Baltic J. Modern Computing, Vol. 14 (2026), No. 3, pp. 517–537
https://doi.org/10.22364/bjmc.2026.14.3.02

# A Deep Learning Model for Distributed Denial of Service Attack Detection

Khadidja CHAOUI[1], Farida BOUACHI[2]

[1] Department of Mathematics and Computer Science, Faculty of Science and Technology, University of Ain Temouchent- Belhadj Bouchaib, 46000 Ain Temouchent, Algeria
[2] Faculty of Computer Science, RIIMA Laboratory, University of Science and Technology Houari Boumediene, BP 32 El Alia 16111, Bab Ezzouar, Algiers, Algeria

khadidja.chaoui@univ-temouchent.edu.dz, fbouachi@usthb.dz

ORCID 0000-0001-6877-6858, ORCID 0009-0008-9607-8860

**Abstract.** Denial of Service attacks aim to prevent legitimate users from accessing targeted services. Attackers achieve this by flooding the target system with packets, overwhelming its resources and disrupting service availability. In this paper, we propose a hybrid deep learning model combining CNN, LSTM, and DNN. Evaluated on the CIC-DDoS-2019 dataset, the model achieves 0.9996 accuracy for binary classification and 0.97 for multiclass classification, demonstrating its ability to distinguish legitimate traffic from attacks and identify different types of DDoS attacks, making it a promising solution for real-time network intrusion detection systems.

**Keywords:** DoS, DDoS, Classification, Deep Learning

## 1 Introduction

Currently, Internet-based services have experienced rapid expansion across various domains, particularly in healthcare, education, and communication. Despite significant efforts to secure these systems, cybercrime continues to pose a serious threat, compromising the confidentiality, integrity, and availability of digital services. Among the wide range of cyber threats are Denial of Service (DoS) attacks, malware, zero-day exploits, data breaches, social engineering, and phishing attacks. The frequency and sophistication of these threats have grown exponentially over the past decade, creating major challenges for the security and resilience of modern information systems Ahsan et al. (2022).

In 2020, experts identified approximately 1.56 million more Distributed Denial of Service (DDoS) attacks than in 2019, corresponding to an average increase of 130,000 attacks per month. The year 2021 also experienced a high level of DDoS activity; although the number of attacks recorded during the first half of 2021 was lower than in

2020, it still reflected a sustained threat landscape. This trend continued into 2022, with attack volumes remaining elevated. During the first half of that year alone, experts identified 61,019,888 DDoS attacks originating from 190 different countries. In 2023, the number of DoS and DDoS attacks continued to rise, with more than 70 million attacks reported worldwide. Throughout 2025, attack levels remained similarly high, confirming a concerning trend of sustained growth in the prevalence of these threats Khadka et al. (2025); De Neira et al. (2023).

DDoS attacks are designed to prevent legitimate users from accessing targeted services Nuiaa et al. (2021); Ussatova et al. (2022); Al-Hadhrami and Hussain (2021). Due to the severity of the problem, numerous defence mechanisms have been proposed Sinha (2025); Reddy (2024); Li et al. (2023); Alatawi (2021), both at the network and application levels, to combat these attacks. This has led to a considerable number of studies focusing on the various mechanisms capable of detecting this type of attack, such as artificial intelligence Bravo and Mauricio (2018); Al-Shareeda et al. (2023); Banitalebi Dehkordi et al. (2021); M Nalayini and Katiravan (2022).

Most often, artificial intelligence is used in attack detection because of its effectiveness in processing large amounts of data, its ability to learn complex relationships, identify anomalies, and automate responses to security threats Mittal et al. (2023b); Eliyan and Di Pietro (2021); Adedeji et al. (2023); Mughal (2018).

Based on these concerns, we propose a hybrid deep learning model that combines convolutional neural networks (CNN), long short-term memory networks (LSTM), and deep neural networks (DNN). The CNNs extract local features from traffic segments, the RNNs model the sequential nature of network flows, and the LSTMs capture long-term dependencies. The model is trained and evaluated using the CIC-DDoS-2019 dataset. The model achieves 0.9996 accuracy for binary classification and 0.97 for multiclass classification, demonstrating its ability to distinguish legitimate traffic from attacks and identify different types of DDoS attacks, making it a promising solution for real-time network intrusion detection systems.

The rest of this document is organized as follows. Section 2 provides a synthesis of related work. Section 3 describes our proposed approach. Section 4 presents experimental results and comparisons. Section 5 discusses real-world deployment considerations, followed by the conclusion in Section 6.

## 2 Related work

According to our literature review, many studies have focused on detecting DDoS attacks using artificial intelligence.

Garcia and Blandon (2022) proposed an intrusion detection and prevention system (IDS/IPS) named Dique, designed to detect and mitigate Denial of Service (DoS) attacks. The system employs a deep learning–based classification model that analyses incoming web server packets and categorises them as either benign (normal traffic) or malicious (potential DoS traffic). Dique also features a graphical user interface (GUI) to facilitate monitoring and management. The proposed DoS attack detection model (Template 3) is based on a multilayer feedforward neural network and was trained using the CIC-DDoS-2019 dataset, achieving a classification accuracy of 0.994.

Similarly, Yungaicela-Naula et al. (2021) developed an architecture to detect transport- and application-layer DoS attacks in SDN environments. To achieve this, they evaluated different machine learning methods and deep learning models to determine the most effective approach. They used the CIC-DDoS-2019 and CIC-DoS-2017 datasets for training and testing and reported an accuracy above 0.99.

Nagaraju et al. (2023) proposed a deep learning model with the binary fruit fly algorithm as an optimiser to detect SYN flood attacks. To train and test the model, they used 18,230 samples from the KDD Cup-99 dataset. Their model achieved an accuracy of 0.9996 in detecting SYN flood attacks.

