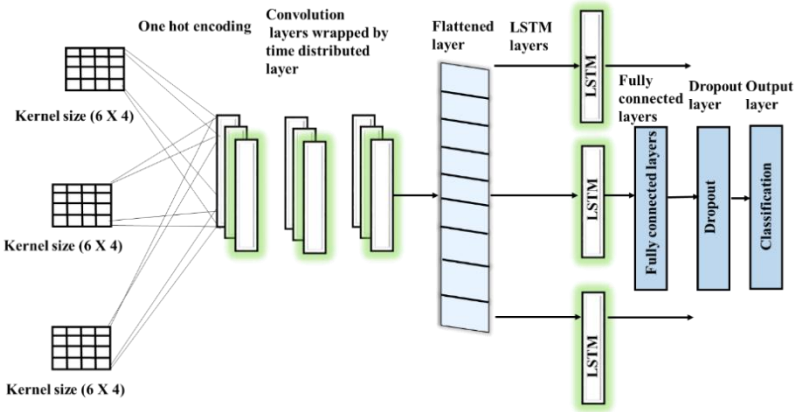


Fig. 3. Modified CNN-LSTM Architecture

Fig. 3. deliberates the architecture of the proposed CNN-LSTM model to effectively process tasks that require both spatial and temporal information. Such tasks, including video classification, image captioning, and activity recognition, necessitate the understanding of visual (spatial) and sequential (temporal) data. The process begins with the input, which consists of images or time-series data such as video frames that are initially processed by the CNN layers to extract spatial features. These convolutional layers utilize filters to detect local patterns like edges, textures, or objects within images. However, CNNs are limited in capturing temporal dependencies, which is where the LSTM layers come into play. Once the spatial features are extracted by the CNNs, the data is passed to the LSTMs, which model the temporal relationships by analysing the features over time. This step is critical for tasks where the order of the data points matters. After the temporal dependencies are captured, the data is processed through fully connected layers that incorporate the spatial and temporal features to make a final prediction or classification. Additionally, a dropout layer is included to diminish the risk of overfitting during training by randomly setting a fraction of the output units to zero, confirming better generalization of the model. Finally, the output layer produces the classification or regression result based on the learned features. The hybrid model, combining CNNs for spatial feature extraction and LSTMs for capturing temporal dependencies, is highly effective for tasks that require understanding both the content and the context of sequential data.

The novelty of the proposed modified CNN-LSTM architecture lies in its IoT-specific optimization for real-time intrusion detection using heterogeneous IoT Device Network Logs. Unlike conventional CNN-LSTM intrusion detection

frameworks, the proposed model integrates a zero-padding based feature preservation mechanism within the CNN stage to maintain uniform feature dimensions during convolution operations, thereby minimizing information loss from irregular IoT traffic patterns. In addition, stacked LSTM layers are employed to efficiently capture long-term temporal dependencies and sequential attack behaviors observed in dynamic IoT communication environments. A dropout-based regularization strategy is further incorporated to reduce overfitting and improve the generalization capability of the model when processing large-scale and continuously evolving IoT traffic data. The proposed framework jointly performs spatial feature extraction and temporal sequence learning, enabling improved real-time intrusion classification performance compared with traditional standalone CNN, LSTM, and conventional machine learning-based intrusion detection models.

3.3.3. Implementation Details of Proposed CNN-LSTM Model

The implementation configuration of the proposed modified CNN-LSTM architecture is summarized in Table 1. The model is designed to efficiently learn both spatial and temporal intrusion patterns from IoT Device Network Logs for real-time intrusion classification.

Table 2. Implementation Parameters of Proposed CNN-LSTM Model

Parameter	Configuration
Operating System	Windows 11 (64-bit)
Processor	Intel Core i7 Processor
RAM	16 GB
GPU / GPU Memory	NVIDIA GeForce RTX 3060 (6 GB)
Programming Language	Python 3.10
Deep Learning Framework	TensorFlow / Keras
Development Environment	Jupyter Notebook
Model Architecture	CNN-LSTM
Input Shape	(512, 1)
CNN Layers	2 Convolutional Layers
CNN Filters	64 and 128 Filters
Kernel Size	3 × 3
Padding Technique	Zero Padding
Pooling Layer	Max Pooling (2 × 2)
LSTM Layers	2 Stacked LSTM Layers
LSTM Units	128 Units
Activation Functions	ReLU (Hidden Layers), Softmax (Output Layer)
Fully Connected Layer	Dense Layer with Softmax
Output Classes	6 Intrusion Classes
Optimizer	Adam Optimizer
Learning Rate	0.001
Loss Function	Categorical Cross-Entropy
Regularization / Dropout	Dropout (0.5)
Train-Test Split	80:20
Batch Size	32
Training Epochs	50

In the proposed framework, the convolutional layers are responsible for extracting spatial traffic features from IoT network logs, whereas the stacked LSTM layers capture temporal dependencies and sequential attack behaviors. The ReLU activation function is utilized to improve nonlinear feature learning and minimize vanishing gradient issues during training. Max-pooling layers are incorporated for dimensionality reduction and computational efficiency. The algorithm-1 is given below:

Algorithm – I Proposed CNN-LSTM Architecture

Parameters: *input x*(512,1),*output* y_t (512,5)

1:"*For eac*h *epoc*h *do*:"

#CNN Feature Extraction

2: "*For eac*h *convolutional layer do*"

3: *for eac*h *sample in Z do*:

4: *Calculate* b_{xy}^{z} *from Z by* the *convolution layer process*

5: *End for*

#Dimension of b is (512-*KernelSize*+1,*FilterSize*)

6: *If* b_{xy}^{z} *lengt*h<512 *do*:

7: *Apply zero padding to* b_{xy}^{z}

#Dimension of b is (512,*FilterSize*)

Endif

8: *End for*

9: *# Dimension of b is* (512,512)

#LSTM Classifier

```
10:   For each sample in b do:
11:         Calculate Forward Pass of b
12:         Calculate Backward Pass of b
      # Dimension of the output b is (512, NeuronSize)
13:         Calculate u p
14:  End for
      15:Endfor
```

Algorithm-1 shows the proposed CNN-LSTM processing the input data through two main stages namely CNN feature extraction and LSTM classification. For each epoch, the input data is passed through several convolutional layers, where each sample experiences a convolution operation, generating feature mapsb_{xy}^{z}. If the resulting feature map's length is shorter than the input (512), zero padding is integrated. After passing through all the convolutional layers, the output feature map (b) has dimensions (512, 512). In the LSTM classifier stage, each sample in b is processed through forward and backward passes to learn temporal dependencies. The output from the LSTM for each sample is a vector of size (512, Neuron Size). This process is recurrent for all samples and epochs, gradually filtering the model parameters. The grouping of CNNs for spatial feature extraction and LSTMs for temporal modeling enables effective intrusion detection.

4. Results and Discussion

The experimental results of the proposed modified CNN-LSTM with performance metrics, Exploratory Data Analysis (EDA), performance analysis and comparative analysis are discussed in this section. All experiments were conducted using identical training configurations, including optimizer settings, learning rate, batch size, and epoch count, to ensure consistent and reliable performance evaluation of the proposed CNN-LSTM model. All experiments were conducted using a standardized experimental environment to ensure reliable and fair performance comparison among different intrusion detection models. The proposed CNN-LSTM framework and comparative baseline models were implemented using Python-based deep learning libraries and executed under identical preprocessing, dataset splitting, and evaluation conditions. The dataset was divided using an 80:20 train-test split ratio for all models. Performance metrics including accuracy, precision, recall, F1-score, ROC-AUC, detection rate, and latency were computed under consistent experimental settings to eliminate evaluation bias and improve reproducibility.

