A Novel Hybrid CNN-LSTM Framework for Robust DDoS Attack Detection and Classification

Ammar O. K. Al-Hasani[1,*], Islam R. Abdelmaksoud[1] , Amira Rezk[1]

[1]Dept. Of Information Systems, Faculty of Computers and Information, Mansoura University, Egypt

Emails: ammar11alhasany@gmail.com; islam-cis@mans.edu.eg; amira_rezk@mans.edu.eg

Abstract

Distributed Denial of Service (DDoS) assaults could be the most prevalent and impactful cybersecurity threats, aiming to disrupt networking services and stop legitimate users from getting access to the service. This paper presents a novel hybrid deep learning framework that employs Convolutional Neural Networks (CNN) for spatial feature extraction and Long Short-Term Memory (LSTM) networking to get long-term dependencies within network traffic. In the experiments on the CIC-DDoS-2019 database, a good classification performance of the proposed model is achieved with accurateness of 99.63%, preciseness of 99.24%, recall of 99.22%, F1 score of 99.22%, and Micro-AUC of 99.71%, surpassing traditional machine learning models such as LGBM, DNN, and standalone CNN and LSTM. In addition, Fuzzy Logic was implemented for risk management using three risk categories low, medium, and high .The findings uncovered that the proposed hybrid CNN-LSTM model gives the best evaluation metrics, despite the complexity and imbalance of the dataset classes. This is due to the capability of the model to combine special and non-permanent features out of the data. The proposed model also is proven to support integration in the whole system including time detection, blocking and alerting, such that it is considered a powerful system for network security.

Keywords: DDoS Attack Detection; Convolutional Neural Network (CNN); Long Short-Term Memory (LSTM); Hybrid Deep Learning Model; Cybersecurity Threat Classification

1. Introduction

Software Defined Networking (SDN) is a promising method in networking concepts, especially it differentiates between network management and data routing functionalities. This differentiation affects the role design for network components. Also, SDN considered a successful approach for network layers controlling to fulfil the requirements of new networking technologies [1].

One of the utmost effective cybering-attacks is (Dos), which mainly obstruct network services and devices functionalities. Furthermore, the more effective upgrade of DoS attacks is (DDoS), which has the ability to utilize different sources to concentrate the effect of the attack [2]. First introduced by Gligor in the context of operating systems [3, 4], DoS attacks have since become a widely exploited method. DDoS attacks, which involve synchronized actions from various systems to target a victim, protruded as one of the utmost security threats today. DDoS attacks often target a certain service with more traffic that it can handle, so that this service becomes unavailable for beneficiaries. Since DDoS attacks continue to be updated to be more powerful, the recent system architectures have to be enhanced in terms of performance, scalability and protection against DDoS attacks. Artificial Intelligence (AI) particularly machine learning (ML) arise to be the utmost solution for detecting and isolating these attacks. However, some challenges have appeared such as data high dimensionality with big network traffic.

The existence of the abovementioned challenges becomes more complicated when combined with the fact that this data includes normal and abnormal types or malicious, which make the task of detection and isolation become more difficult. To overcome these challenges Deep Learning (DL) has proven a strong capability in dealing with high dimensional data, unlike the traditional ML techniques with require dimensionality reduction and/or feature extraction techniques a mandatory preprocessing stage before model training. DL models such as (CNN) have the

DOI: https://doi.org/10.54216/JCIM.170103
Received: March 03, 2025 Revised: June 01, 2025 Accepted: July 05, 2025

ability to make automatic feature extraction and catch temporal features. On the other hand, (RNN) with its main types of neurons (LSTM), and (GRU) is more appropriate for sequence learning as they can store relevant information from the previous epochs during training as well as interconnections between the data features [5]. Hence, by combing the two types of networks the model can leverage the advantages of each layer's types.

For the risk management fuzzy logic was implemented for its adaptability especially in dynamic and evolving attack strategies. This can identify risks in network traffic, which enhances automatic decision-making, and gives better interpretation of the results.

DDoS attacks work especially with Internet of Things (IoT) devices and services, and can target different protocols including (TCP – UDP – DNS - HTTP) on multiple layers. The complexity of these attacks requires robust framework to identify, block and report about potential threats. The illustration of DDoS attacks types is defined in Fig. 1.

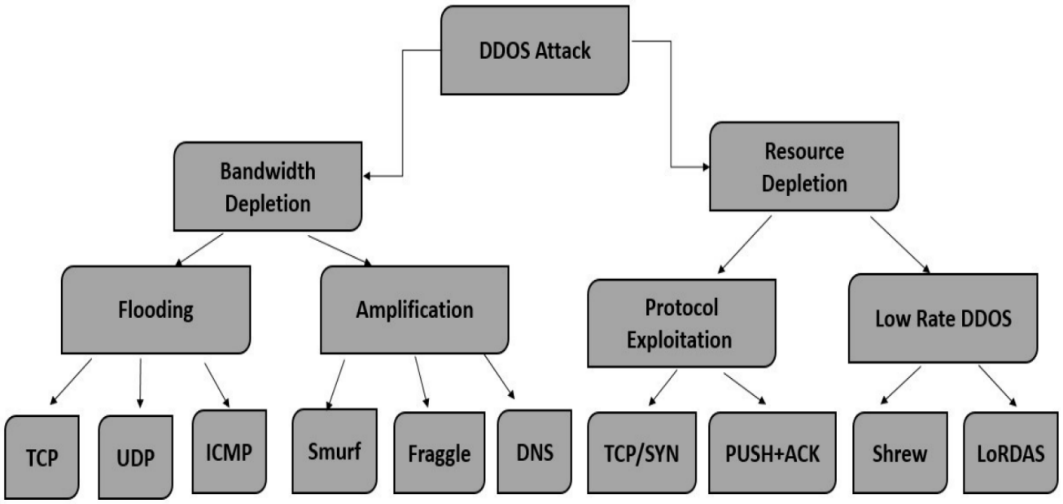


Figure 1. DDoS attacks types and Categories

The work proposed in this paper shall include DDoS detecting system centered upon deep learning techniques to eliminate the effect of DDoS attacks on SDN network environments. The main contributions are:

- Literature Survey and analysis of the main DDOS detection frameworks.
- Examining several feature-extracting techniques before training.
- Employing various machine/Deep learning models during training to enhance the DDoS detection system's efficiency.
- Development of novel hybrid dual channel model consisting of one-dimensional CNN for feature extraction followed by LSTM layers for temporal features capturing especially to overcome the challenges of the CICIDS 2019 dataset.
- Evaluating the optimized DL models by real databases and experiments to efficaciously estimate the effectiveness of detecting system.

2. Related Work

The growth in cyber-attacks especially (DoS) and (DDoS) attacks recently encourages researchers to study the effects of these attacks and build systems, including frameworks that have the ability to identify the attack and take relevant preventive actions to protect legitimate network users from losing service. This section discusses the literature work in the corresponding field, and gives brief insights to their results.