Evmorfos et al. (2020) used a random neural network to detect SYN flood attacks in IoT systems. For training and testing, the authors collected their own data in a laboratory environment, where they used Scapy, a Python-based tool for creating and manipulating packets. They did not provide additional details on data preprocessing or the number of features used. They provided a basic description of their model, as well as an LSTM model used for comparison. The descriptions included the structure of the models (number of layers and nodes), the loss function, and the optimiser. Their model achieved an accuracy of 0.807, compared to the LSTM model, which achieved an accuracy of 0.627 in detecting SYN flood attacks.

Devarakonda and Dasari (2022) evaluated three optimisation techniques (Adam, SGD, and LBFGS) and three correlation methods for feature selection (Pearson, Spearman, and Kendall) to determine which combination was most effective for detecting SYN flood attacks using MLP models. They used the CIC-DDoS-2019 dataset, which provides 2 GB of SYN flood traffic, and divided it into an 80% training set and a 20% test set. The authors used the three correlation methods to reduce the number of features from the original 83 to 22, 16, and 9 features using the Pearson, Spearman, and Kendall methods, respectively. Feature selection aims to speed up the training and data processing stages. The authors trained several MLP models with different optimisers and feature sets. All versions achieved an accuracy above 0.99, with the highest being the model trained on the Pearson feature set with Adam as the optimiser, achieving an accuracy of 0.9996. The paper does not provide details on the model structures or mention the training and testing times for each version.

Ortet Lopes et al. (2021) proposed CyDDoS, an integrated intrusion detection system (IDS) that combines a set of feature engineering techniques with a deep neural network. Feature selection is performed using an ensemble of five machine learning classifiers, which are employed to identify and extract the most relevant features for the predictive model.

Chartuni and Márquez (2021) proposed a multi-class DDoS attack classification system for computer networks based on artificial neural networks. The primary objective is to identify and distinguish among multiple types of DDoS attacks, enabling more fine-grained analysis of network traffic. The model leverages the learning capabilities of supervised neural networks to capture complex relationships between traffic features, thereby improving classification performance. The proposed approach achieved approximately 0.94 in key evaluation metrics, including precision, accuracy, recall, and F1-score.

The detection of Distributed reflection Denial of Service (DrDoS) attacks in IoT environments was investigated by Shurman et al. (2020), where two methodologies were proposed: a hybrid intrusion detection system (IDS) approach and a deep learning model based LSTM.

Bashaiwth et al. (2023) used an LSTM model to classify DDoS attacks. Predictions of 17 DDoS attack types are explained using the LIME, SHAP, Anchor, and LORE methods. The results highlight 51 intrinsic characteristics that can be used to classify attacks.

Roopak et al. (2020) proposed an IDS designed to protect IoT networks against DDoS attacks. The IDS is based on the fusion of a multi-objective optimisation method (NSGA-II), adapted to the Jumping Gene algorithm for data dimensionality reduction, and a convolutional neural network integrating LSTM deep learning techniques for attack classification. The experiment was conducted using CIS-IDS-2017 data and achieved an accuracy of 0.9903.

Salih and Abdulrazaq (2024) proposed lightweight deep learning (DL) models, referred to as Cyber-Net models, for the detection and recognition of distributed denial-of-service (DDoS) attacks. These models are designed with a limited number of trainable parameters (fewer than 225,000), justifying their classification as lightweight. The proposed models were evaluated using the CIC-DDoS-2019 dataset, and experimental results demonstrated strong performance, achieving 0.9999 in accuracy, recall, and F1-score for the detection model, and 0.9976 for the recognition model.

Boonchai et al. (2022) designed two models with a simple DNN structure and a convolutional autoencoder. The maximum accuracies obtained with these models reached 0.87 and 0.919, respectively. The overall results demonstrate the strong performance of the proposed networks, with high precision, recall, and F1-scores. The models were tested on the CIC-DDoS-2019 dataset.

Table 1 presents a summary of the analyzed studies, outlining the deep learning algorithms, datasets used, and accuracy obtained.

**Table 1.** Summary of analyzed studies

| | CIC-DoS-2017 | CIC-IDS-2017 | KDD Cup 99 | CIC-DoS-2019 |
|---|---|---|---|---|
| CNN | Yungaicela-Naula et al. (2021)<br>0.9888 | | | Garcia and Blandon (2022)<br>0.994<br>Yungaicela-Naula et al. (2021)<br>0.9913 |
| DNN | | | | Boonchai et al. (2022)<br>0.919<br>Ortet Lopes et al. (2021)<br>0.996 |
| MLP | Yungaicela-Naula et al. (2021)<br>0.9813 | | | Yungaicela-Naula et al. (2021)<br>0.9989<br>Devarakonda and Dasari (2022)<br>0.9996 |
| LSTM | Yungaicela-Naula et al. (2021)<br>0.9947 | Bashaiwth et al. (2023)<br>0.998 | | Yungaicela-Naula et al. (2021)<br>0.9987<br>Shurman et al. (2020)<br>0.9985<br>Bashaiwth et al. (2023)<br>0.998 |
| GRU | Yungaicela-Naula et al. (2021)<br>0.9942 | | | Yungaicela-Naula et al. (2021)<br>0.9980 |
| DLBFFA | | | Nagaraju et al. (2023)<br>0.9996 | |
| CNN LSTM | | Roopak et al. (2020)<br>0.9903 | | Salih and Abdulrazaq (2024)<br>0.99 |

Most of the work mentioned focuses primarily on the binary classification of DoS attacks, and these studies have shown satisfactory results. However, there is a significant difference in the accuracy of the model when tested on different attacks separately. For some attacks, the model manages to predict the attack correctly, with an accuracy similar to that of the training. On the other hand, for some attacks, the model fails to predict them correctly, with results significantly lower than the accuracy presented by the model. For example, in the model presented by Garcia and Blandon (2022), although it achieved an accuracy of 0.9994, it failed to correctly predict SYN flood and TCP flood attacks.