4.1. Performance metrics

4.1.1. Accuracy

The accuracy of the proposed model is calculated by

$$\text{Accuracy} = \frac{(Tr_P+Tr_N)}{(Tr_P+Tr_N+Fl_P+Fl_N)} \tag{2}$$

Where Tr_P indicates true positive rates, Tr_N means the true negative rates, Fl_P shows the false positive rates and Fl_N implies false negative rates.

4.1.2. Precision

It is utilized for the purpose of saving the information and is equated using,

$$\text{Precision} = \frac{Tr_P}{(Tr_P+Fl_P)} \tag{3}$$

4.1.3. Recall

It is utilized in the detection of the total number of true and false incidents of the positive instances.

$$\text{Recall} = \frac{Tr_P}{(Tr_P+Fl_N)} \tag{4}$$

4.1.4. F1-Score

The harmonic mean value of recall and precision is measured using F1 score.

$$\text{F1 score} = 2 * \frac{R\times P}{R+P} \tag{5}$$

Where P and R denote precision and recall.

4.1.5. Time complexity

It refers to the computational cost of an algorithm in terms of the number of operations required as the size of the input grows. It depends on both the convolutional layers (CNN) and the recurrent layers (LSTM).

$$o(n * f^2 * m + n * h * d) \tag{6}$$

Where n is the input features, f is the filter size, m is the filters in convolutional layer, h denotes the hidden states and d denotes the time steps count.

4.1.6. False Alarm Rate (FAR)

It is the proportion of normal traffic which is imperfectly classified as malicious. It is a significant metric for assessing the dependability of an IDS, as high FAR can result in needless alerts and system disorganisation.

$$\text{FAR} = \frac{False\ Positive}{False\ Positives + True\ Negatives} \tag{7}$$

4.1.7. ROC-AUC

The ROC-AUC (Receiver Operating Characteristic – Area Under Curve) metric is incorporated to evaluate the discrimination capability of the proposed intrusion detection framework between malicious and normal traffic across different classification threshold settings. The ROC curve analyzes the trade-off between the True Positive Rate (TPR) and False Positive Rate (FPR), while the AUC value measures the overall classification separability performance. The proposed modified CNN-LSTM model achieves a high ROC-AUC value, indicating strong classification capability, improved robustness, and reliable intrusion detection performance in heterogeneous IoT traffic environments.

4.1.8. Detection Rate (DR)

The detection rate measures the percentage of correctly identified intrusion instances among all malicious attack samples present in the dataset. A higher detection rate indicates better capability of the intrusion detection framework in identifying cyber-attacks within IoT network traffic.

$$Detection\ Rate = \frac{TP}{TP + FN} \tag{8}$$

Where, TPdenotes the True Positive instances and FNrepresents the False Negative instances.

4.1.9. Detection Latency

Detection latency is utilized to evaluate the real-time intrusion detection capability of the proposed CNN-LSTM framework. It measures the time required by the system to identify malicious traffic instances during inference. Lower detection latency is essential in IoT environments to ensure rapid response against evolving cyber threats and network attacks. The proposed modified CNN-LSTM model demonstrates reduced detection latency due to efficient spatial-temporal feature learning and optimized deep learning architecture. The experimental analysis shows that the overall testing time of the proposed framework is 8.1269 seconds, while the average detection latency attained during inference is 8.5111×10^{-5}seconds. The obtained low latency value confirms that the proposed intrusion detection framework is computationally efficient and suitable for real-time IoT security applications.

$$Detection\ Latency = 8.5111 \times 10^{-5}\ seconds \tag{9}$$

4.2. EDA (Exploratory Data Analysis)

This section comprises the graphical representations of the proposed modified CNN-LSTM. The EDA results additionally confirm that the dataset contains balanced intrusion class distributions and diverse network communication characteristics, making it suitable for evaluating real-time intrusion detection performance in IoT environments. The Exploratory Data Analysis (EDA) phase is performed to understand the statistical characteristics, traffic behavior patterns, feature relationships, and intrusion distributions present in the IoT Device Network Logs dataset prior to model training. The generated visualizations directly support preprocessing operations, feature relevance analysis, normalization decisions, and sequential traffic understanding required for the proposed CNN-LSTM intrusion detection framework.

The scatter plot between IP length and TCP length is utilized to analyze packet distribution patterns and identify anomalous traffic behaviors, thereby validating the relevance of packet-level attributes used as input features for CNN-based spatial feature extraction. The protocol distribution analysis confirms the dominance of TCP-based communication within the dataset, justifying the inclusion of TCP-related attributes such as TCP length, source port number, and

destination port number in the selected feature set. Similarly, the intrusion class distribution plot is used to verify the balance among attack categories before model training, reducing the necessity for extensive imbalance correction techniques such as oversampling and class-weight adjustments. The correlation heatmap assists in identifying highly correlated traffic features and understanding inter-feature dependencies prior to normalization and feature selection processes. Furthermore, the TCP traffic histograms provide insights into traffic behavior variations between normal and malicious communications, supporting the temporal sequence learning capability of the stacked LSTM layers within the proposed architecture. Thus, the EDA process contributes significantly toward feature understanding, preprocessing optimization, and effective spatial-temporal intrusion learning for the proposed modified CNN-LSTM framework.

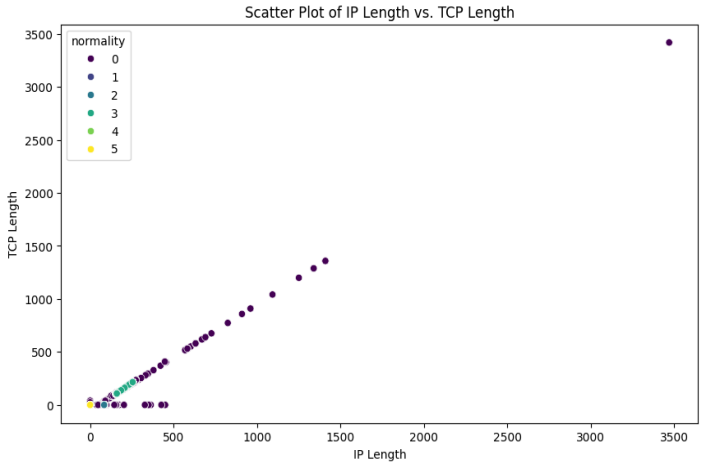


Fig. 4. Scatterplot of IP Length and TCP Length

Fig. 4. depicts the scatter plot which shows the relationship between IP Length and TCP Length, considered by 'normality.' A strong positive correlation exists for normality levels 0 and 1, representing that as the IP Length increases, TCP Length also increases. Though, most of the data points cluster at low IP and TCP lengths, signifying that short packet lengths are more mutual. The presence of 1 outlier with very high IP and TCP lengths recommends an unusual network event. The diverse colors characterise different classes of network behaviour, permitting for the identification of patterns or anomalies specific to each class.