Concerning internet of things IoT devices, many researches have put spot on the network attack detection. The focus was on studying the challenges and effects related to DDoS attacks on IoT devices, especially the effect on the perception layer or the sensing layer. This layer, which is responsible for transducing other interfaces such as RFID, GPS, and wireless sensing, networks (WSN). This layer is vulnerable to many kinds of attacks and jamming. In the networking layer, which is in charge of handling how data interact between devices, the attack usually happens by sending too much traffic or what so called flooding, another way is by reflection or pretending to be

DOI: https://doi.org/10.54216/JCIM.170103
Received: March 03, 2025 Revised: June 01, 2025 Accepted: July 05, 2025

someone else. In the middle ware layer in which different parts of the system can communicate flooding and signature wrapping are common attacks. In the application layer where the user interacts with the system, the common attacks are reprogramming and path-based DoS attacks.

Recent researches study the capability of ML techniques to detect DDoS attacking, in [6, 7] the authors tried six ML models (Naïve Bayes (NB) – K-Nearest Neighbors (KNN) – Decision Trees (DT) – Support Vector Machines (SVMs) – Random Forests (RFs) – Logistic Regression (LR)), The works conducted on CICDDoS2019 dataset. However, the work applied to simple binary classification of normal and data with attack. Both Decision Trees and Random Forest achieved 99% accuracy. On the other hand, another study could achieve the same accuracy using CNN, but when applying to multiple attack types the accuracy could not achieve more than 87%.

The researchers in [8] studied several machine learning techniques for the same simple binary classification including (SVM, KNN, NB, RF, Adaboost, XGBoost). Among these models, Adaboost and XGBoost achieved the best accuracy of 100% according to the paper that may be considered to have overfitting, where XGBoost could be considered slightly better in terms of less training time and less resources are required. In [9] the models (RFs, KNN, DTs, and Artificial Neural Networks ANNs) were applied on the CICDDoS2019 dataset. ANNs achieved 99% accuracy of 99% for the binary classification as well. Another research [10] applied mathematical and machine learning models to CAIDA 2007 dataset; the results varied from 98% to 100% where Logistic Regression given better results. Another research applied machine learning models to CICDS2017 [11], where Random Forest made the maximal accurateness of 99.885% and low false alarm of 0.05%.

As an advancing in applying DL models, (RNNs) and their internal neuron types were applied for detecting DDoS attacks [12]. These variants are essential for time-based and sequence features capturing. This will enhance in analyzing network traffic patterns. Applying gradient descent with momentum for improving learning speed and stability of training, make the model achieved 99% accuracy. A more advanced version of RNN is using LSTM neurons and Bi-Directional LSTM (BiLSTM), which can read the internal data flow through training in forward and backward directions. When applying BiLSTM on the BoT-IoT achieved 99% accuracy. The common accuracy achieved while using RNNs Variants is typically 99% on CICDDoS2019 and CICIDS2017 datasets. However, Gated Recurrent Units (GRUs) offer faster processing time whie training when compared to other variants. This indicated that RNNs is very powerful in identifying patterns.

The next stage in the literature in combining ML and DL models to leverage the advantages of each algorithm through a hybrid model. For example, in [13], the authors combine RNNs a with extreme learning machines (ELMs) into one hybrid model, such that RNNs to handle sequenced data, and ELMs for classification to achieve 99% accuracy. On another research [14], the hybrid model was consisted of CNNs and BiLSTM, such that CNNs layers were used for feature extraction and BiLSTM for sequence feature capturing to achieve accuracy of 99.76% on CICIDS2017 dataset. In [15], that Kalman backpropagation was introduced for training a neural network that achieves 99% accuracy, showing that with less complexity backpropagation techniques ANNs still have the ability to correctly identifying attacks.

Though using ML models widely researched in DDoS attacking detection, the use of DL models in DDoS detection, particularly on the CICDDoS2019 dataset, remains to some extent unexplored. DL algorithms have surpassed traditional Machine Learning (ML) methods in terms of precision and accuracy, primarily because DL algorithms are able to process myriad of volumes of data and identify sophisticated patterns in networking trafficking and performance metrics.

In light of the above, the current research is seeking to use RNN models to conduct multi-class classification in the CICDDoS2019 dataset with the aim of improving DDoS detection and promoting network security research. In conclusion, the literature revealed that a wide variety of ML and DL models were used in DDoS attacking detection in the literature with outcomes of Logistic Regression (LR), LoR, Decision Trees (DT), SVM, Naive Bayes (NB), KNN, RF, XGBoost, and AdaBoost being tested on datasets such as CICIDS2017, KDD and CICDDOS2019.

The main approach in the previously discussed research papers showed that the mainstream of researching is only for identifying the data in to normal or had DDoS attack without classifying the types of the attacks, even with both CICDDoS2017 and CICDDoS2019 datasets that contain several types of DDoS attacks, except in only one paper that couldn't achieve 90% accuracy.

3. Proposed Methodology

This part is showings the methodology of the DDoS attack detecting framework that named Intrusion Detection System (IDS). The first stage is the preprocessing stage that mainly applying scaling, which transforms data into a uniformly normalized format. Then, multiple models are defined to be applied to the data including ML and DL models. The models under testing are Gradient Boosting, (DNN), (CNN), (RNN)-based models, and the final

DOI: https://doi.org/10.54216/JCIM.170103
Received: March 03, 2025 Revised: June 01, 2025 Accepted: July 05, 2025

proposed model is the hybrid approach of CNN-LSTM. The models are trained within two main supervised approaches: instant learning for (BM, DNN, CNN) and sequence learning for (RNN, CNN-LSTM) in a multi class classification models.

To obtain the best model's performance, adaptive architectures were introduced that have the ability to be optimized automatically by adjusting the count of neurons in the veiled layers for DL models. This approach includes predefinition of different ML and DL based architectures, and for the hybrid-based CNN-LSTM models.

A. Models Definition

1. LGBM (Light Gradient Boosting Machine)

Gradient Boosting has proven great performance especially in terms of speed in forecasting tasks with large datasets. LGBM was built by Microsoft using boosted trees that can bring accurate prediction for both regression and classification tasks. The speed of LGBM makes it powerful candidate for detecting DDoS attacks from large dataset.

2. DNN (Deep Neural Network)

AS the neural networks designed for supervised learning, which can capture the internal patterns of the dataset. This makes this type of models suitable for detecting DDoS attacks by capturing meaningful insights from the complex dataset.

3. CNN (Convolutional Neural Network):

CNN are the type of neural networking that called deep neural networks, It has proven great success in computer vision tasks through its capability in dealing with spatial data, which qualify it to be excellent stage for feature extraction. In our task, CNN will excel in detecting local patterns in network traffic.

4. RNN (Recurrent Neural Network):

For handling sequential data, RNN is the best approach as they have the ability to detect patterns rolling with time. This makes RNN well suited for the analysis of the temporal behavior of the network traffic while DDoS attacks are taking place. The standard LSTM architecture is shown in **Error! Reference source not found.**.

