Applied Computing and Informatics

Developing realistic distributed denial-of-service (DDoS) attack dataset for software-defined networking (SDN)

Mahmoud Hassan, Khaled Metwally and Mohamed Elshafey
Department of Computer Engineering and Artificial Intelligence, Military Technical College, Cairo, Egypt

Received 3 July 2025
Revised 6 February 2026
Accepted 30 March 2026

Abstract

Purpose – With the rapid proliferation of Internet-connected devices, software-defined networking (SDN) has become an attractive paradigm for efficient network management compared to traditional network infrastructure. Nevertheless, SDN networks remain susceptible to distributed denial-of-service (DDoS) attacks, significantly degrading network performance and reliability. This research seeks to overcome the current limitation posed by the absence of realistic SDN-based datasets for DDoS detection and proposes an enhanced detection methodology tailored explicitly for SDN networks.

Design/methodology/approach – This research introduces SDN-DDoS2025, a novel dataset developed by adapting the CIC-DDoS2019 dataset to an SDN environment while preserving flow- and traffic-level characteristics unique to SDN. Comprehensive exploratory data analysis (EDA) is performed to thoroughly investigate feature distributions, trends and correlations within the dataset. Furthermore, a hybrid deep learning model integrating one-dimensional convolutional neural networks with long short-term memory layers (CNN-LSTM) is proposed, demonstrating enhanced efficiency and effectiveness in detecting DDoS attacks.

Findings – The experimental results reveal that the proposed CNN-LSTM hybrid model outperforms recent benchmark methods on both the original CIC-DDoS2019 and the newly introduced SDN-DDoS2025 datasets. The model attains superior detection accuracy and demonstrates robust generalization capabilities, handling real-world SDN traffic patterns.

Originality/value – This research presents the first SDN-specific dataset derived from the CIC-DDoS2019 benchmark, incorporating realistic SDN-oriented features and proposes a robust deep learning architecture specifically designed for effective DDoS detection in SDN contexts. By integrating dataset generation, comprehensive EDA and advanced deep learning modeling, this research addresses a significant research gap within the SDN security domain.

Keywords DDoS, CIC-DDoS2019, SDN-DDoS2025, SDN, EDA, CNN, LSTM, CNN-LSTM

Paper type Research article

1. Introduction and background

Traditional network infrastructures, though foundational for several decades, are inherently challenging to deploy and manage due to their reliance on manual configurations for any operational adjustments. Such complexity frequently results in human errors, device incompatibilities and interruptions in service availability. Additionally, these networks exhibit limited flexibility in adapting swiftly to failures or fluctuations in traffic conditions [1].

Software-defined networking (SDN) is an agile and economically viable networking paradigm that distinctly separates the control plane from the data plane, enhancing programmability and adaptability to contemporary high-bandwidth requirements. This

Conflict of interest: The authors declare no commercial or financial conflicts of interest related to this research.

Applied Computing and Informatics
Emerald Publishing Limited
e-ISSN: 2210-8327
p-ISSN: 2634-1964
DOI 10.1108/ACI-07-2025-0278

architectural transformation emphasizes stronger network security mechanisms such as advanced firewalls and antivirus solutions to reinforce protection against cyber threats [2].

As depicted in Figure 1, a typical SDN architecture comprises three fundamental planes: the control plane, data plane and application plane, all coordinated through centralized control and programmable network functionalities. The application plane and the control plane exchange information via the northbound Application Programming Interfaces (APIs), whereas OpenFlow primarily facilitates interactions between the control and data planes through southbound APIs. SDN decouples the control plane from the data plane, allowing flexible network management through programmable software applications [3].

From a cybersecurity perspective, denial-of-service (DoS) and distributed denial-of-service (DDoS) attacks represent critical threats to online services and infrastructure. Both attack types aim to disrupt system functionality by overwhelming system resources with malicious traffic, thereby preventing legitimate user access. In a DoS attack, a single host generates excessive traffic to overwhelm the target machine. On the other hand, a DDoS attack leverages multiple coordinated sources, typically forming a botnet, to inundate and incapacitate the target network or service [4].