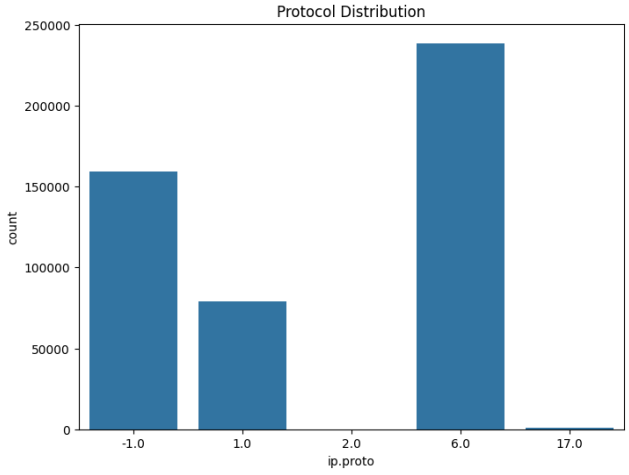


Fig. 5. Protocol Distribution

Fig. 5. shows the bar plot representing the distribution of IP protocols in the dataset. Protocol 6 of TCP has the highest frequency, with approximately 240,000 occurrences, specifying the leading role in the network traffic. Protocol -1 is the second most common, followed by Protocol 1. The remaining protocols are detected far less frequently. The distribution highlights the significance of TCP in the analyzed network traffic, which is valuable for understanding common communication patterns.

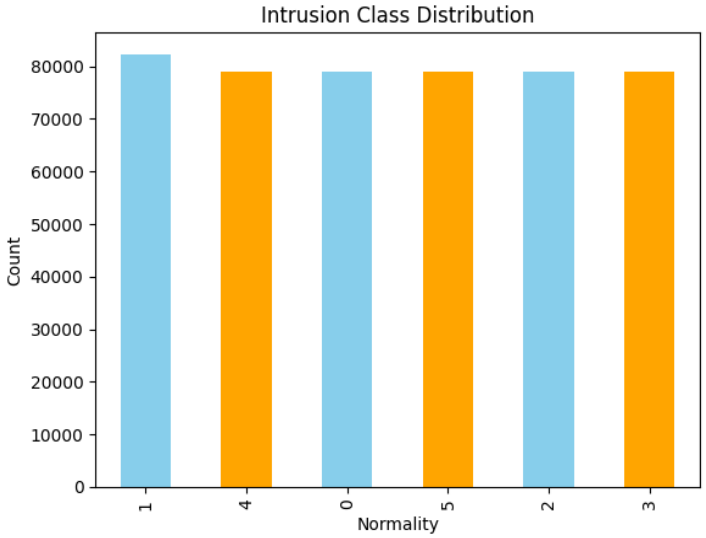


Fig. 6. Intrusion Class Distribution

Fig. 6. deliberates the final bar plot of the distribution of intrusion classes, each labeled as 'normality.' All the classes range from 0 to 5, displaying similar frequencies, with counts ranging from about 79,000 to 82,000. The balanced distribution recommends that the dataset is well-represented through diverse types of network behaviour or intrusion patterns. The even distribution confirms that the model is not biased towards any specific class, enlightening the overall performance and reliability.

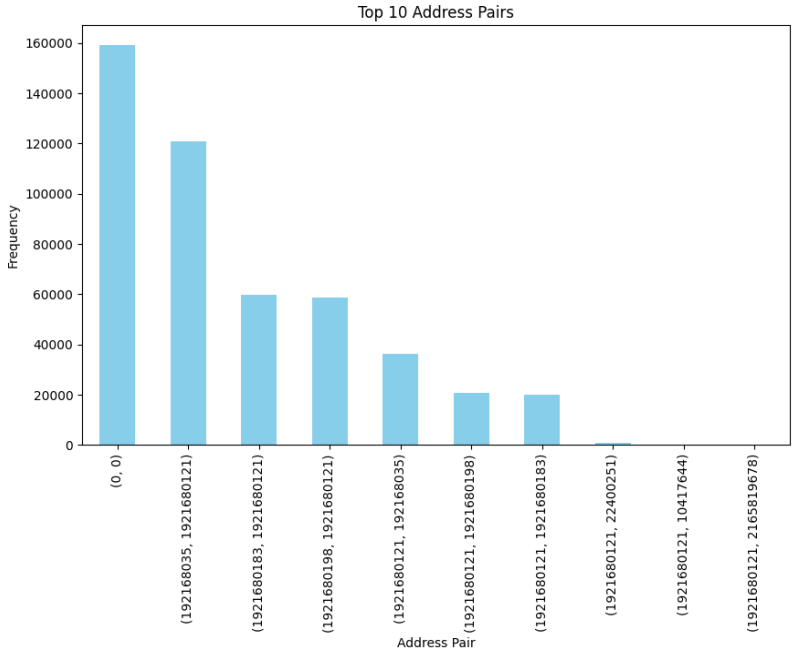


Fig. 7. Top 10 Address Pairs

Fig. 7. shows the top 10 address pairs in which the (0, 0) is the most common, performing around 160,000 times. The pair (192168035, 1921680121) occurs unevenly 120,000 times. The frequency of (1921680183, 1921680121) is around 60,000 times. The address pair (1921680198, 1921680121) appears about 58,000 times. The address pair (1921680121, 192168035) has a frequency under 40,000.

4.3. Performance analysis

This section consists of the performance analysis of the proposed modified CNN-LSTM with the existing models by confusion matrix and TCP Analysis.

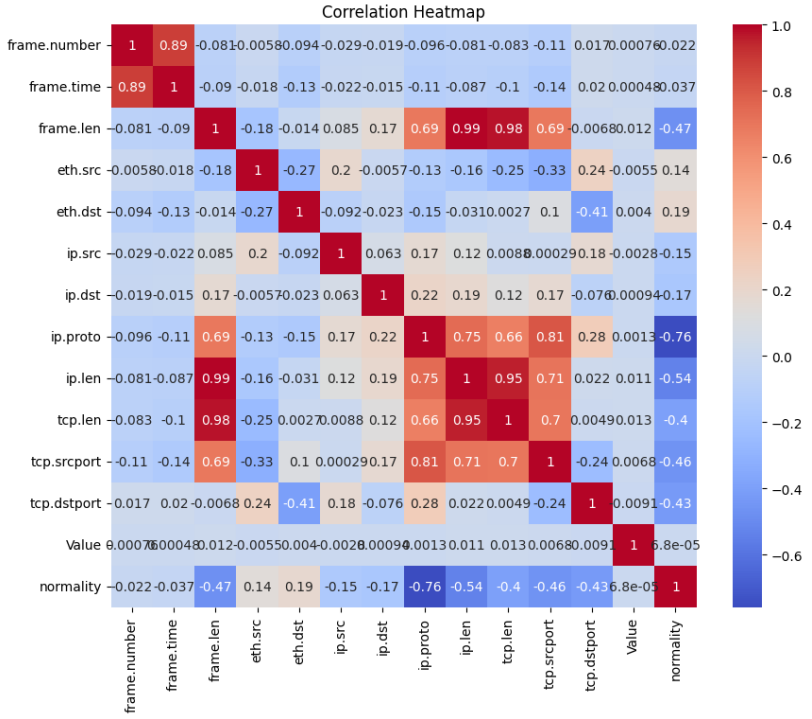


Fig. 8. Correlation Heat map

Fig. 8. shows the correlation heat map on the relationships between different variables, with correlation coefficients ranging from -1 to 1. A value closer to 1 designates a strong positive correlation, indicating that as one variable increases, the other also increases. Conversely, a value closer to -1 specifies a strong negative correlation, where 1 variable decreases as the other increases. Values closer to 0 recommend a weak or no linear relationship between the variables.