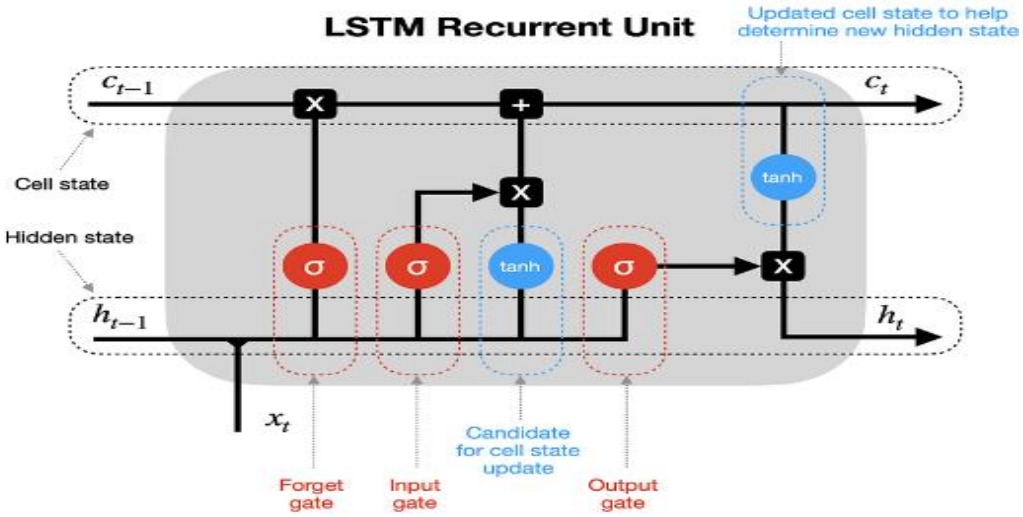


Figure 2. Standard LSTM Architecture

5. Hybrid CNN-LSTM

This model integrates both benefits of CNN and RNN-LSTM architectures in two parallel lines to handle multiclass classification problems. It has the capability to capture both spatiality and temporality of patterns and characteristics of the data. This hybrid method proved outstanding performance and efficiency in detecting complex DDoS attacks. The proposed architecture is in Fig. 3.

DOI: https://doi.org/10.54216/JCIM.170103
Received: March 03, 2025 Revised: June 01, 2025 Accepted: July 05, 2025

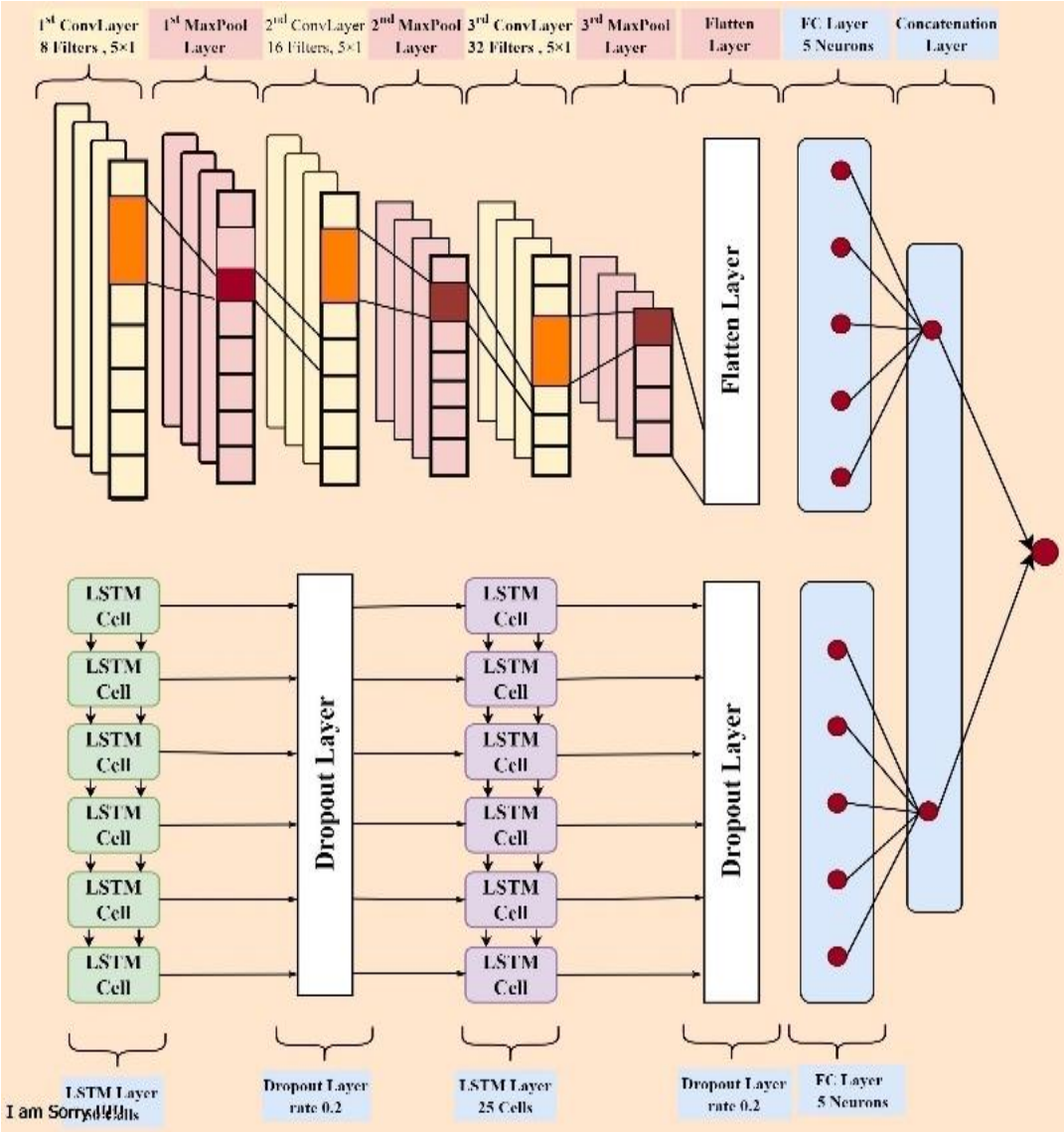


Figure 3. Proposed Hybrid CNN-LSTM Architecture

B. Proposed hybrid CNN-LSTM Classification Framework

The proposed hybrid model was designed for classifying seven categories that separated into six types of DDoS attacking and one class for typical traffic. The model was designed with two branches of CNN and RNN-LST respectively for ultimate feature extraction.

The CNN branch consists of three (one dimensional convolution layers [Conv1D]) with 16, 32, and 64 filters respectively each has ReLu activation function and same size padding to maintain the input shape. Batch normalization was used after each layer with MaxPooling for one dimensional data [MaxPooling1D] were applied after each convolutional layer to complete the feature extraction scheme. Then Dropout layer was added for regularization and to avoid overfitting. After the convolutional layers, the extracted data was fattened to a dense layer of ten units with tanh activation.

The RNN-LSTM branch has LSTM layer of 64 units, batch-normalization, and dropout layer for regularization. Then another LSTM layer of 32 units, batch-normalization, and dropout layer followed it. Then the output of this branch was passed to the dense layer of ten units with tanh activation.

The extracted features from both branches were concatenated together to combine both spatial and temporal features, then the final layer of 7 neurons as corresponding to 7 classes with SoftMax activation function to convert the output data into probabilities for classifying DDoS attacks. The loss function used in backpropagation is categorical cross entropy to calculate loss through probabilities. The hybrid model architecture is in Fig. 4.