Another observation is that some studies use datasets that are several years old or do not contain enough data, which may not be sufficient for proper learning. An example is the study by Nagaraju et al. (2023), where the authors used only 18,230 examples from the KDD Cup 99 dataset. The dataset itself is old and contains only 743 MB of data in total. This amount of data is not sufficient to train a model that can be used in real-world applications. One aspect that studies limited to certain DoS attacks do not address is the relevance of the features selected for detection. For example, in the study by Devarakonda and Dasari (2022), correlation methods were used to select the least correlated features in order to avoid redundancy. However, the relevance of these features for detecting SYN flood attacks was not considered. The three lists of features used did not include the ACK flag count and the total number of bytes sent in the initial window in the forward direction (init-win-byte forward), which Garcia and Blandon (2022) considered to be highly relevant features in SYN flood traffic. The existence of additional features and the lack of relevant features has a negative effect on the model's performance.

## 3 Dataset and Experimental setup

### 3.1 Dataset

A dataset is a structured collection of data, organised in such a way as to facilitate access, management and analysis. In the context of deep learning, a dataset is a collection of examples, each comprising input data and corresponding output data.

These datasets are used to train and evaluate deep learning models. Choosing the right dataset is crucial to ensuring that the model is effectively trained to solve specific objectives Euijong Whang et al. (2021). It is also used to evaluate model performance during and after training, as well as to improve its ability to generalise to new data. In the context of our work, it is imperative to choose a recent dataset that reflects current trends and developments in denial of service attacks. This dataset must also be large, containing a large number of examples to enable comprehensive and representative analysis.

We drew on the study by Mittal et al. (2023a), in which the authors examined various DoS, DDoS, and DrDoS datasets. This study included 15 datasets from different environments published between 2012 and 2022. Table 2 presents the three datasets that we found most suitable for our work.

**Table 2.** Comparative study table of datasets.

| Dataset | Attack Type | Legitimate traffic | Labeled data | Availability |
|---|---|---|---|---|
| CIC-DDoS-2019 Sharafaldin et al. (2019) | NTP, DNS, LDAP, NetBIOS,SNMP,SSDP, UDP, TFTP,TCP | HTTP, HTTPS, FTP, SSH | Yes | Available |
| PVAMUDDoS-2020 Alam et al. (2021) | UDP, TCP, ICMP | HTTP, HTTPS, FTP Telnet, SSH, SMTP, RTSP, POP3, SIP, BitTorrent | Yes | Available on request |
| DDoS-AT-2022 Mittal et al. (2023a) | UDP, TCP, HTTP | HTTP, TCP, UDP, DNS, ICMP | Not specified | Not available |

We selected the CIC-DDoS-2019 dataset because it offers a significant amount of data and a wide range of attacks.

**CIC-DDoS 2019**: is a dataset generated by Sharafaldin et al. (2019). It contains DDoS and DrDoS attacks that resemble real-world data. It also includes the results of network traffic analysis with labelled flows based on timestamps, source and destination IP addresses, source and destination ports, protocols, and attacks. The dataset has been organised by day. Each day has raw recorded data, including network traffic and event logs (Windows and Ubuntu Event Logs) in the machines used. In the process of extracting features from the raw data, the authors used CICFlowMeter-V3 1and extracted more than 80 features. CIC-DDoS-2019 provides more realistic traffic using real network data.

The dataset is available in two formats, PCAP and CSV, on the Canadian Institute for Cybersecurity website [1]. The PCAP files contain raw traffic information captured by CICFlowmeter, and the CSV files are classified into flows, processed and labelled. The dataset starts with 87 attributes, then some network attributes were removed so that the model could be deployed in future work and other network structures Sharafaldin et al. (2019).

Table 3 shows the number of flows and the percentage of each type of attack available in the dataset, as well as legitimate traffic.

[1] https://www.unb.ca/cic/datasets/ddos-2019.html

**Table 3.** Dataset distribution table.

| Attack Type | Flow Count | Attack Percentage |
|---|---|---|
| TFTP | 20,082,580 | 39.65% |
| Syn | 1,582,289 | 3.12 % |
| UDPLag | 366,461 | 0.72 % |
| DNS | 5,071,011 | 10.01 % |
| LDAP | 2,179,930 | 4.30 % |
| NTP | 1,202,642 | 2.37 % |
| MSSQL | 4,522,492 | 8.93 % |
| SNMP | 5,159,870 | 10.19 % |
| NETBIOS | 4,093,279 | 8.08 % |
| UDP | 3,134,645 | 6.19 % |
| SSDP | 2,610,611 | 5.15 % |
| Portmap | 186,960 | 0.37 % |
| Legitimate traffic | 56,863 | 0.11 % |
| Total | 50,649,133 | 100% |

### 3.2 Proposed model

Our model is specifically designed to detect denial of service attacks, using an approach that integrates different types of neural networks to extract input data features and classify them efficiently.

The model consists of three distinct types of neural networks:

-**CNN**: is responsible for extracting local features from input data by identifying specific patterns in localised areas. These features enable an understanding of the spatial patterns specific to the data, which is essential for attack detection.

-**LSTM**: is responsible for extracting temporal features from input data, thereby capturing evolving patterns and inherent sequential dependencies. These features are crucial for understanding the sequential nature and temporal evolution of data, thereby contributing to more accurate attack detection.

-**DNN**: combines the features extracted by the CNN and LSTM to perform the final classification of the data, distinguishing attacks from legitimate activities.

Figure 1 illustrates the global architecture of our detection model, showing how these different components integrate for effective detection of denial of service attacks.