In an SDN network, DDoS attacks can target various SDN critical components, i.e. controllers, forwarding devices, Northbound APIs, or communication links, resulting in substantial disruptions to network performance. Unlike traditional networks characterized by tightly integrated control and data planes, SDN distinctly separates these planes, making the SDN centralized controller a primary vulnerability. Although SDN inherently provides improved security mechanisms, the centralization of control underscores the necessity of securing the controller to maintain overall network integrity [5].

The primary contributions of this paper are summarized as follows:

(1) Introduction and detailed description of a new SDN-based dataset, namely SDN-DDoS2025, derived from the established CIC-DDoS2019 benchmark. The dataset was generated within a controlled SDN environment with OpenFlow-specific attributes.

(2) Perform a comprehensive exploratory data analysis (EDA) on SDN-DDoS2025, examining feature correlations, skewness, central tendency and outlier analysis.

(3) Propose an improved hybrid Deep Learning (DL) model combining convolutional neural networks (CNN) and long short-term memory (LSTM) models for effective DDoS detection. Experimental validation demonstrates that the proposed CNN-LSTM model achieves superior performance, robustness and generalizability compared to recent state-of-the-art methods across both traditional and SDN networks.

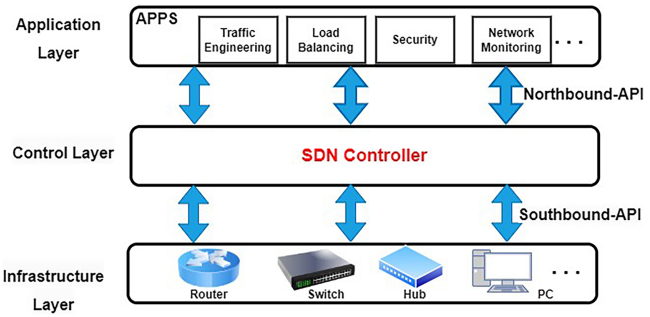

Figure 1. SDN architecture. **Source**: Created by the authors

The remainder of the paper is structured as follows. Section 2 reviews related literature and datasets pertinent to DDoS detection. Section 3 outlines the methodology employed in the generation of the proposed SDN-DDoS2025 dataset. Section 4 details the architecture and implementation of the proposed CNN-LSTM model for DDoS detection. Section 5 presents the experimental setup, evaluation metrics and comparative analysis results. Finally, Section 6 concludes the paper and discusses future research directions.

2. Related work

2.1 Candidate DDoS attack datasets

Several datasets exist that are being used for DDoS detection in networks. Some of them are for traditional networks and others for SDN. The UNSW-NB15 dataset, developed by the Australian Centre for Cyber Security in 2015, comprises 49 features and 9 attack types, including DoS and Exploits. Although it provides realistic, labeled network traffic, it is not tailored for DDoS attacks, with class imbalance and incomplete sample coverage that restrict its applicability in SDN-specific studies [6].

The CIC-IDS2017 dataset, developed by the Canadian Institute for Cybersecurity (CIC) in 2017, includes 80 features and various labeled attacks such as DoS, DDoS and Brute Force. However, it is not specifically focused on DDoS, has class imbalance, large file sizes and lacks SDN-specific design [7].

CIC-IDS 2018 is an enhanced version of the CIC-IDS 2017 dataset. It has 80 features and covers a range of attacks, including Brute Force, Heartbleed, Botnet, DoS, DDoS, Web assaults and network penetration. It provides a detailed feature set and simulates real-world traffic. Its drawbacks include high-class imbalance and resource-intensive processing. It does not support SDN networks [7].