DOI: https://doi.org/10.54216/JCIM.170103
Received: March 03, 2025 Revised: June 01, 2025 Accepted: July 05, 2025

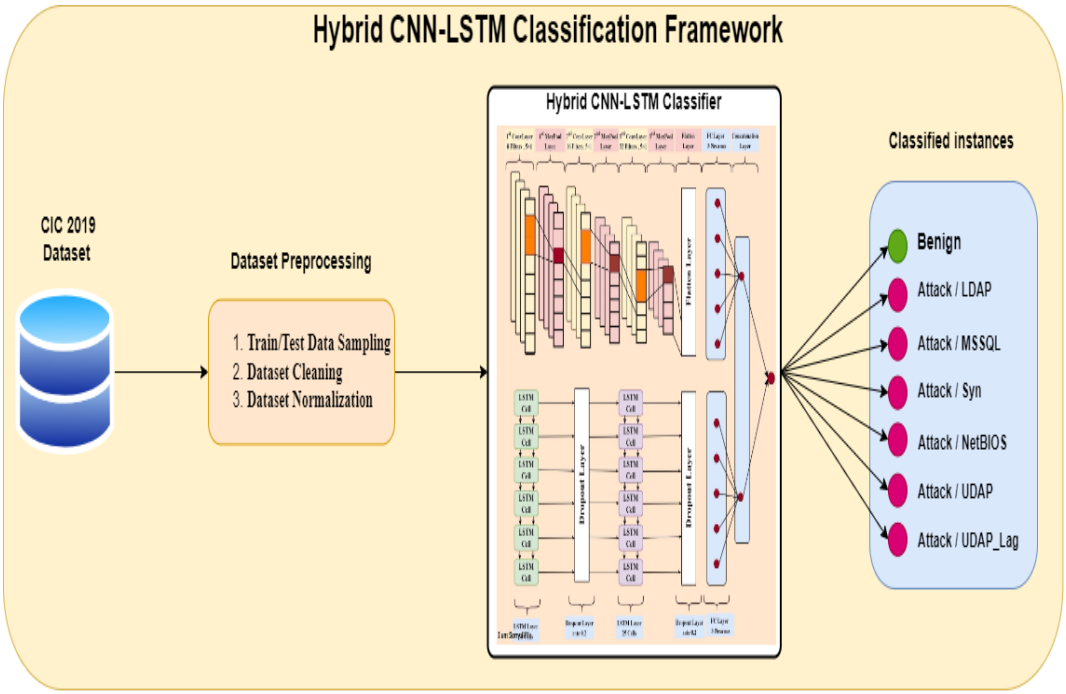


Figure 4. Proposed Hybrid CNN-LSTM classification Framework

C. Dataset

1. Dataset Description

The CIC-DDoS-2019 dataset comprises 18 files, which are organized into separate training and testing folders. Each file includes 87 feature columns: the first 80 columns represent features extracted using Cl-CFlowMeter, while the remaining 7 columns pertain to socket-related attributes, such as Flow ID, Source IP, Source Port, Destination IP, Destination Port, Protocol, and Timestamp. This dataset encompasses 18 distinct types of attacks, including DrDoSLDAR, DrDoS-MSSQL, DrDoS-NetBIOS, DrDoSNMP, DrDoS-SSDP, DrDoS-UDP, UDP-lag, WebDDoS, Syn, TFTR, DrDoS-DNS, DrDoS-NTR, Portmap, NetBIOS, LDAP, MSSQL, UDP, and UDPLag. Notably, over 99% of the data is classified as malicious traffic, as outlined in the dataset overview Table 1.

Table 1: CIC-DDoS-2019 dataset rows describing

Traffic type	Rows	
	Number	Ratio %
Normal	113,828	0.16
Malicious	70,313,809	99.84

For training (ML) and (DL) models to detect and mitigate (DDoS) attacking, the CICIDS 2019 dataset is often used. This dataset provides 87 traffic features, each important for analyzing network behavior. Key features of the dataset include sourcing and destination ports, flowing duration, and the rate of packets and bytes per second in both back and forth sides. It also monitors packet lengths (maximal, minimal, mean, and standard deviation), flow inter-arrival times, push and URG flags, header lengths, packeting rates, downloading/uploading ratios, bulk rates, subflows, and active time prior to becoming idle. These comprehensive features enable a detailed analysis of network traffic, making the dataset highly suitable for training models aimed at detecting and mitigating DDoS attacks.

As Fig. 5 uncovers, the testbed consists of two completely disconnected networking. Not like the former datasets, in the Victim-Network, we probably make use of entirely and publically adopted and vital prerequisites like router firewall, switching, along with numerous versions of the commonly adopted systems of operation.

DOI: https://doi.org/10.54216/JCIM.170103
Received: March 03, 2025 Revised: June 01, 2025 Accepted: July 05, 2025

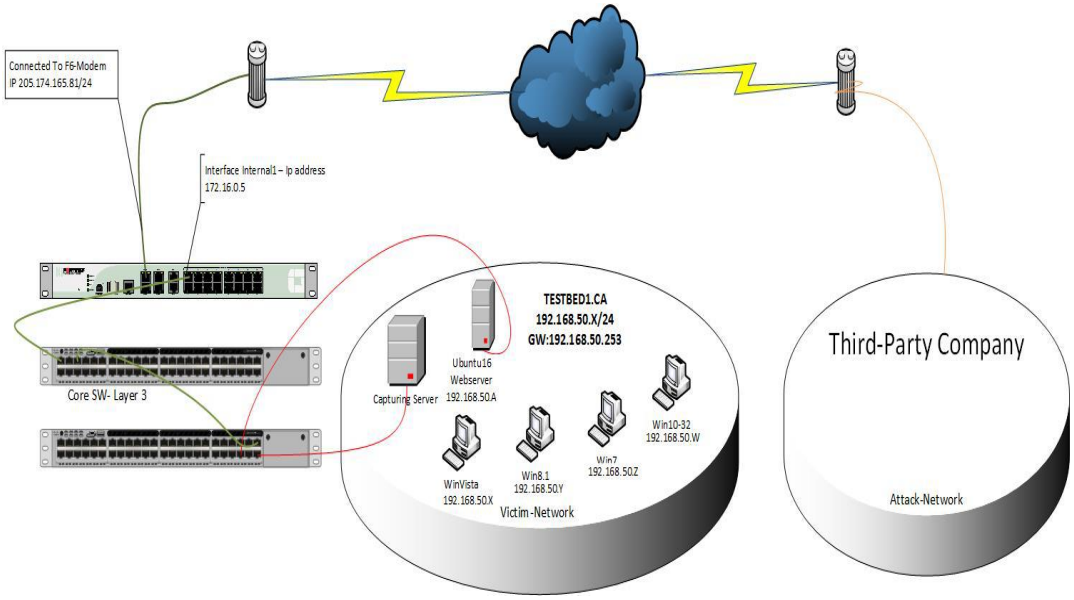


Figure 5. CIC2019 Testbed Architecture

1. Dataset Challenges

Class Imbalance

The dataset exhibits significant class imbalance: Benign traffic dominates, while specific attack types have far fewer samples. This imbalance makes it challenging to train machine-learning models that generalize well across all classes, especially rare attacks. Figure 1 the distribution of class over the dataset of training.