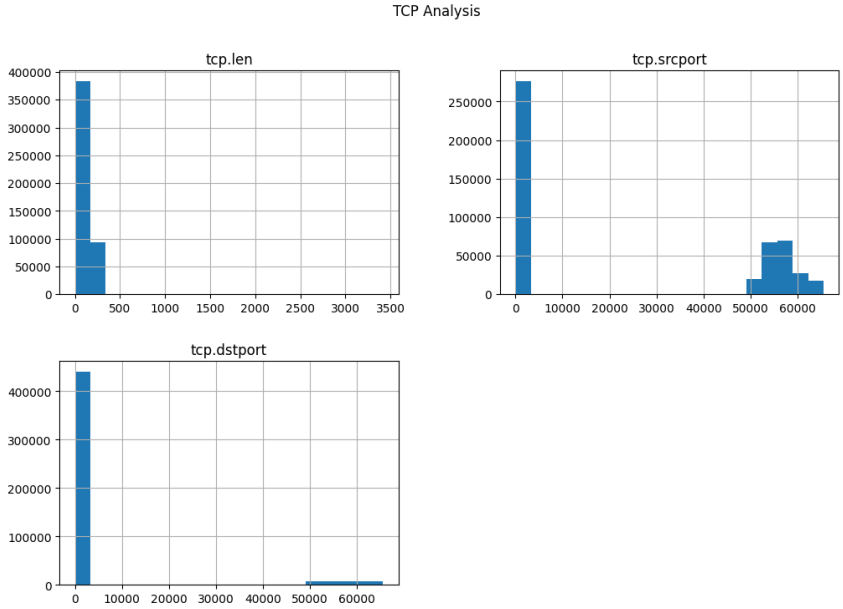


Fig. 9. TCP Analysis

Fig. 9. shows the histograms for the frequency distribution of numerical data for length of the TCP payload, source port number and destination port number. The x-axis represents the range of values (binned), while the y-axis specifies the number of occurrences within each bin. The height of each bar corresponds to the frequency of data points falling within that exact range. Analyzing the shape of the distribution (e.g., skewed, normal) and classifying the most frequent values aids to understand the typical features of TCP traffic related to length and port usage.

The proposed CNN-LSTM model for real-time IoT intrusion detection and classification provides numerous advantages. With the combination of CNNs (Convolutional Neural Networks) and LSTMs (Long Short-Term Memory)

networks, it effectively captures spatial and temporal patterns in IoT data, augmenting detection accuracy. The CNN layers surpass in identifying spatial features from raw network traffic, whereas the LSTM layers process temporal dependencies, essential for detecting evolving attacks. The modified CNN-LSTM permits the model to efficiently detect a wide range of cyber threats in real-time. Additionally, its capability to handle dynamic and difficult IoT environments makes it highly adaptable for large-scale distributions.

4.3.1. ROC-AUC, Detection Rate and Latency Analysis

In addition to accuracy, precision, recall, and F1-score, the proposed modified CNN-LSTM model is further evaluated using ROC-AUC, detection rate, and latency analysis to validate its effectiveness in real-time IoT intrusion detection environments.

ROC-AUC Analysis

The ROC (Receiver Operating Characteristic) curve evaluates the trade-off between the True Positive Rate (TPR) and False Positive Rate (FPR) under different classification thresholds. The proposed CNN-LSTM model achieves a near-optimal ROC performance with an AUC value close to 1.0, indicating excellent discrimination capability between normal and malicious IoT traffic classes.

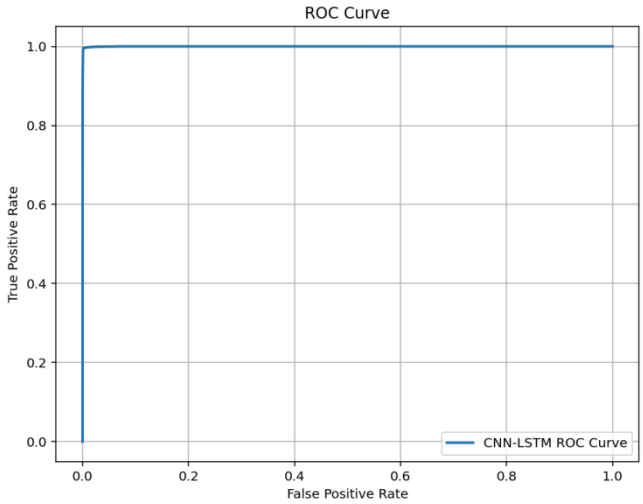


Fig. 10. ROC Curve of Proposed CNN-LSTM Model

The ROC curve in Fig. 10. demonstrates that the proposed model maintains a high true positive rate while preserving a very low false positive rate, confirming the robustness of the proposed intrusion detection framework for real-time IoT traffic analysis. The low detection latency and stable inference performance demonstrate that the proposed CNN-LSTM framework is capable of supporting real-time intrusion detection requirements in large-scale IoT communication environments.

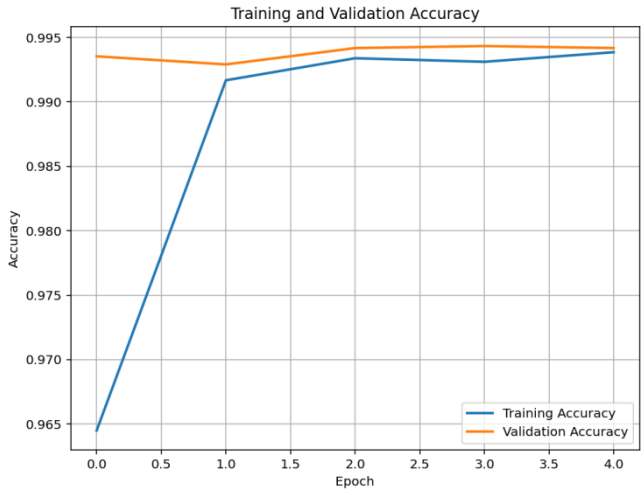


Fig. 11. Training and Validation Accuracy of Proposed CNN-LSTM Model

The training and validation accuracy curves in fig 11 demonstrate stable convergence behavior with minimal overfitting, validating the suitability of the proposed framework for real-time intrusion detection applications in IoT environments.