CIC-DDoS2019 was developed by the CIC. This dataset contains 88 features; this is a comprehensive and up-to-date DDoS attack dataset. It offers detailed labels and is large enough for in-depth DDoS analysis. However, it has imbalanced classes, can be slow to process due to its large volume and requires SDN adaptation for some use cases. It has limited support for SDN networks [8].

The SDNFlow dataset was published by researchers from the University of Catania, Italy. SDNFlow dataset includes 40–50 features and is specific to SDN. It provides SDN-specific metrics and flow-level statistics. Its limitations include class imbalance, limited public documentation, resource-intensive processing and high dimensionality. It supports SDN [9].

InSDN [10] is a publicly available intrusion detection dataset created specifically for SDN environments. InSDN includes 83 flow-level features collected directly from an SDN controller, allowing for native SDN-aware analysis. The dataset includes a variety of attack categories, including DDoS, probing, botnet activity and brute-force attacks, as well as benign traffic. Its SDN-centric design makes it ideal for testing machine learning and deep learning approaches aimed at controller-level visibility and flow behavior in SDN architectures. Despite its advantages, InSDN presents several practical challenges. The large number of features and the volume of collected flow records lead to high computational and memory requirements during preprocessing and model training. This can limit its applicability in real-time or resource-constrained SDN controllers, especially when complex deep learning models are employed.

BoT-IoT [11] is a large-scale Internet of Things security dataset that was released in 2018. It was originally designed to model realistic botnet behavior in IoT environments. The dataset contains 46 extracted features covering a variety of attack types such as DDoS, DoS and network scanning. Although BoT-IoT was not originally collected in an SDN environment, it has been extensively reused in SDN-based research by mapping its flow features to SDN controller statistics, allowing for partial SDN applicability.

TON-IoT [12], which was released in 2020, is an extension of previous IoT security datasets that includes telemetry data gathered from network, operating system and IoT sensors.

The dataset covers a wide range of attacks, including DDoS, DoS and malware-related activities, and contains 44 features suitable for intrusion detection research. Similar to BoT-IoT, TON-IoT has been used in SDN-based research by integrating network traffic traces with SDN simulation platforms.

CICEV2023 was developed by CIC researchers at the University of New Brunswick, a specialized dataset intended to investigate DDoS attacks that target the authentication procedure in electric vehicle (EV) charging infrastructures. CICEV2023 focuses on how attackers can paralyze charging stations by flooding them with fraudulent or corrupted authentication requests, in contrast to traditional network datasets that concentrate on general traffic [13].

Table 1 provides a comparative overview of the most commonly used public datasets for DDoS detection, with a focus on their suitability for SDN environments. It is clear that traditional datasets, e.g. UNSW-NB15 and CIC-IDS series, provide rich and diverse attack scenarios, but lack native SDN awareness and thus require additional adaptation when used in SDN-based studies. In contrast, datasets like InSDN provide flow-level and controller-aware features that are specifically designed for SDN environments, albeit at the expense of increased computational complexity. Furthermore, IoT-oriented datasets, such as BoT-IoT and TON-IoT, are frequently reused in SDN research via emulation or feature mapping, providing valuable large-scale DDoS traffic patterns but with limited controller visibility. Overall, the tabulated datasets highlighted the lack of a unified dataset that provides comprehensive DDoS coverage, balanced labeling and native SDN support, emphasizing the need for improved or hybrid dataset generation to enable robust DDoS detection in SDN architectures.