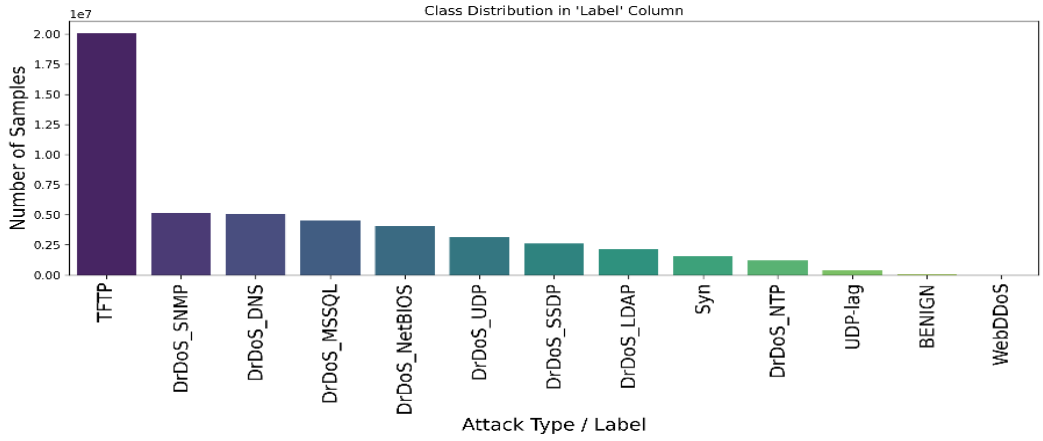


Figure 1. Class Distribution over training Dataset

High Dimensionality

The dataset has a myriad of features (87 Feature): Many of them are redundant or highly correlated. Featuring selection and reduction of dimensionality are necessary to avoid overfitting and improve model performance.

Heterogeneous Attack Types

The dataset encompasses a broad range of attacking forms, i.e. brute force, DoS/DDoS, infiltration, botnets, and web-based attacks. Each attack type is distinct and exhibits unique patterns, requiring models to learn from a variety of patterns.

Noise and Missing Values

The dataset contains realistic noise and missing values: some features include NaN, inf, or invalid entries. Addressing these inconsistencies is essential to ensure accurate preprocessing and reliable analysis.

DOI: https://doi.org/10.54216/JCIM.170103
Received: March 03, 2025 Revised: June 01, 2025 Accepted: July 05, 2025

Dataset Preprocessing

Data Sampling

We developed automated functions to load and preprocess large datasets efficiently. Each reads a CSV file, down-samples the majority class (non-BENIGN traffic) to balance the data, and merges it with the minority class (BENIGN traffic). The load file function processes the entire file at once, while load huge file handles larger files by reading them in chunks to prevent memory issues. Both maintain a specified balance ratio between classes, making them well suited for training models on imbalanced network traffic data.

Categorical Encoding:

We applied Label Encoding to transform categorical labels—such as attack types like "BENIGN" or "DDoS" into numerical values (e.g., 0, 1, 2) to ensure compatibility with machine learning models. The Label Encoder was used to map each unique category to a distinct integer, maintaining consistency between training and inference stages. The final Classes in the training and testing sets are: ['BENIGN': 0, 'LDAP': 1, 'MSSQL': 2, 'NetBIOS': 3, 'Syn': 4, 'UDP': 5, 'UDP-lag': 6]

Normalization:

The data normalization function uses Standard Scaler to standardize both training and test datasets, transforming them to have a mean of 0 and a standard deviation of 1. The scaler is close fitting to the data of training and then to the test set to ensure consistent scaling. By returning the normalized data, the function helps improve model performance and convergence, which is essential for effective machine learning training.

Feature Selection and Engineering

Both training and testing datasets underwent feature selection and engineering to boost model performance. String-based features like timestamps were converted into numeric hashes using the MD5 algorithm, ensuring consistent numerical encoding for text or categorical data. Unlimited values were substituted with NaN to allow safe numerical operations. Columns such as Flowing Packets/s and Flowing Bytes/s were converted to numeric types, with invalid entries coerced to NaN and filled with 0.

To analysis different types of DDOS attacks with original class distribution, a bar plot visualization was conducted to check classes' composition. DrDoS_DNS and TFTP DDoS attacks and their corresponding data rows were filtered for further data refinement. Aldo attack names were trimmed such as "DrDoS_" in order to simplify the class's names for further encoding.

Furthermore, the datetime column that was named Timestamp was splitted into two separate columns date and time. The time column was shown to the level of milliseconds, and then hour values were numerically encoded. Unnecessary columns including Source IP, Destination IP, Flow ID, Similar HTTP, and Unnamed: 0 were dropped to reduce dimensionality. Finally, the Label column was label-encoded to convert categorical attack names into numeric values, which were then stored back into the same column

4. Fuzzy Logic-Based Risk Scoring System for Network Security

Under the uncertainty of the data, Fuzzy Logic considered a powerful technique especially for decision-making [15], this approach can handle the challenges of the data, such as incompleteness, impreciseness, and complexity. So that Fuzzy Logic was applied to assess the risk of both key features flow, packets and SYN flag count.

The membership functions of mapping the raw data into the fuzzy sets. Then, flow packets were classified within three categories of risk including low, medium, and high. This is performed via triangular membership functions that based on three quartiles (25%, 50%, and 75%). The same was performed for SYN flag count with the similar approach.

The three categories then used as an input to make fuzzy rules. These fuzzy rules considered the relations between the inputs and the outputs, such that if both Flow Packets and SYN Flag are high in risk then the output is high and so on.

The dataset was sampled for both of 10,000 entries with high variation in Flow Packets/s (mean ~799,917) and mostly zero SYN flag counts. The results showed that the risk scores range from 11.00 to 88.58, with a mean of 50.87, reflecting moderate risk across the dataset. The risk class distribution is relatively balanced, with 3,629 high-risk, 3,344 low-risk, and 3,027 medium-risk instances, which indicated the diversity of network risk levels. The membership function curves showed the fuzzy logic effectiveness. Triangular or trapezoidal curves were used for visualizing the membership of the input features (Flow Packets/s and SYN Flag Count) Figure 1 and the output (Risk) Figure 1.