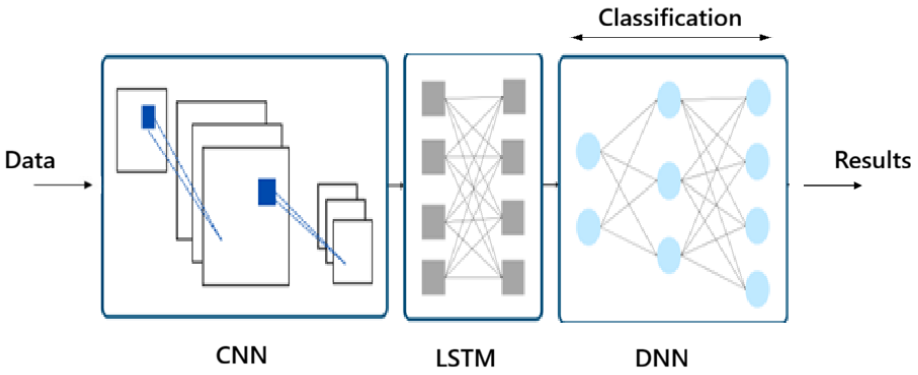


**Fig. 1.** Scheme of the proposed model.

The CNN part of our model, illustrated in Figure 2, consists of two one-dimensional convolution layers, two pooling layers and a normalisation layer. The normalisation layer stabilises and accelerates the learning process by normalising the activations of the previous layer. The max pooling layer reduces the spatial dimensions of the input by capturing the most important patterns. Instead of another max pooling layer, we used an average pooling layer that calculates the average value of each feature map, thus providing a more generalised representation.

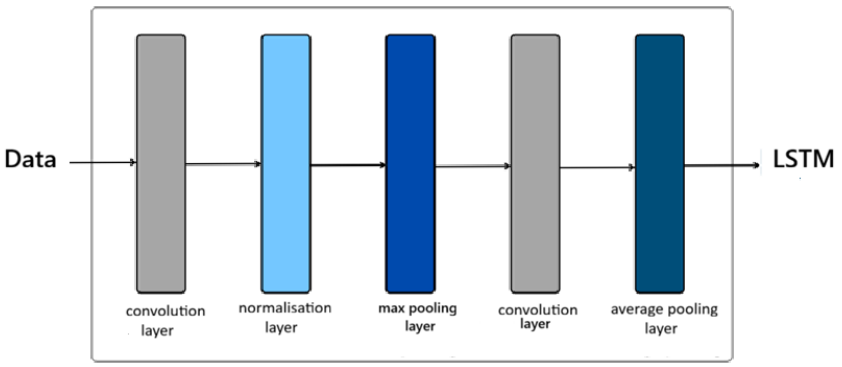


**Fig. 2.** Scheme of the CNN.

For the LSTM part illustrated in Figure 3, after the pooling layers, two recurrent LSTM layers are added to capture the temporal features crucial for attack detection. LSTM layers were chosen to avoid gradient vanishing issues.

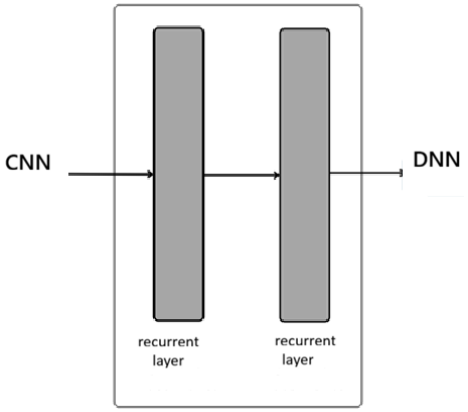


**Fig. 3.** Scheme of the LSTM.

Finally, once the local and temporal features have been captured, a four-layer DNN is used to classify the input data and effectively detect attacks. This DNN model has four dense layers and a dropout layer used to prevent overfitting by randomly removing a fraction of the neurons during training. The last dense layer (output layer) uses an activation function for final classification based on the number of classes. Figure 4 illustrates the structure of the DNN.

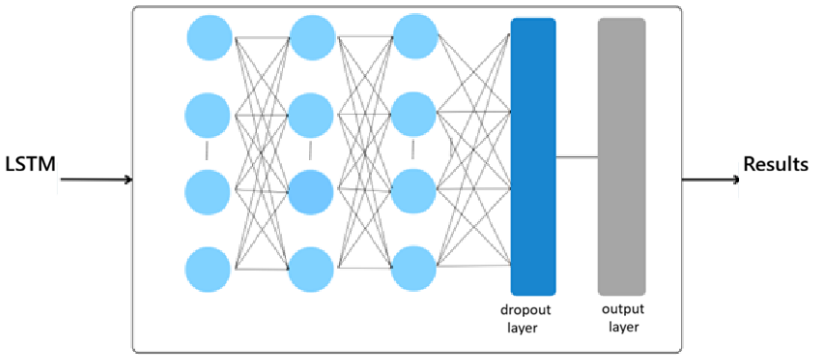


**Fig. 4.** Scheme of the DNN.

By combining CNN and LSTM, the hybrid model can effectively analyse data and extract local and temporal features from network traffic, enabling more robust and accurate detection of denial of service attacks. The DNN further optimises error and loss, improving the overall performance of the model.

### 3.3 Model Training

Before starting training, we divided the data set into training and test sets. We allocated 80%, of the data to training and 20 %,to testing. Figure 5 shows the distribution of flows in the training dataset.

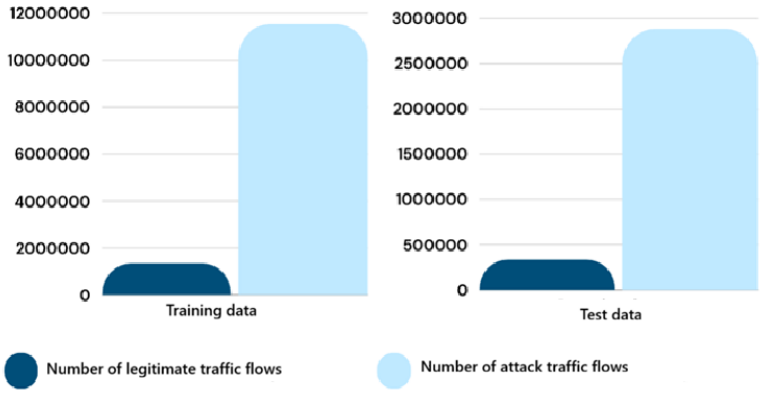


**Fig. 5.** Distribution of flows in the training and test datasets.

Optimising and initialising hyperparameters is an important step in developing a good deep learning modelHeaton (2018)Goodfellow (2016). Below is a list of some of the hyperparameters we chose to start our training.