Moreover, most of the aforementioned datasets are imbalanced, high-dimensional and lack essential SDN-specific attributes such as flow installation times, rule modifications and real-time network state changes, limiting their suitability for SDN research. Although CIC-

Table 1. Publicly available datasets used in DDoS and DDoS-SDN research

Dataset	Year	# Features	Attacks	DDoS focus	SDN support	Limitations
UNSW-NB15 [6]	2015	49	DoS, Exploits, Fuzzers, Reconnaissance	Partial	No	Not SDN-aware; class imbalance
CIC-IDS2017 [7]	2017	80	DoS, DDoS, Brute Force, Web attacks	Partial	No	Large size; no SDN context
CIC-IDS2018 [8]	2018	80	Botnet, DoS, DDoS, Web assaults	Partial	No	High imbalance; resource-intensive
CIC-DDoS2019 [9]	2019	88	Multiple DDoS variants (UDP, SYN, HTTP)	Yes	Limited	Large volume; SDN adaptation required
SDNFlow [14]	2018	40–50	DDoS, benign traffic	Yes	Yes	Limited documentation; imbalance
InSDN [10]	2020	83	DDoS, Probe, Botnet, Brute Force	Yes	Yes	High computational cost
BoT-IoT (SDN-used) [11]	2018	46	DDoS, DoS, scanning	Partial	No	Not natively SDN
TON-IoT (SDN-used) [12]	2020	44	DDoS, malware, DoS	Partial	No	Limited SDN awareness
CICEV 2023 [13]	2023	~70	Authentication-based DDoS	Yes (domain-specific)	No	Highly specialized

DDoS2019 is widely adopted for its comprehensive DDoS traffic coverage [8, 14, 15], it lacks native SDN characteristics. To address this, the proposed SDN-DDoS2025 dataset is specifically generated within an SDN environment, capturing critical features such as traffic management behavior, flow rule dynamics and controller-level metrics to support accurate modeling and evaluation of SDN-based security solutions.

2.2 Related works for DDoS detection

In the context of DDoS detection in SDN networks, this research suggests a unique method for detecting DDoS attacks in SDN systems. The proposed approach improved the detection capabilities by utilizing two popular DL techniques: CNN and LSTM. To fortify the originality of this study, the paper attempts to examine relevant literature work in the IDS context.

The authors in Ref. [16] provided a CNN-based approach, named CNN-BRS, for efficient detection and mitigation of DDoS attacks. They proposed an efficient approach using balanced random sampling (BRS) and CNNs to detect DDoS attacks in SDN environments. The proposed model achieved high performance in binary and multi-classification using the CIC-DDoS2019 dataset, but not on large-scale SDN networks. The authors adopted Softmax as an activation function, unsuitable for binary classification.

The Bays-CNN model integrates Bayesian machine learning with CNN, utilizing principal component analysis for feature extraction on the CIC-DDoS2019 dataset. While the model demonstrated strong performance across standard metrics, undersampling introduced data loss and evaluations were limited to individual dataset types without hybrid analysis [17]. In Ref. [18], a CNN-based IDS (CNN-WRS) was proposed, employing a weighted random sampler (WRS) to address class imbalance and shift from binary to multi-class DDoS detection. Although the approach improved accuracy and efficiency, it introduced sampling bias and overfitting risks due to imbalanced attack class sizes.

In [19], Cybernet, a lightweight DL model with fewer than 225,000 parameters, was introduced for efficient DDoS detection. By combining 1D-CNN and LSTM, the model demonstrated resilience and speed on the CIC-DDoS2019 dataset. However, its use of short training epochs may lead to underfitting on large or complex datasets, and the inclusion of LSTM layers significantly increases memory and computational demands compared to CNN-only architectures.

In summary, in all the aforementioned related works for DDoS detection, the widely adopted datasets are based on the captured traditional network traffic, even if the SDN network was mentioned in the research. While CIC-DDoS2019 and CIC-IDS2017 are widely used datasets for DDoS studies in SDN, they do not include SDN-specific design considerations that can affect DDoS detection accuracy in SDN networks. The direct applicability of these datasets to SDN environments is in need of additional adaptation or complementary data.

3. Proposed methodology

3.1 Proposed SDN-DDoS2025 dataset generation

It is worth noting that CIC-DDoS2019 is a widely used dataset for DDoS detection, featuring traditional network traffic with benign flows and 13 common DDoS attacks, including NTP, DNS, LDAP, MSSQL and others. It captures the behavior of 25 users across HTTP, HTTPS, FTP, SSH and email protocols, available in both PCAP and CSV formats for download [8].