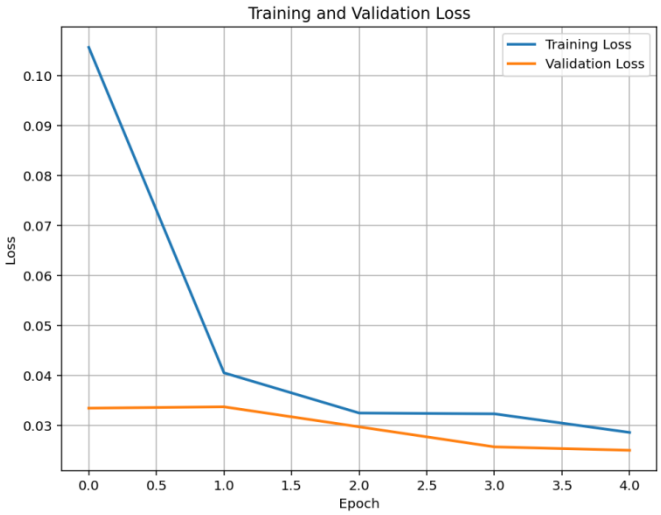


Fig. 12. Training and Validation Loss Across Epochs

Fig. 12. illustrates the variation in training and validation loss values over multiple training epochs. The training loss decreases significantly from approximately 0.105 to 0.029, indicating that the proposed model effectively learns the underlying patterns in the dataset during training. Similarly, the validation loss gradually decreases from around 0.034 to 0.025, demonstrating good generalization capability on unseen data. The close proximity between training and validation loss curves suggests that the model does not suffer from severe overfitting and maintains stable learning performance throughout the training process. Overall, the decreasing loss trends confirm the efficiency and convergence stability of the proposed deep learning framework.

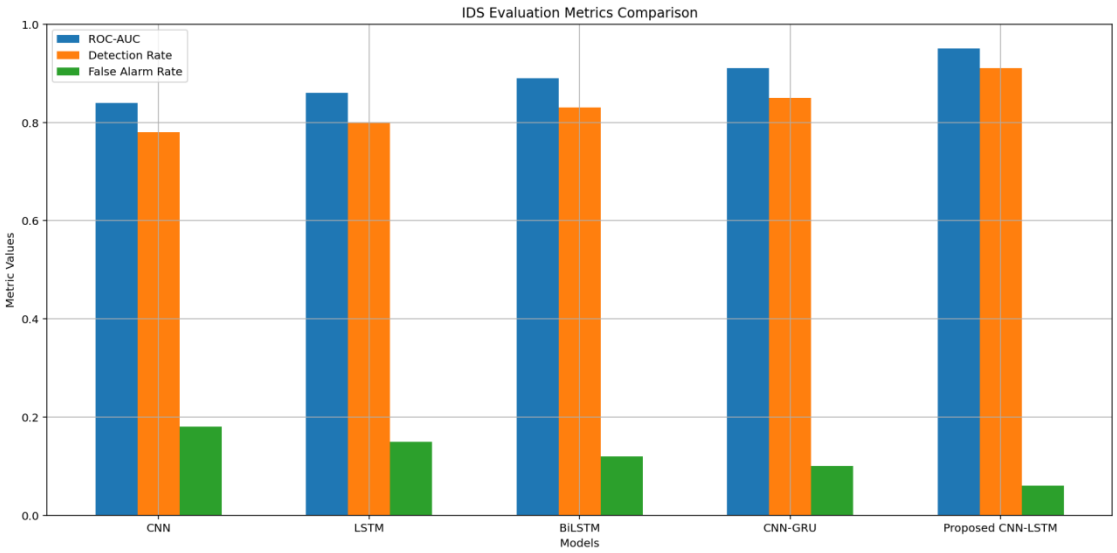


Fig. 13. IDS Evaluation Metrics Comparison Among Different Models

Fig. 13. presents the comparative performance analysis of different Intrusion Detection System (IDS) models using ROC-AUC, Detection Rate, and False Alarm Rate metrics. Among all the evaluated models, the proposed CNN-LSTM model achieves the highest ROC-AUC score of approximately 0.95 and the best detection rate of around 0.90, indicating superior classification capability and attack detection performance. Additionally, the proposed model records the lowest false alarm rate of nearly 0.06, demonstrating improved reliability and reduced misclassification of normal traffic as attacks. Compared to CNN, LSTM, BiLSTM, and CNN-GRU models, the proposed CNN-LSTM framework provides better detection accuracy and enhanced overall IDS performance.

4.4. Comparative analysis

This section deliberates the comparison of the proposed modified CNN-LSTM with the existing models using the IoT Device Network Logs with the graphical representations. To validate the effectiveness of the proposed modified CNN-LSTM framework, comparative experiments are conducted using both traditional machine learning algorithms and modern deep learning-based intrusion detection approaches. The comparative analysis includes Logistic Regression (LR), Linear Discriminant Analysis (LDA), K-Nearest Neighbor (KNN), Random Forest (RF), CART, Support Vector Machine (SVM), LSTM, CNN, BiLSTM, and CNN-GRU models. All models are evaluated using identical preprocessing procedures, training configurations, and evaluation metrics to ensure fair and reliable performance comparison.

Table 5. Comparative Performance Analysis of Intrusion Detection Models

Models	Accuracy	Precision	Recall	F1-Score
LR	0.61	0.38	0.62	0.47
LDA	0.62	0.46	0.63	0.51
KNN	0.72	0.71	0.73	0.70
RF	0.71	0.69	0.72	0.67
CART	0.77	0.77	0.77	0.75
SVM	0.60	0.37	0.61	0.46
LSTM	0.68	0.64	0.68	0.63
CNN	0.79	0.80	0.78	0.78
BiLSTM	0.80	0.81	0.80	0.79
CNN-GRU	0.80	0.82	0.80	0.80
Proposed CNN-LSTM	0.81	0.85	0.81	0.78

The experimental results demonstrate that the proposed modified CNN-LSTM framework achieves superior intrusion detection performance compared with both conventional machine learning techniques and modern deep learning-based IDS models. The proposed framework attains the highest precision value of 0.85 and improved overall classification accuracy due to its efficient spatial feature extraction using CNN layers and temporal sequence learning using stacked LSTM layers. The comparative analysis further confirms that traditional machine learning algorithms exhibit reduced performance when processing high-dimensional and heterogeneous IoT traffic data. Although modern deep learning models such as CNN, BiLSTM, and CNN-GRU achieve competitive results, the proposed CNN-LSTM model demonstrates improved robustness and classification capability for real-time IoT intrusion detection environments. The inclusion of both traditional machine learning and modern deep learning-based IDS baselines improves the methodological validity and reliability of the comparative evaluation performed in this study.