DOI: https://doi.org/10.54216/JCIM.170103
Received: March 03, 2025 Revised: June 01, 2025 Accepted: July 05, 2025

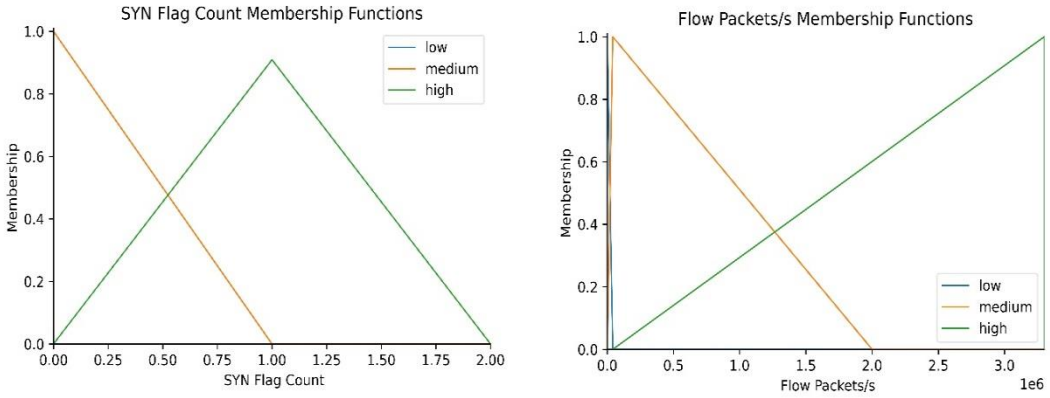


Figure 7. SYN Flag and Flow Packets Membership Functions

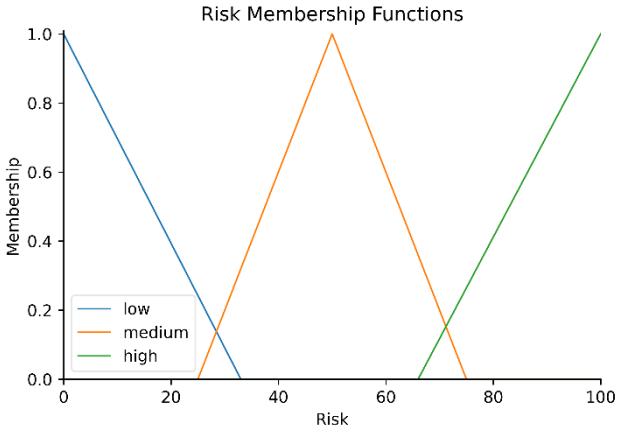


Figure 8. Risk Membership Functions for both selected Features

5. Experiments and Evaluation

A. Experimental Setup

The models in the current research were developed with the use of the Python programming language and the experiment was performed using a Jupyter Notebook. The DL models were optimized by multiple libraries such as pandas, matplotlib, scikit-learn, Keras, and scipy. The model was run on a machine with a dedicated Nvidia 1050Ti graphics card and 4 GB of memory and the training of the model lasted about 2 hours

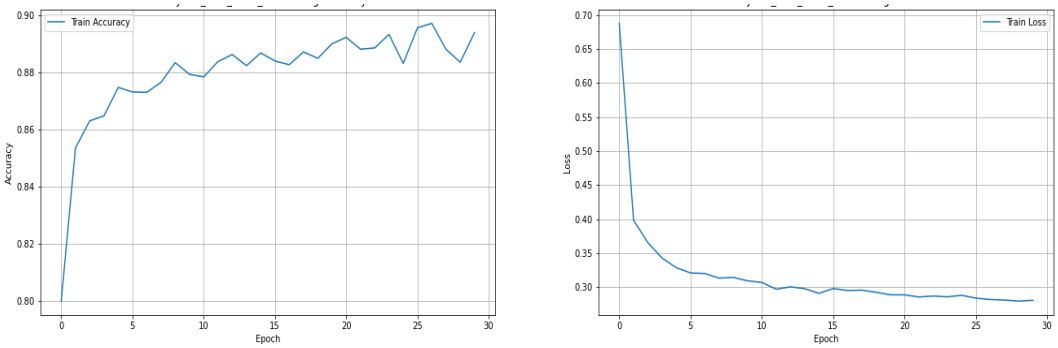
B. Experiments

In this part, the experimental evaluation is presented of the proposed methodology; the goal is to emphasize the results obtained from the data-preprocessing phase and the accuracy of the model. Specifically, for each (DL) model as well as its varieties, the error metrics on the dataset were reported. Additionally, the corresponding number of parameters was illustrated, which were determined by automatically adjusting the count of neurons in the nonseen layers.

1. Model training

The model is trained using a set number of epochs and batch size. By default, it runs for 30 epochs, processing the full dataset each time. The batch size is set to 16, meaning the model updates its weights after every 16 samples. During training, the Adam optimizer is used with predetermined learning ratios, and the loss function are scarce categorical cross-entropy, that are close-fit for multi-class classification. This training process allows the model to gradually learn from the data across multiple iterations, both training loss and training accurateness curves in Figure .

DOI: https://doi.org/10.54216/JCIM.170103
Received: March 03, 2025 Revised: June 01, 2025 Accepted: July 05, 2025

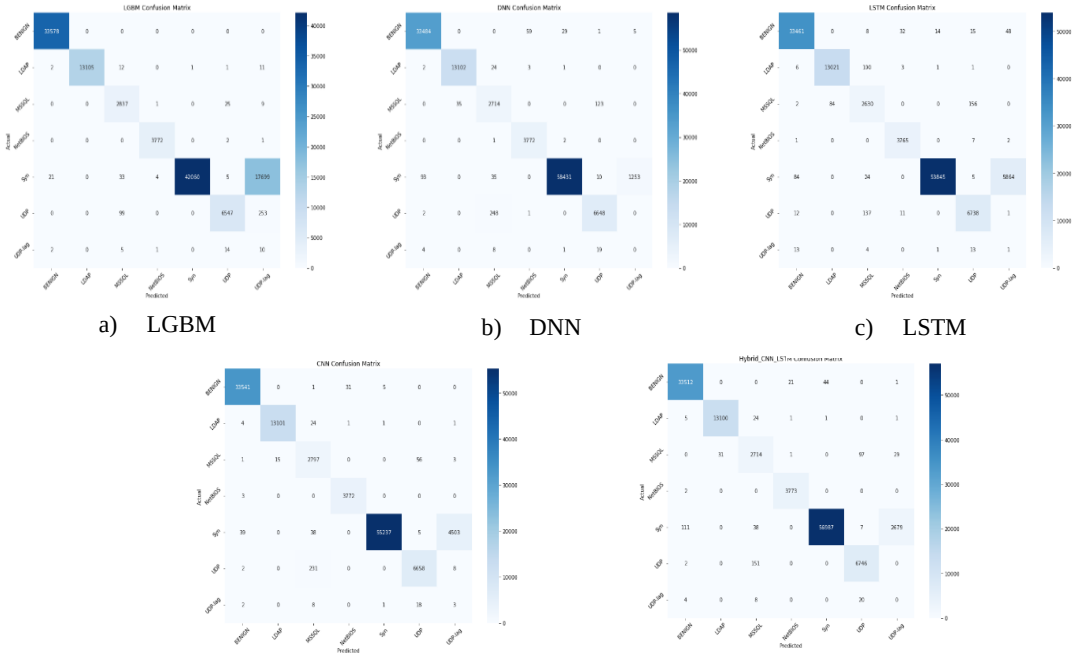


a) Curve of Training Accuracy b) Curve of Training Loss

Figure 9. Training Curves for Hybrid CNN-LSTM Mode

2. Performance metrics

The confusion matrix is efficaciously adopted to meticulously estimate the classification model's performance, as illustrated in Fig. 10 [13]. It uncovers the counts of true positives (TP), false positives (FP), true negatives (TN), and false negatives (FN) for a given set of predictions. Different models' confusion matrices are mentioned in **Error! Reference source not found.**.



a) LGBM b) DNN c) LSTM

d) CNN e) Hybrid CNN-

Figure 10. Confusion matrices for the proposed LGBM, DNN, LSTM, CNN, and Hybrid CNN-LSTM models, by Multi-classification on the CIC2019 Dataset.