As illustrated in Figure 2, the CIC-DDoS2019 authors constructed two distinct network segments: an Attack-Network and a Victim-Network. The Victim Network features a secure architecture comprising a firewall, router, switches, multiple widely-used operating systems and a benign agent on each host. The Attack-Network is a third-party infrastructure that facilitates the launch of DDoS attacks. Network traffic was captured in PCAP format and subsequently processed using CICFlowMeter-V3, which generated bidirectional flow records and extracted 88 statistical features stored in CSV format [8]. The dataset's authors stated that

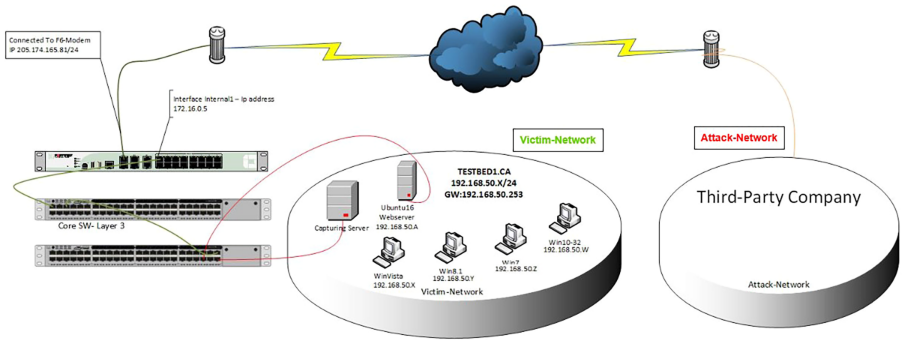

Figure 2. Typical CIC-DDoS2019 network architecture. **Source**: Created by the authors of CIC-DDoS2019

researchers can directly utilize CSV files for preprocessing and analysis or apply a feature extraction tool for additional features from raw PCAP files, enabling diverse analytical approaches [20].

From the aforementioned considerations, this study aims to construct a novel SDN-based dataset by leveraging the raw PCAP files from CIC-DDoS2019. As a result, the CIC-DDoS2019 raw PCAP files are replayed within a controlled SDN environment, enabling the generation of a new SDN-based dataset, namely the SDN-DDoS2025. This dataset incorporates new SDN-relevant attributes derived from OpenFlow command and response interactions.

Most DDoS-SDN literature studies relied on a standardized SDN architecture comprising Mininet, a centralized controller (commonly Ryu), OpenFlow v1.3 switches and isolated attack and victim networks. The proposed architecture adheres to this model to ensure consistency with existing literature while enabling enhanced SDN-aware feature extraction [21].

The following steps outline the methodology for generating the SDN-augmented CSV dataset, as shown in Figure 3:

(1) Establish SDN Virtual Environment: The experimental setup commenced with the installation of Ubuntu 22.04.3 LTS version.

(2) Install Mininet and Ryu controller: Mininet, an advanced network emulator, was deployed to simulate a virtual SDN environment. It enables the creation of virtual hosts, switches, controllers and links within a single machine, allowing testing of network configurations and protocols without requiring physical infrastructure [22]. In addition, the Ryu controller, a widely adopted, open-source SDN controller developed in Python, was integrated into the environment. Ryu provides a dynamic platform for developing SDN applications and managing network flows and policies. It supports the Open-Flow protocol along with other southbound interfaces, enabling direct communication with network devices like switches and routers for real-time configuration [23].