- **Weight initialisation**: the Glorot uniform method stabilises the variance of layer inputs and outputs, which balances the model.
- **Initialising biases**: Initialising to zero is a common practice that ensures biases do not add erroneous initial contributions. This allows all features to be considered equally at the start of training.
- **Optimiser**: the use of the ADAM (Adaptive Moment Estimation) optimiser provides rapid and stable convergence of the loss function.
- **Learning rate**: initialised at 0.01, this hyperparameter may need to be adjusted to avoid missing the optimal convergence during training.
- **Number of epochs**: initialised at 10, this hyperparameter may need to be adjusted during training.
- **Loss function**: The cross-entropy function is often used in classification. It effectively measures the divergence between the model's predictions and the actual labels.
- **Activation function**: the ReLU function is one of the most widely used activation functions in deep learning. It is preferred for its simplicity, efficiency and ability to mitigate the gradient vanishing problem. We used ReLU for the hidden layers and softmax for the final classification layer because this function transforms values

into probabilities, making it easier to interpret the results by providing probabilities for each possible class.
– **Batch size**: initialised at 1,000, allowing us to benefit from batch calculations and efficient use of computational resources

In summary, these hyperparameter choices are the result of a methodical approach aimed at optimising model performance while ensuring efficient and stable training. Table 4 summarises our choices.

**Table 4.** Table of hyperparameter.

| Hyperparameter | Value |
|---|---|
| Weight initialization | Glorot uniform |
| Initialisation of biases | Zero |
| Epochs | 10 |
| Optimiser | ADAM |
| Learning rate | 0.01 |
| Loss function | Cross entropy |
| Batch size | 1000 |
| Convolutional layers | Number of filters: 64 |
| | Kernel size: 2 |
| Pooling layers | Window size: 2 |
| LSTM layers | Number of memory cells: 100 |
| | Dropout rate: 0.2 |
| Dense layers | Number of neurons: 100 |
| | Dropout rate: 0.3 |

In our training process, we adopted the trial and error principle to determine the best parameters. This method involves experimenting with different hyperparameter values in the hope of improving the model's performance. These adjustments include changing the number of neurons, adjusting the dropout rate, adjusting the learning rate, etc.

To ensure we chose the right hyperparameters, we reserved 25% of the training set for validation after each epoch. This allowed us to verify that the selected hyperparameters were the most appropriate for the model.

In the end, we changed the learning rate to 0.001, the number of epochs to 10-30, and added recurrent dropout with a rate of 0.2 to improve the model's performance.

## 4 Results and Discussions

In order to evaluate our proposed model, we compared it with some deep learning models that use the same CICDDoS-2019 dataset. We also extracted and trained CNN, LSTM, and DNN neural networks separately from the proposed model in order to compare them with our model.

### 4.1 Binary classification

After training, we evaluated the model using the test set. The final accuracy of our model is 0.9996, reflecting its training accuracy and highlighting the robustness of our model. Table 5 presents the results obtained.

The confusion matrix (Figure 6) demonstrates the robustness of our model in distinguishing between legitimate traffic and DDoS attacks. The matrix results confirm the low false alarm rate and the rarity of undetected attacks.

**Table 5.** Table of binary classification results.

| Model | Precision | Recall | F1-score |
|---|---|---|---|
| Legitimate traffic | 0.9977808 | 0.998469 | 0.0998138 |
| Attack | 0.999821 | 0.999734 | 0.999782 |

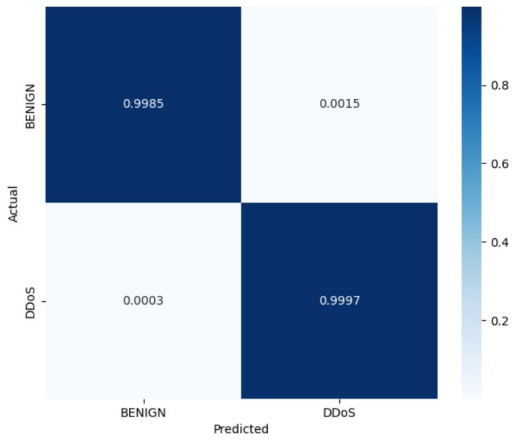


**Fig. 6.** Binary classification confusion matrix.

Table 6 shows the accuracy of the different deep learning algorithms, compared to our proposed model, for binary classification.

Our model stands out with an accuracy of 0.99961, placing it among the top-performing models. It slightly outperforms LSTM, DNN, and CNN+LSTM models, highlighting its strong performance and reliability for classification tasks.

**Table 6.** Comparative table of binary classification results.

| Model | Accuracy |
|---|---|
| CNN | 0.9829 |
| LSTM | 0.9995 |
| DNN | 0.9985 |
| LSTM Shurman et al. (2020) | 0.9985 |
| DNN Ortet Lopes et al. (2021) | 0.9960 |
| CNN+LSTM Roopak et al. (2020) | 0.9903 |
| proposed model | 0.99961 |

### 4.2 Multiclass classification

#### a. four-class classification

The accuracy of our model is 0.97 for four-class classification. Table 7 presents the results obtained after evaluating the model on the test set.