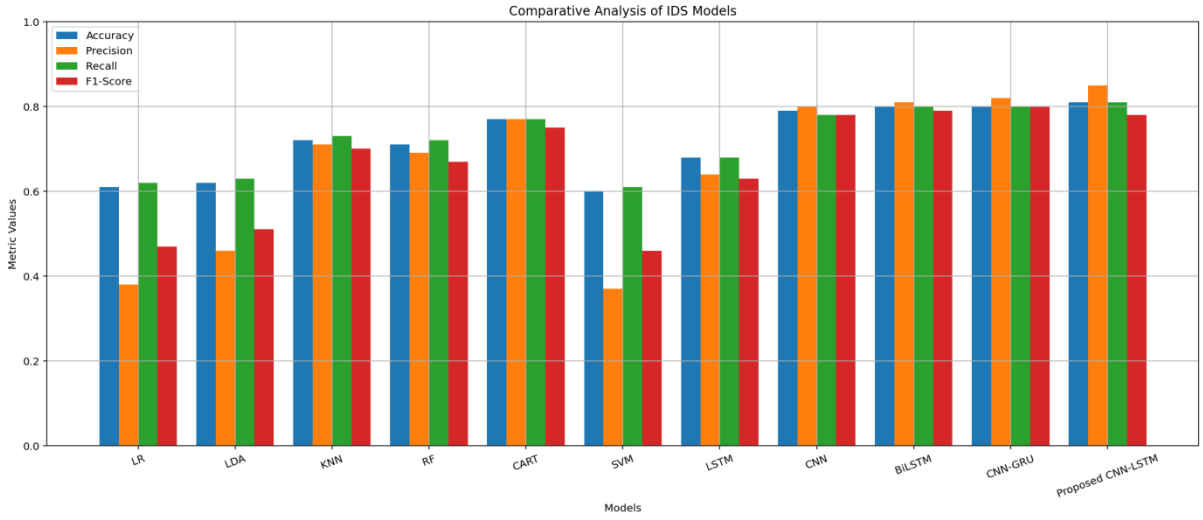


Fig. 14. Comparative Analysis of IDS models

Fig. 14. presents a comparative analysis of eleven different Intrusion Detection System (IDS) models evaluated across four key performance metrics: Accuracy, Precision, Recall, and F1-Score. Traditional machine learning models such as Logistic Regression (LR), Linear Discriminant Analysis (LDA), and Support Vector Machine (SVM) exhibit the lowest overall performance, particularly struggling with precision and F1-scores that fall below 0.50. In contrast,

ensemble and distance-based approaches like KNN, Random Forest (RF), and CART demonstrate moderate improvements, with CART maintaining balanced metrics near 0.75. Deep learning architectures (LSTM, CNN, BiLSTM, and CNN-GRU) show a clear advantage, consistently achieving scores around or above the 0.80 threshold. Notably, the Proposed CNN-LSTM model delivers the overall best performance, achieving the highest precision score (around 0.85) while maintaining superior, well-balanced levels of accuracy and recall, proving its effectiveness over standard baseline models.



Fig. 15. Performance comparison of IDS models

Fig. 15. illustrates a comparative performance evaluation of eleven Intrusion Detection System (IDS) models across four core evaluation criteria: Accuracy, Precision, Recall, and F1-Score. Traditional machine learning classifiers like LR, LDA, and SVM exhibit the lowest overall performance, with distinctive drops in precision down to nearly 0.38, showing a severe imbalance across metrics. Non-linear and ensemble methods such as KNN, RF, and CART achieve moderate stability, with CART peaking among traditional methods by tightly clustering all four metrics near a balanced 0.77. Deep learning models (CNN, BiLSTM, CNN-GRU) show steady upward trends, stabilizing firmly around the 0.80 performance threshold. Ultimately, the Proposed CNN-LSTM model secures the peak overall performance, driven by a sharp upward trend in Precision reaching approximately 0.85, establishing it as the most effective and reliable framework in this evaluation.

4.4.1. Discussion on Experimental Consistency and Computational Performance

The experimental results obtained from different machine learning and deep learning models were carefully revalidated to ensure consistency and reliability. All models were evaluated using the same dataset split ratio, identical preprocessing operations, and uniform evaluation metrics under the same hardware and software environment to ensure fair comparative analysis. The comparatively lower precision values observed for certain traditional machine learning algorithms such as Logistic Regression (LR) and Support Vector Machine (SVM) are primarily caused by class overlap among IoT attack categories, imbalance in network traffic patterns, and the limited capability of conventional machine learning models to capture temporal dependencies and nonlinear traffic relationships present in dynamic IoT environments. In addition, the larger computational time observed for the SVM classifier is attributed to the high-dimensional nature and large-scale size of the IoT Device Network Logs dataset. Since SVM performs kernel-based optimization and multiclass classification, the computational complexity increases significantly when processing large-scale IoT traffic data containing heterogeneous traffic features and intrusion categories. Consequently, the SVM model demonstrates substantially higher training overhead compared with deep learning-based approaches optimized for sequential feature learning. The proposed modified CNN-LSTM framework demonstrates improved computational efficiency and stable intrusion classification performance due to the integration of convolutional feature extraction, stacked temporal learning, max-pooling operations, and dropout-based regularization. The obtained results confirm that the proposed framework is more suitable for real-time IoT intrusion detection applications than conventional machine learning-based IDS models. The experimental observations further indicate that deep learning-based IDS models exhibit superior scalability and computational adaptability compared with conventional machine learning algorithms when processing large-scale IoT network traffic datasets.

4.4.2. Limitations of the Proposed Framework

Despite achieving improved intrusion detection performance, the proposed CNN-LSTM framework has certain limitations that should be acknowledged. First, the model requires higher computational resources during training due to

the integration of convolutional and recurrent neural network layers, which may limit direct deployment on highly resource-constrained IoT edge devices. Although the framework demonstrates efficient inference performance, additional optimization techniques may be required for low-power embedded environments. Second, the experimental evaluation was conducted using a single IoT Device Network Logs dataset. Therefore, the generalization capability of the model across different IoT datasets, network architectures, and emerging attack scenarios requires further investigation. Third, as the volume of IoT traffic continues to grow, scalability becomes an important consideration for large-scale deployments involving millions of connected devices and real-time data streams. Future studies should investigate model compression, lightweight architectures, edge-based deployment strategies, and cross-dataset validation to further enhance scalability, computational efficiency, and generalization capability in practical IoT environments.

5. Conclusion and Future recommendations

The deployment of the CNN-LSTM model for real-time intrusion detection in IoT ecosystems results in a substantial encroachment in cybersecurity. As IoT devices become primary to various domains, conventional IDS has faced challenges such as device heterogeneity, restricted computational resources, and diverse communication protocols. The proposed modified CNN-LSTM model has tackled the issues by efficiently capturing spatial and temporal patterns in IoT data. The CNN layers surpass in extracting spatial features from raw network traffic, although the LSTM networks process temporal dependencies, permitting the model to detect evolving attacks with high accuracy and speed in contrast to conventional IDS techniques. The IoT Device Network Logs dataset is used in training and testing the model which demonstrates the real-world network activities in IoT environments, comprising varied device interactions and network traffic data, which permitted the model to detect various cyber-attacks such as DDoS, spoofing, and man-in-the-middle attacks. As a result, the proposed modified CNN-LSTM model has achieved a testing accuracy of 81%, precision of 85%, recall of 81%, and an F1-score of 78%, highlighting the superior performance in contrast to prevailing methodologies. The future work will focus on enhancing the model's architecture to decrease computational overhead for resource-constrained IoT devices. Furthermore, increasing the dataset to contain more attack situations and discovering progressive AI techniques such as FL (Federated Learning) and RL (Reinforcement Learning) will augment the model's performance.

All the Declarations and Statements

Author Contributions Statement

Sabeena S – Conceptualization, Methodology, and Supervision: Proposed research ideas, Constructed the overall framework, and supervised project execution.
Dr. Chitra S– Data Curation and Software Implementation: Handled data acquisition, dataset preprocessing, and implementing the research model.
Dr. Chitra S – Model Training, Validation, and Performance Evaluation: Led the model training process, validated results using standard metrics, and benchmarked performance against existing methods.
Dr. Chitra S – Formal Analysis, Visualization, and Statistical Analysis: Performed in-depth analysis of experimental results, prepared performance charts, and ensured the statistical robustness of the evaluation.
Sabeena S – Writing – Drafted the initial manuscript, contributed to the literature survey, and documented the technical background of the study.
Sabeena S – Writing – Review and Editing, and Project Management: Reviewed and edited the manuscript, ensured clarity and coherence, and helped coordinate project milestones and deadlines.