(3) Construction of the SDN network: In the third step, an SDN topology was implemented to replicate the architectural structure of the CIC-DDoS2019 dataset, as shown in Figure 4. Two isolated virtual networks were designed: the *Attack Network* and the *Victim Network*. The Victim-Network comprises multiple devices connected to the SDN controller via a switch, which facilitates the generation of benign traffic patterns on each host. The Attack-Network operates as a distinct infrastructure that launches different DDoS attacks. Moreover, OpenFlow 1.3 was selected for

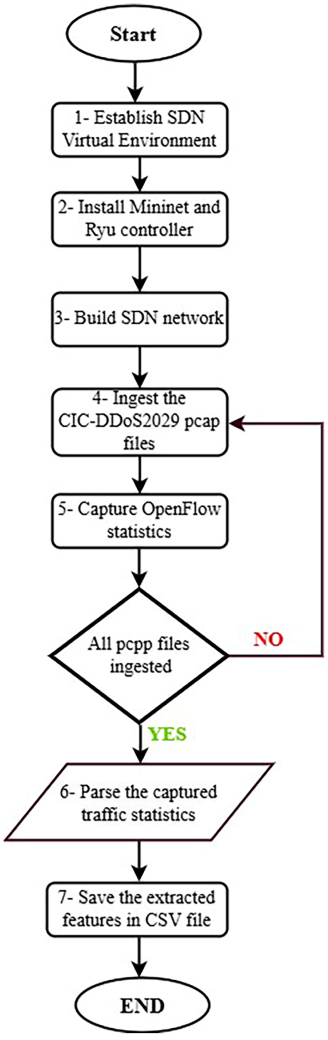

Figure 3. Proposed SDN-DDoS2025 generation flowchart. **Source**: Created by the authors

compatibility with the RYU controller, enabling seamless communication between hosts.

(4) Ingestion of CIC-DDoS2019 pcap files into the SDN network: The fourth step involved injecting the raw PCAP traffic files from the CIC-DDoS2019 dataset into the virtual SDN environment through designated network interfaces. Using Wireshark, a widely used open-source network protocol analyzer for capturing and inspecting network traffic [24].

(5) Capture of OpenFlow statistics via Ryu's REST API and Wireshark: In this phase, a developed Python script was employed to retrieve network statistics from the Ryu controller using its RESTful API interface. The Ryu Controller offers a REST API [25] that facilitates network programmability and automation by allowing external

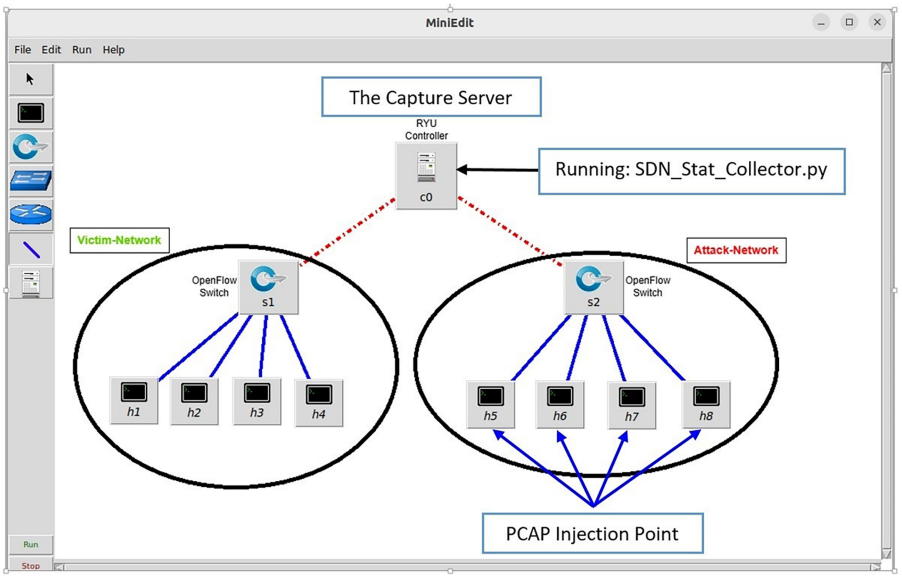

Figure 4. Proposed SDN-DDoS2025 network architecture. **Source**: Created by the authors

applications to interact with the SDN infrastructure. Flow-level, port-level and switch-level statistics, such as packet counts, byte counts and switch descriptions, were collected and processed.