**Table 7.** Table of results obtained from four-class classification.

| Model | Precision | Recall | F1-score |
|---|---|---|---|
| Legitimate Traffic | 0.998114 | 0.999775 | 0.998944 |
| DNS/LDAP/SNMP/NetBIOS Attacks | 0.984164 | 0.981519 | 0.982839 |
| MSSQL Attacks | 0.908920 | 0.897391 | 0.903119 |
| NTP/SSDP/UDP/SYN Attacks | 0.965303 | 0.975330 | 0.970291 |

Analysis of the results shows that the model offers remarkably high performance in network traffic classification. Legitimate traffic is almost perfectly identified, with precision, recall and F1 scores approaching 1. The combined categories of DNS/LDAP/SNMP/NetBios and NTP/SSDP/UDP/SYN attacks also show solid performance, with F1 scores of 0.982839 and 0.970291 respectively, indicating a good ability to detect these types of attacks. The MSSQL attack, although slightly lower, still has a respectable F1 score of 0.903119, showing reliable performance. The model's overall results, with precision, recall and F1 scores around 0.97, demonstrate excellent effectiveness in detecting and classifying network traffic and various attacks. However, there is still a slight margin for improvement in detecting MSSQL attacks.

Confusion matrix (Figure 7) shows that the model performs well with a high true positive rate for most classes. Some classification errors were observed, but they are minimal and do not affect the overall quality of the model.

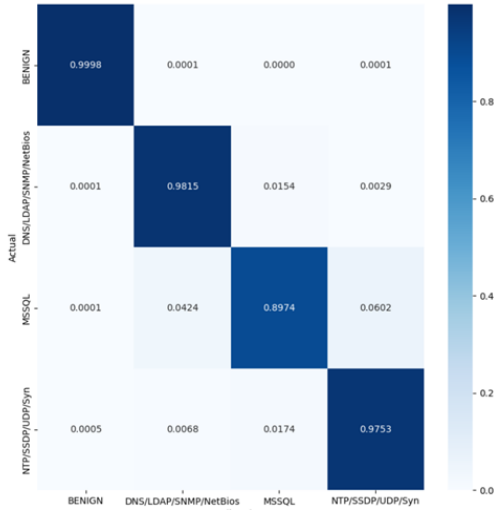


**Fig. 7.** Distribution of flows in the training and test datasets.

Table 8 shows the accuracy of the different deep learning algorithms, compared to our proposed model, for four-class classification.

**Table 8.** Comparative table of four-class classification results.

| Model | Accuracy |
|---|---|
| CNN | 0.95028 |
| LSTM | 0.92987 |
| DNN | 0.95098 |
| LSTM Bashaiwth et al. (2023) | 0.98 |
| proposed model | 0.97 |

Our model achieved an accuracy of 0.97 and demonstrated highly competitive performance, outperforming individual CNN, LSTM, and DNN models, while remaining slightly inferior to the LSTM model reported by Bashaiwth et al. (2023).

### b. eight-class classification

The accuracy of our model is 0.8867 for eight-class classification. Table 9 presents the results obtained after evaluating the model on the test set.

**Table 9.** Table of results obtained from eight-class classification.

| Model | Precision | Recall | F1-score |
|---|---|---|---|
| Legitimate Traffic | 0.997696 | 0.999828 | 0.998761 |
| DNS/LDAP Attacks | 0.883951 | 0.798548 | 0.839082 |
| MSSQL Attacks | 0.762448 | 0.958287 | 0.849223 |
| NTP Attacks | 0.977481 | 0.824031 | 0.894221 |
| NetBIOS Attacks | 0.933914 | 0.994874 | 0.963430 |
| SNMP Attacks | 0.779389 | 0.832442 | 0.805042 |
| SSDP/UDP Attacks | 0.973401 | 0.822011 | 0.891323 |
| SYN Attacks | 0.999539 | 0.997884 | 0.998711 |

Analysis of the results shows that the model performs excellently in classifying legitimate traffic, with precision, recall and F1 scores all close to 1. SYN and NetBios attacks are also detected well, with very high F1 scores. However, certain categories of attacks, such as MSSQL and SNMP, show relatively lower performance, with F1 scores of 0.849223 and 0.805042 respectively, indicating some difficulty in identifying them correctly. DNS/LDAP and NTP attacks, although better detected, still have room for improvement. Overall, the model shows good robustness with overall accuracy, recall and F1 scores close to 0.89, but there are still opportunities to improve the detection of certain categories of attacks.

Confusion matrix (Figure 8) shows that the model performs well with a high true positive rate for Netbios and SYN attacks, as well as low false negatives indicating accurate predictions. Some classification errors were observed for other types of attacks, indicating that there is room for improvement in the model.

Table 10 shows the accuracy of the different deep learning algorithms, compared to our proposed model, for eight-class classification.

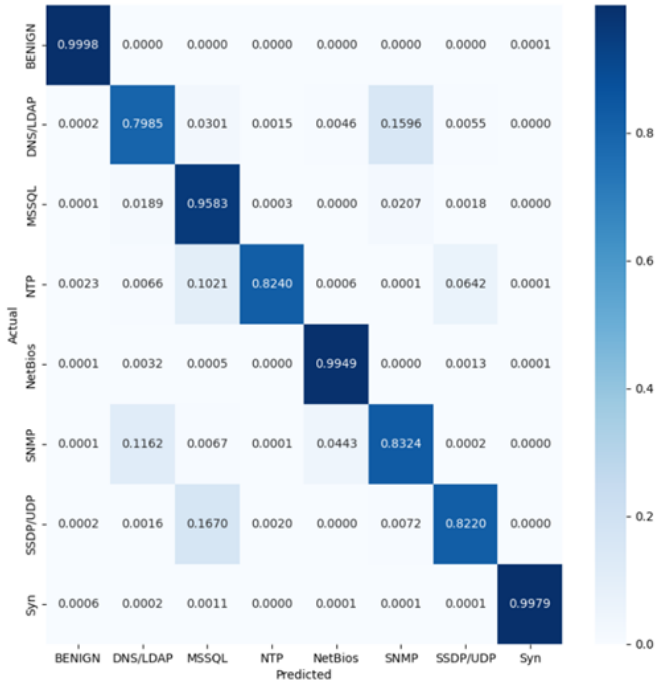


**Fig. 8.** Distribution of flows in the training and test datasets.

**Table 10.** Comparative table of eight-class classification results.

| Model | Accuracy |
|---|---|
| CNN | 0.870123 |
| LSTM | 0.884052 |
| DNN | 0.885907 |
| DNN+CNN Boonchai et al. (2022) | 0.87 |
| DNN Chartuni and Márquez (2021) | 0.94 |
| proposed model | 0.886715 |