These values are used to compute important evaluation metrics like accurateness, preciseness, recalling, and the F1 score. Precision reflects the percentage of positive predictions that are accurate, while recall indicates how many of the actual positive instances the model correctly identified. The F1 score combines precision and recall into a single measure using their harmonic mean, making it especially valuable for imbalanced datasets where one class dominates the other. In such cases, accuracy alone can be misleading. Maximal accurateness could result simply out of guessing the majority class and neglecting the minority class. Below are the equations for these evaluation metrics.

DOI: https://doi.org/10.54216/JCIM.170103
Received: March 03, 2025 Revised: June 01, 2025 Accepted: July 05, 2025

$$Precesion = \frac{True\ Positive\ (TP)}{True\ Positive\ (TP) + False\ Positive\ (FP)}$$

$$Recall = \frac{True\ Positive\ (TP)}{True\ Positive\ (TP) + False\ Negative\ (FN)}$$

$$F1_{Score} = 2 \times \frac{Precesion \times Recall}{Precesion + Recall}$$

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

3. Results

In this part, it uncovers the evaluation findings of the proposed models using the CIC-DDoS-2019 dataset, which finally consists of 7 labels after dataset sampling. The models were efficaciously trained and definitely tested with the available training and testing datasets the performance of the models was evaluated based on accuracy, precision, recall, F1 score, and micro-AUC. **Error! Reference source not found.** uncovers the metrics of performance for the CIC-DDoS-2019 dataset in the multi-label classification setup. The proposed Hybrid CNN-LSTM model delivers the optimal outcomes, achieving 99.22% accurateness, 99.24% preciseness, 99.22% recalling, an F1 score of 99.22%, and a micro-AUC of 99.71%. Other models—such as LGBM, DNN, LSTM, and CNN—also perform well, though they show slightly lower accuracy and other metric values. Among them, the LGBM model has the lowest overall performance, with 84.85% accuracy, 99.79% precision, 84.85% recall, and an F1 score of 91.02%.

Table 2: Multi-class classification results of the proposed Model

Method	**Accurateness**	**Preciseness**	**Recalling**	**F1 Score**	**Micro-AUC**
LGBM	0.848464	0.997861	0.848464	0.910195	0.987296
DNN	0.929298	0.992648	0.929298	0.958810	0.996856
LSTM	0.949446	**0.992552**	0.949446	0.969985	0.989912
CNN	0.954084	0.992032	0.954084	0.972162	0.994052
Hybrid CNN_LSTM (Proposed)	**0.99632**	0.992443	**0.992232**	**0.992219**	**0.997143**

In multi-label classification, although the proposed models perform well, they face challenges due to the complexity of the CIC-DDoS-2019 dataset. This dataset contains extra classes and is less balanced compared to other datasets. It also includes classes with very few samples and attack types that are semantically similar but labelled differently, making accurate classification more difficult. These factors lead to a drop in performance when compared to simpler datasets. Nevertheless, the Hybrid CNN-LSTM model still achieves the best results among all tested models, as shown in Fig. 11, demonstrating its efficiency in investigating as well as categorizing DDoS attacking across the CIC-DDoS-2019 dataset.

DOI: https://doi.org/10.54216/JCIM.170103
Received: March 03, 2025 Revised: June 01, 2025 Accepted: July 05, 2025

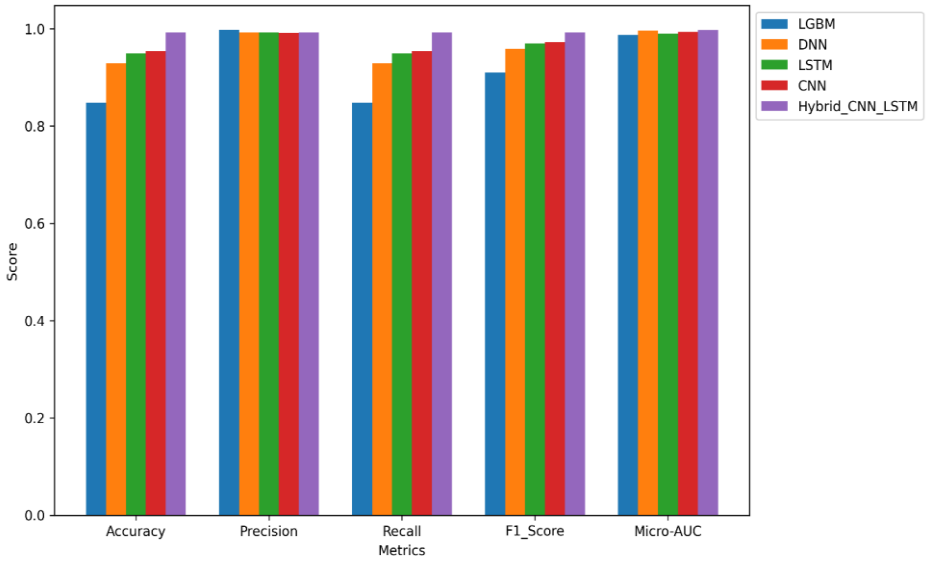


Figure 11. Models Performance Evaluation

The results especially of the proposed approach showed that it could outperform the other approaches in the literature presented in Table 3with wider variants of DDoS attacks classes.

Table 3: CIC-DDoS-2019 dataset literature summary

Technique Used	Number of Classes	Metrics with Values
Random Forest, Light Gradient Boosting, CatBoost, CNN	2	Accuracy: 99.99740% (RF), 99.99346% (Light GB), 99.98392% (CatBoost), 98.29388%
ID3, RF, Naive Bayes, Logistic Regression	2	F1 score: 0.69 (ID3), 0.62 (RF), 0.05 (Naive Bayes), 0.04 (Logistic Regression)
Bidirectional LSTM	2	DDoS: Accuracy 97.57%, Live detection 96%; XSS/SQL: Accuracy 89.34%, Live detection 90%
LSTM	2	TPR 100% for modern DDoS, >90% for attacking packets; FPR <3%; TFOR avg 0.344765%;
CNN	2	F1 Score: 0.9824; Accuracy: 0.9883
LR, KNN, RF, SVM, NB, DT, AdaBoost, XGBoost with PCA	4	RF: Accuracy 98.80%, AUC 0.99

In ML and DL models, measuring model performance is crucial step to ensure the validity of the model to be arrayed in real systems, especially in multiclass classification tasks. The AUC-ROC (area under the curve for receiving operational characteristics) plots true positive rates countering false positive rates, which indicates the behaviour of the model in terms of sensitivity and specificity. The higher the value of AUC the higher the model efficiency and vice versa however, an AUC around 0.5 indicates that the model failed to learn any meaningful patterns. The proposed hybrid CNN-LSTM model achieved a micro-AUC of 99.71% that reflects the rigidity of the model and its potentiality to differentiate between different types of classes of DDoS attacking and typical traffic. This is due to the advanced hybrid approach of feature extraction within the model's two branches. In addition, the model could handle the complex nature of the CIC-DDoS-2019 dataset, unlike other models in the literature that avoided dealing with more than 4 classes at maximum. The AUC-ROC results in Fig. 12.