All authors have read and agreed to the published version of the manuscript.

Conflict of Interest Statement

The authors declare no conflicts of interest.

Funding Declaration

There is no funding support for this study.

Data Availability Statement

There is no research data outside the submitted manuscript file.

Ethical Declarations

Not applicable for this research

Acknowledgments
We sincerely thank the experts for their professional evaluation and valuable recommendations, which have contributed to improving the quality of the experiment and the reliability of its results.

Abbreviations
The following abbreviations are used in this manuscript:

IoT - Internet of Things.
IDS - Intrusion Detection Systems.
ID - Intrusion Detection.
NIDS - Network-based Intrusion Detection Systems.
HIDS - Host-based Intrusion Detection Systems.
AI - Artificial Intelligence.
ML - Machine Learning.
DL - Deep Learning.
CNN / CNNs - Convolutional Neural Networks.
LSTM - Long Short-Term Memory.
NNs - Neural Networks.
RNN - Recurrent Neural Networks.
DoS - Denial-of-Service.
DDoS - Distributed Denial-of-Service.
MitM - Man-in-the-Middle.

Appendix A/B/C…, with appendix title
Not applicable

References

[1] Z. Yang, X. Liu, T. Li, D. Wu, J. Wang, Y. Zhao, and H. Han, "A systematic literature review of methods and datasets for anomaly-based network intrusion detection," Computers & Security, vol. 116, Art. no. 102675, 2022.
[2] S. Muneer, U. Farooq, A. Athar, M. Ahsan Raza, T. M. Ghazal, and S. J. J. o. E. Sakib, "A critical review of artificial intelligence based approaches in intrusion detection: A comprehensive analysis," *Journal of Engineering* vol. 2024, no. 1, p. 3909173, 2024, doi: https://doi.org/10.1155/2024/3909173.
[3] M. Shafi, A. H. Lashkari, A. H. J. J. o. N. Roudsari, and S. Management, "Toward Generating a Large Scale Intrusion Detection Dataset and Intruders Behavioral Profiling Using Network and Transportation Layers Traffic Flow Analyzer (NTLFlowLyzer)," *Journal of Network Systems Management* vol. 33, no. 2, p. 44, 2025, doi: https://doi.org/10.1007/s10922-025-09917-0.
[4] E. Hallaji, R. Razavi-Far, M. J. C. Saif, and Security, "Expanding analytical capabilities in intrusion detection through ensemble-based multi-label classification," *Computers Security* vol. 139, p. 103730, 2024, doi: https://doi.org/10.1016/j.cose.2024.103730.
[5] H. Satilmiş, S. Akleylek, and Z. Y. J. I. A. Tok, "A systematic literature review on host-based intrusion detection systems," *Ieee Access* vol. 12, pp. 27237-27266, 2024, doi: 10.1109/ACCESS.2024.3367004.
[6] O. I. Falowo, M. Ozer, C. Li, and J. B. J. I. A. Abdo, "Evolving malware & ddos attacks: Decadal longitudinal study," *IEEE Access* 2024, doi: 10.1109/ACCESS.2024.3376682.
[7] B. Isong, O. Kgote, and A. J. E. Abu-Mahfouz, "Insights into Modern Intrusion Detection Strategies for Internet of Things Ecosystems," *Electronics* vol. 13, no. 12, p. 2370, 2024, doi: https://doi.org/10.3390/electronics13122370.
[8] K. K. BN, R. Balakrishna, and M. P. Rao, "AN ENSEMBLE COGNITIVE LEARNING-BASED INTRUSION DETECTION SYSTEM FOR SECURE IOT ENVIRONMENTS," doi: https://doie.org/10.0113/Jbse.2025967844.
[9] A. Zohourian, S. Dadkhah, H. Molyneaux, E. C. P. Neto, A. A. J. C. Ghorbani, and Security, "IoT-PRIDS: Leveraging packet representations for intrusion detection in IoT networks," *Computers Security* vol. 146, p. 104034, 2024, doi: https://doi.org/10.1016/j.cose.2024.104034.
[10] S. Remya, M. J. Pillai, C. Arjun, S. R. Subbareddy, and Y. J. I. A. yun Cho, "Enhancing security in llns using a hybrid trust-based intrusion detection system for rpl," *IEEE Access* 2024, doi: 10.1109/ACCESS.2024.3391918.
[11] S. Tabbassum, R. K. J. V. R. Pathak, and I. Hardware, "Effective data transmission through energy-efficient clus-tering and fuzzy-based IDS routing approach in WSNs," *Virtual Reality Intelligent Hardware* vol. 6, no. 1, pp. 1-16, 2024, doi: https://doi.org/10.1016/j.vrih.2022.10.002.
[12] A. I. Gide and A. A. J. B. J. o. I. o. T. Mu'azu, "A real-time intrusion detection system for dos/ddos attack classification in IoT networks using KNN-neural network hybrid technique," *Babylonian Journal of Internet of Things* vol. 2024, pp. 60-69, 2024, doi: https://doi.org/10.58496/BJIoT/2024/008.
[13] M. Thankappan, H. Rifà-Pous, and C. J. I. J. o. I. S. Garrigues, "A distributed and cooperative signature-based intrusion detection system framework for multi-channel man-in-the-middle attacks against protected Wi-Fi networks," *International Journal of Information Security vol. 23, no. 6, pp. 3527-3546, 2024*, doi: https://doi.org/10.1007/s10207-024-00899-9.
[14] H. Fereidouni, O. Fadeitcheva, M. J. S. Zalai, and Privacy, "IoT and man-in-the-middle attacks," *Security Privacy* vol. 8, no. 2, p. e70016, 2025, doi: https://doi.org/10.1002/spy2.70016.