Our model, with an accuracy of 0.886715, slightly outperforms the CNN, LSTM, DNN, and DNN+CNN models presented in Boonchai et al. (2022). However, it remains inferior to the DNN model reported in Chartuni and Márquez (2021), which achieved an accuracy of 0.94. These results highlight the advantages of our approach while indicating that there is still room for improvement.

## 5 Real-World Deployment Considerations

It is important to emphasize the fundamental difference between DDoS attack classification and DDoS mitigation mechanisms. DDoS attacks manifest as abnormal traffic patterns, including high-rate volumetric floods, protocol exploitation, and application-layer request saturation. In contrast, mitigation strategies involve active network-level interventions such as traffic filtering, rate limiting, blackholing, or traffic rerouting through scrubbing centers.

In this context, the proposed model does not directly perform mitigation actions. Instead, it operates as a decision-support component that identifies the type of DDoS attack based on observed traffic behavior. Although this task is classification-oriented, it is operationally meaningful because mitigation strategies are typically attack-type dependent. For example, volumetric attacks are often mitigated using rate limiting or upstream filtering, whereas application-layer attacks may require behavioral analysis or deep packet inspection. Consequently, inaccurate classification may lead to inappropriate mitigation policies and reduced system effectiveness.

In realistic cybersecurity architectures, attack classification modules are commonly integrated into multi-layer defense pipelines involving firewalls, Intrusion Prevention Systems (IPS), and Software-Defined Networking (SDN) controllers. In such systems, the classification output serves as an input signal for higher-level policy engines that determine the appropriate mitigation response. From this perspective, the proposed model can be viewed as an intelligence layer that supports automated decision-making rather than a standalone mitigation solution.

From a deployment standpoint, operational DDoS defense systems must satisfy strict constraints, including low latency, high throughput, and continuous processing of high-volume traffic streams. In contrast to offline experiments on benchmark datasets, real network environments are characterized by noise, concept drift, and severe class imbalance. While the proposed model is evaluated in an offline setting, its design prioritizes inference efficiency and makes it compatible with near real-time deployment scenarios. Nevertheless, practical deployment in large-scale infrastructures may require additional system-level optimizations, such as model compression, pruning, or hardware acceleration, to meet stringent performance requirements.

## 6 Conclusion

This paper proposes a hybrid deep learning-based model that combines CNN, LSTM, and DNN architectures for DDoS attack classification. The model is evaluated on the CIC-DDoS-2019 dataset, achieving an accuracy of 0.9996 for binary classification and 0.97 for multiclass classification tasks.

Despite these promising results, the study has several limitations. The evaluation is conducted using benchmark datasets, which may not fully capture the complexity of real-world network environments, including encrypted traffic, evolving attack patterns, and traffic noise. In addition, the proposed approach has not been validated in a live production environment or under real-time streaming conditions. Therefore, the presented work should be considered as a research-oriented classification framework designed to support, rather than replace, operational DDoS defense systems.

Future work will focus on improving the practical applicability of the proposed approach. This includes integration with SDN/NFV-based network architectures, evaluation on real-time traffic streams, and the development of adaptive learning strategies capable of handling evolving attack behaviors. Further research will also address computational efficiency and scalability to enable deployment in high-speed network environments.

## References

Adedeji, K. B., Abu-Mahfouz, A. M. and Kurien, A. M. (2023). Ddos attack and detection methods in internet-enabled networks: Concept, research perspectives, and challenges, *Journal of Sensor and Actuator Networks* **12**(4): 51.

Ahsan, M., Nygard, K. E., Gomes, R., Chowdhury, M. M., Rifat, N. and Connolly, J. F. (2022). Cybersecurity threats and their mitigation approaches using machine learning—a review, *Journal of Cybersecurity and Privacy* **2**(3): 527–555.

Al-Hadhrami, Y. and Hussain, F. K. (2021). Ddos attacks in iot networks: a comprehensive systematic literature review, *World Wide Web* **24**(3): 971–1001.

Al-Shareeda, M. A., Manickam, S. and Saare, M. A. (2023). Ddos attacks detection using machine learning and deep learning techniques: analysis and comparison, *Bulletin of Electrical Engineering and Informatics* **12**(2): 930–939.

Alam, S., Alam, Y., Cui, S., Akujuobi, C. and Chouikha, M. (2021). Toward developing a realistic ddos dataset for anomaly-based intrusion detection, *2021 IEEE International Conference on Consumer Electronics (ICCE)*, IEEE, pp. 1–6.

Alatawi, F. (2021). Defense mechanisms against distributed denial of service attacks: comparative review, *Journal of Information Security and Cybercrimes Research* **4**(1): 81–94.

Banitalebi Dehkordi, A., Soltanaghaei, M. and Boroujeni, F. Z. (2021). The ddos attacks detection through machine learning and statistical methods in sdn: Ab dehkordi et al., *The Journal of Supercomputing* **77**(3): 2383–2415.

Bashaiwth, A., Binsalleeh, H. and AsSadhan, B. (2023). An explanation of the lstm model used for ddos attacks classification, *Applied Sciences* **13**(15): 8820.

Boonchai, J., Kitchat, K. and Nonsiri, S. (2022). The classification of ddos attacks using deep learning techniques, *2022 7th International Conference on Business and Industrial Research (ICBIR)*, IEEE, pp. 544–550.