DOI: https://doi.org/10.54216/JCIM.170103
Received: March 03, 2025 Revised: June 01, 2025 Accepted: July 05, 2025

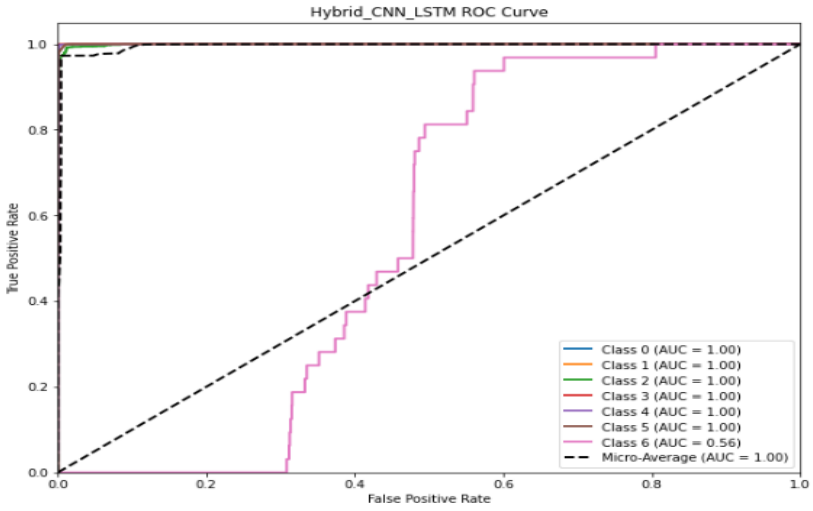


Figure 12. AUC-ROC Curve for Hybrid CNN-LSTM model over Multi-Classes problem

6. Conclusion

This paper proposed hybrid dual channel one dimensional CNN and RNN with LTM, STM for both spatial and temporal featuring extraction respectively. The proposed approach was deployed for DDoS attack detection and categorizing of normal and various types of DDoS attack. The recommended model was trained and definitely assessed on CIC-DDoS-2019 dataset, with multi label classification approach.

The evaluation metrics for proposed model were compared with multiple ML and DL models like LGBM, LSTM, and CNN. The metrics of assessment showed that proposed hybrid CNN-LSTM model could outperform the other models, with achieving accurateness of 99.63%, preciseness of 99.24%, recall of 99.22%, F1 score of 99.22%, and a Micro-AUC of 99.71%, in addition to highlighting the AUC-ROC curve, which indicates the potential stability and effectiveness of the model. The findings uncovered the efficacy of the recommended model in detecting and classifying DDoS attacking for various types, unlike the researches in the literature on the same dataset.

The feature extraction stage through a dual channel is considered the core strength of the proposed model that enables the model in the training process the features to conduct a robust solution that could be deployed in real-life network systems. The real-life implementation, which has the potentiality to make automatic response in real-time for DDoS attacks and take relevant actions such as IP blocking and E-mail alerting, enhances security against DDoS attacks in operational network environments.

The fuzzy logic allowed more rigid approach for decision making, such that the system can handle changes in traffic characteristics the might be indicating a possible DDOS attack. In addition, it presented an adaptive mechanism that can identify attack patterns.

Finally, this framework contributes the great growth in deep-learning based approaches for detecting, classification, as well as migration of DDoS attacking, and enhances the overall network security against potential threats.

References

[1] N. Faujdar, A. Sinha, H. Sharma, and E. Verma, “Network security in Software defined Networks (SDN),” in *2020 International Conference on Smart Technologies in Computing, Electrical and Electronics (ICSTCEE)*, 2020, pp. 377–380.

[2] M. Iqbal, F. Iqbal, F. Mohsin, M. Rizwan, and F. Ahmad, “Security issues in software defined networking (SDN): risks, challenges and potential solutions,” *International Journal of Advanced Computer Science and Applications*, vol. 10, no. 10, pp. 298–303, 2019.

[3] K. Lakshminarayanan, D. Adkins, A. Perrig, and I. Stoica, “Taming IP packet flooding attacks,” *ACM SIGCOMM Computer Communication Review*, vol. 34, no. 1, pp. 45–50, 2004.

[4] V. D. Gligor, “A note on denial-of-service in operating systems,” *IEEE Transactions on Software Engineering*, no. 3, pp. 320–324, 1984.

DOI: https://doi.org/10.54216/JCIM.170103
Received: March 03, 2025 Revised: June 01, 2025 Accepted: July 05, 2025

[5] J. Saikam and K. Ch, "EESNN: Hybrid Deep Learning Empowered Spatial–Temporal Features for Network Intrusion Detection System," *IEEE Access*, vol. 12, pp. 15930–15945, 2024.

[6] R. J. Alzahrani and A. Alzahrani, "Security analysis of DDoS attacks using machine learning algorithms in networks traffic," *Electronics*, vol. 10, no. 23, p. 2919, 2021.

[7] T. Dhamor, S. Bhat, and S. Thenmalar, "Dynamic approaches for detection of DDoS threats using machine learning," *Annals of the Romanian Society for Cell Biology*, vol. 25, no. 4, pp. 13663–13673, 2021.

[8] A. Seifousadati, S. Ghasemshirazi, and M. Fathian, "A Machine Learning approach for DDoS detection on IoT devices," *arXiv preprint arXiv: 2110.14911*, 2021.

[9] R. Amrish, K. Bavapriyan, V. Gopinaath, A. Jawahar, and C. V. Kumar, "DDoS detection using machine learning techniques," *Journal of IoT in Social, Mobile, Analytics, and Cloud*, vol. 4, no. 1, pp. 24–32, 2022.

[10] K. Kumari and M. Mrunalini, "Detecting Denial of Service attacks using machine learning algorithms," *Journal of Big Data*, vol. 9, no. 1, p. 56, 2022.

[11] C. M. Nalayini and J. Katiravan, "Detection of DDoS attack using machine learning algorithms," *Available at SSRN 4173187*, vol. 9, no. 7, 2022.

[12] R. Qamar, B. Zardari, A. Arain, F. Khoso, and A. Jokhio, "Detecting distributed denial of service attacks using recurrent neural network," *Psychology*, vol. 2022, p. 1, 2022.

[13] I. Ullah and Q. H. Mahmoud, "Design and development of RNN anomaly detection model for IoT networks," *IEEE Access*, vol. 10, pp. 62722–62750, 2022.

[14] K. Saurabh et al., "LBDMIDS: LSTM based deep learning model for intrusion detection systems for IoT networks," in *2022 IEEE World AI IoT Congress (AIIoT)*, 2022, pp. 753–759.

[15] L. A. Zadeh, "Fuzzy sets," *Information and Control*, vol. 8, no. 3, pp. 338–353, 1965.

[16] R. Qamar, "Gradient techniques to predict distributed denial-of-service attack," *Iraqi Journal for Computer Science and Mathematics*, vol. 3, no. 2, pp. 55–71, 2022.

DOI: https://doi.org/10.54216/JCIM.170103
Received: March 03, 2025 Revised: June 01, 2025 Accepted: July 05, 2025