[15] S. Javanmardi, M. Ghahramani, M. Shojafar, M. Alazab, A. M. J. C. Caruso, and Security, "M-RL: A mobility and impersonation-aware IDS for DDoS UDP flooding attacks in IoT-Fog networks," *Computers Security* vol. 140, p. 103778, 2024, doi: https://doi.org/10.1016/j.cose.2024.103778.
[16] S. Racherla, P. Sripathi, N. Faruqui, M. A. Kabir, M. Whaiduzzaman, and S. A. J. I. A. Shah, "Deep-IDS: A Real-Time Intrusion Detector for IoT Nodes Using Deep Learning," *IEEE Access* 2024, doi: 10.1109/ACCESS.2024.3396461.
[17] B. Sharma, L. Sharma, C. Lal, and S. J. E. S. w. A. Roy, "Explainable artificial intelligence for intrusion detection in IoT networks: A deep learning based approach," *Expert Systems with Applications* vol. 238, p. 121751, 2024, doi: 10.1016/j.eswa.2023.121751.
[18] J. Manokaran and G. J. I. A. Vairavel, "Dl-ads: Improved grey wolf optimization enabled ae-lstm technique for efficient network anomaly detection in internet of thing edge computing," *IEEE Access* 2024, doi: 10.1109/ACCESS.2024.3405628.
[19] B. Xu, L. Sun, X. Mao, R. Ding, and C. Liu, "IoT intrusion detection system based on machine learning," Electronics, vol. 12, no. 20, Art. no. 4289, 2023.
[20] E. Altulaihan, M. A. Almaiah, and A. J. S. Aljughaiman, "Anomaly detection IDS for detecting DoS attacks in IoT networks based on machine learning algorithms," *Sensors* vol. 24, no. 2, p. 713, 2024, doi: https://doi.org/10.3390/s24020713.
[21] M. Sarhan, S. Layeghy, N. Moustafa, M. Gallagher, M. J. D. C. Portmann, and Networks, "Feature extraction for machine learning-based intrusion detection in IoT networks," *Digital Communications Networks* vol. 10, no. 1, pp. 205-216, 2024, doi: https://doi.org/10.1016/j.dcan.2022.08.012.
[22] R. Kumar, M. J. J. o. N. Swarnkar, and C. Applications, "QuIDS: A Quantum Support Vector machine-based Intrusion Detection System for IoT networks," *Journal of Network Computer Applications* vol. 234, p. 104072, 2025, doi: https://doi.org/10.1016/j.jnca.2024.104072.
[23] H. Nandanwar and R. J. E. S. w. A. Katarya, "Deep learning enabled intrusion detection system for Industrial IOT environment," *Expert Systems with Applications* vol. 249, p. 123808, 2024, doi: 10.1016/j.eswa.2024.123808.
[24] R. Alasmari and A. A. J. I. A. Alhogail, "Protecting smart-home IoT devices from MQTT attacks: An empirical study of ML-based IDS," *IEEE Access* vol. 12, pp. 25993-26004, 2024, doi: 10.1109/ACCESS.2024.3367113.
[25] F. Wahab, A. Shah, I. Khan, B. Ali, M. J. C. Adnan, and E. Engineering, "An SDN-based Hybrid-DL-driven cognitive intrusion detection system for IoT ecosystem," *Computers Electrical Engineering* vol. 119, p. 109545, 2024, doi: https://doi.org/10.1016/j.compeleceng.2024.109545.
[26] M. Benmalek, A. J. D. S. Seddiki, and Management, "Particle swarm optimization-enhanced machine learning and deep learning techniques for Internet of Things intrusion detection," *Data Science Management* 2025, doi: https://doi.org/10.1016/j.dsm.2025.02.005.
[27] M. B. Musthafa *et al.*, "Optimizing IoT intrusion detection using balanced class distribution, feature selection, and ensemble machine learning techniques," *Sensors* vol. 24, no. 13, p. 4293, 2024, doi: https://doi.org/10.3390/s24134293.
[28] M. A. Qathrady *et al.*, "SACNN-IDS: A self-attention convolutional neural network for intrusion detection in industrial internet of things," *CAAI Transactions on Intelligence Technology* vol. 9, no. 6, pp. 1398-1411, 2024, doi: https://doi.org/10.1049/cit2.12352.
[29] L. Saraladeve *et al.*, "A Multiclass Attack Classification Framework for IoT Using Hybrid Deep Learning Model," *Journal of Cybersecurity Information Management* vol. 15, no. 1, 2025, doi: 10.54216/JCIM.150112.
[30] S. Abbas *et al.*, "Evaluating deep learning variants for cyber-attacks detection and multi-class classification in IoT networks," *PeerJ Computer Science* vol. 10, p. e1793, 2024, doi: 10.7717/peerj-cs.1793.
[31] A. Momand, S. U. Jan, and N. J. W. P. C. Ramzan, "ABCNN-IDS: attention-based convolutional neural network for intrusion detection in IoT networks," *Wireless Personal Communications* vol. 136, no. 4, pp. 1981-2003, 2024, doi: https://doi.org/10.1007/s11277-024-11260-7.
[32] C. Hazman, A. Guezzaz, S. Benkirane, M. J. T. S. Azrour, and Technology, "Enhanced IDS with deep learning for IoT-based smart cities security," *Tsinghua Science Technology* vol. 29, no. 4, pp. 929-947, 2024, doi: 10.26599/TST.2023.9010033.
[33] A. Rajak and R. J. I. J. o. I. T. Tripathi, "DL-SkLSTM approach for cyber security threats detection in 5G enabled IIoT," *International Journal of Information Technology* vol. 16, no. 1, pp. 13-20, 2024, doi: https://doi.org/10.1007/s41870-023-01651-7.
[34] S. Hizal, U. Cavusoglu, and D. J. I. o. T. Akgun, "A novel deep learning-based intrusion detection system for IoT DDoS security," *Internet of Things* vol. 28, p. 101336, 2024, doi: https://doi.org/10.1016/j.iot.2024.101336.
[35] B. Olanrewaju-George, B. J. C. S. Pranggono, and Applications, "Federated learning-based intrusion detection system for the internet of things using unsupervised and supervised deep learning models," *Cyber Security Applications* vol. 3, p. 100068, 2025, doi: https://doi.org/10.1016/j.csa.2024.100068.

Authors' Profiles



SABEENA S., Research Scholar, Department of Computer Science, Bishop Heber College, Bharathidasan University, Trichy 620017.



Chitra S., Ph.D. in Computer Science from Bharathidasan University in 2021. Prior to this, completed an M.Phil. in Computer Science from Kurinji College of Arts and Science in 2009, an MCA (Master of Computer Applications) from Bharathidasan University in 2002, and a B.Sc. in Mathematics from Srimathi Indira Gandhi College in 1999.
Work Experience: Over 17 years of teaching experience and additional industry experience in software development and administration. The professional career began as a Teacher at Sri Balaji Matriculation School, Tiruchirappalli**,** from 1 August 1999 to 31 May 2000**.** Subsequently, worked as a VB Programmer at RDS Solutions, Tiruchirappalli**,** from 3 January 2001 to 4 September 2002**,** followed by a role as Computer Staff at Jayaraman Chartered Accountant, Tiruchirappalli**,** from 5 September 2002 to 19 October 2004**.** Joined Srimad Andavan Arts and Science College, Tiruchirappalli**,** as an Assistant Professor on 17 January 2008 and served until 26 June 2023**.** Since 26 June 2023**,** working as an Assistant Professor in the Department of Computer Science at Bishop Heber College, Tiruchirappalli**,** contributing to teaching, research, curriculum development, and student mentoring.
Awards: Received several recognitions for academic and research excellence. In 2021, honored with the Rashtriya Nirman Ratna Award by The National Academy for Art Education**,** Pune, in recognition of outstanding contributions to education and research also received the Natramilasan Award at the State level from Tamil Sangam, Kulithalai**,** on 5 September 2021. In addition, granted an Indian Patent titled "IoT-Based Smart Public Food Distribution Management System to Control" (Patent No**.** 202141047733A**),** which was officially granted in October 2021**,** demonstrating innovative research contributions in the field of the Internet of Things (IoT) and smart public distribution systems.

How to cite this paper

Sabeena S., Chitra S., "Leveraging CNN-LSTM Networks for Real-Time Intrusion Detection and Classification in IoT", International Journal of Wireless and Microwave Technologies(IJWMT), Vol.16, No.5, pp. 377-397, 2026. DOI:10.5815/ijwmt.2026.05.23