Bravo, S. and Mauricio, D. (2018). Distributed denial of service attack detection in application layer based on user behavior., *Webology* **15**(2).

Chartuni, A. and Márquez, J. (2021). Multi-classifier of ddos attacks in computer networks built on neural networks, *Applied Sciences* **11**(22): 10609.

De Neira, A. B., Kantarci, B. and Nogueira, M. (2023). Distributed denial of service attack prediction: Challenges, open issues and opportunities, *Computer Networks* **222**: 109553.

Devarakonda, N. and Dasari, K. (2022). Syn flood ddos attack detection with different multilayer perceptron optimization techniques using uncorrelated feature subsets selected by different correlation methods, *International Conference on Computer Vision, High-Performance Computing, Smart Devices, and Networks*, Springer, pp. 249–260.

Eliyan, L. F. and Di Pietro, R. (2021). Dos and ddos attacks in software defined networks: A survey of existing solutions and research challenges, *Future Generation Computer Systems* **122**: 149–171.

Euijong Whang, S., Roh, Y., Song, H. and Lee, J.-G. (2021). Data collection and quality challenges in deep learning: A data-centric ai perspective, *arXiv e-prints* pp. arXiv–2112.

Evmorfos, S., Vlachodimitropoulos, G., Bakalos, N. and Gelenbe, E. (2020). Neural network architectures for the detection of syn flood attacks in iot systems, *Proceedings of the 13th ACM International Conference on PErvasive Technologies Related to Assistive Environments*, pp. 1–4.

Garcia, J. F. C. and Blandon, G. E. T. (2022). A deep learning-based intrusion detection and preventation system for detecting and preventing denial-of-service attacks, *IEEE Access* **10**: 83043–83060.

Goodfellow, I. (2016). Deep learning.

Heaton, J. (2018). Ian goodfellow, yoshua bengio, and aaron courville: Deep learning: The mit press, 2016, 800 pp, isbn: 0262035618, *Genetic programming and evolvable machines* **19**(1): 305–307.

Khadka, S. K., Bayhan, S., Holz, R., Shokoohi, S., Barcellos, M. and Hesselman, C. (2025). A first look at the adoption of bgp-based ddos scrubbing services: A 5-year longitudinal analysis, *2025 21st International Conference on Network and Service Management (CNSM)*, IEEE, pp. 1–7.

Li, Q., Huang, H., Li, R., Lv, J., Yuan, Z., Ma, L., Han, Y. and Jiang, Y. (2023). A comprehensive survey on ddos defense systems: New trends and challenges, *Computer Networks* **233**: 109895.

M Nalayini, C. and Katiravan, J. (2022). Detection of ddos attack using machine learning algorithms, *Available at SSRN 4173187* **9**(7).

Mittal, M., Kumar, K. and Behal, S. (2023a). Ddos-at-2022: a distributed denial of service attack dataset for evaluating ddos defense system, *Proceedings of the Indian National Science Academy* **89**(2): 306–324.

Mittal, M., Kumar, K. and Behal, S. (2023b). Deep learning approaches for detecting ddos attacks: A systematic review, *Soft computing* **27**(18): 13039–13075.

Mughal, A. A. (2018). Artificial intelligence in information security: Exploring the advantages, challenges, and future directions, *Journal of Artificial Intelligence and Machine Learning in Management* **2**(1): 22–34.

Nagaraju, V., Raaza, A., Rajendran, V. and Ravikumar, D. (2023). Deep learning binary fruit fly algorithm for identifying syn flood attack from tcp/ip, *Materials Today: Proceedings* **80**: 3086–3091.

Nuiaa, R. R., Manickam, S. and Alsaeedi, A. H. (2021). Distributed reflection denial of service attack: A critical review, *International Journal of Electrical and Computer Engineering* **11**(6): 5327.

Ortet Lopes, I., Zou, D., Ruambo, F. A., Akbar, S. and Yuan, B. (2021). Towards effective detection of recent ddos attacks: A deep learning approach, *Security and Communication Networks* **2021**(1): 5710028.

Reddy, R. P. (2024). A survey of distributed denial of service (ddos) attack mitigation techniques, *International Journal of Computer Trends and Technology (IJCTT)* **72**(12): 69–77.

Roopak, M., Tian, G. Y. and Chambers, J. (2020). An intrusion detection system against ddos attacks in iot networks, *2020 10th annual computing and communication workshop and conference (CCWC)*, IEEE, pp. 0562–0567.

Salih, A. and Abdulrazaq, M. (2024). Cybernet model: A new deep learning model for cyber ddos attacks detection and recognition, *Computers, Materials, & Continua* **78**(1): 1275.

Sharafaldin, I., Lashkari, A. H., Hakak, S. and Ghorbani, A. A. (2019). Developing realistic distributed denial of service (ddos) attack dataset and taxonomy, *2019 international carnahan conference on security technology (ICCST)*, IEEE, pp. 1–8.

Shurman, M., Khrais, R., Yateem, A. et al. (2020). Dos and ddos attack detection using deep learning and ids, *Int. Arab J. Inf. Technol* **17**(4A): 655–661.

Sinha, M. (2025). A comprehensive survey of ddos attack defense systems for different sdn architectures, *Computer Networks* p. 111711.

Ussatova, O., Zhumabekova, A., Begimbayeva, Y., Matson, E. T. and Ussatov, N. (2022). Comprehensive ddos attack classification using machine learning algorithms., *Computers, Materials & Continua* **73**(1).

Yungaicela-Naula, N. M., Vargas-Rosales, C. and Perez-Diaz, J. A. (2021). Sdn-based architecture for transport and application layer ddos attack detection by using machine and deep learning, *Ieee Access* **9**: 108495–108512.

Received February 1, 2026 , revised May 24, 2026, accepted July 8, 2026