(6) Parsing of the captured traffic statistics: the captured traffic data represents a time series of network statistics for each switch, port and flow. It includes a variety of high-level metrics like packet counts, byte counts, flow durations, entropy and CPU usage. These metrics and more were parsed systematically to extract relevant features that reflect the behavioral characteristics of normal and attack traffic, forming the basis for future research in anomaly detection.

(7) Saving the extracted features in CSV format: Finally, saving the new parsed features in a final CSV format that can be used later for training and evaluating DL models in the context of DDoS attack detection.

Algorithm 1 describes the proposed SDN-DDoS2025 network deployment steps, which form the basis of the experimental work methodology.

Algorithm 1. Setup Mininet and Ryu Controller

1: BEGIN

2: Step 1: Install Mininet

sudo apt update

sudo apt install mininet -y

3: Step 2: Set up Python virtual environment for Ryu

python3 -m venv ryu-env

source ryu-env/bin/activate

4: Step 3: Install Ryu and dependencies in the virtual environment

pip install ryu

pip install eventlet = = 0.30.2

5: INSTALL psutil for system and process utilities

#pip install psutil

6: Step 4: Start the Ryu controller

ryu-manager ryu.app.ofctl_rest

7: END

Algorithm 2 details the operation of the developed Python-based traffic monitor script, **SDN_ Stat_Collector.py**, that considered the primary data acquisition engine for the proposed dataset.

Algorithm 2. Network Statistics Collection

1: BEGIN

2: function get-switches

Send GET request to/stats/switches

Return List of switches in JSON format

3: end function

4: function get-port-stats(dpid)

Send GET request to/stats/port/{dpid}

Return Port statistics in JSON format

5: end function

6: function get-flow-stats(dpid)

Send GET request to/stats/flow/{dpid}

Return Flow statistics in JSON format

7: end function

8: function write-to-csv(data)

Append data as a row to network-stats.csv

9: end function

10: END

Existing DDoS and DDoS-SDN-based datasets differ significantly in terms of specificity and generalization. Traditional datasets like CIC-DDoS2019 provide extensive multi-vector DDoS traffic; however lack native SDN visibility, limiting their applicability to controller-aware detection models. Conversely, SDN-native datasets such as InSDN offer controller-level statistics but are restricted in attack diversity and scalability. Other datasets, including BoT-IoT and TON-IoT, are frequently adapted for SDN studies through emulation, providing realistic large-scale attack traffic but without direct OpenFlow interaction. The proposed SDN-DDoS2025 dataset bridges this gap by injecting realistic CIC-DDoS2019 traffic into a generalized SDN architecture and extracting flow-, port- and switch-level OpenFlow statistics

via the Ryu controller. This approach enhances dataset generalization across SDN-based DDoS detection models while preserving attack specificity, enabling fair comparison with both traditional and SDN-oriented datasets.

3.2 Exploratory SDN-DDoD2025 dataset analysis

The generated dataset SDN-DDoD2025 includes 39 features with both flow-level and port-level statistics, offering a dual-layered perspective of network behavior. The physical and logical health of the switches is monitored by port-level attributes like rx-packets, tx-bytes and error/drop counters, which show the immediate effects of volumetric flooding. At the same time, metrics at the flow level, such as duration-sec, packet-count and active-flows, monitor the actions of distinct traffic streams. The dataset incorporates system-level telemetry via CPU-util and statistical measures like entropy to improve the detection of sophisticated attacks, guaranteeing a multi-dimensional representation of both network performance and controller resource consumption. Out of these, 14 key attributes, including volumetric indicators, CPU utilization and entropy, have been identified as critical descriptors of DDoS activity, capturing the significant variations that occur when a network transitions from benign to attack states.

The dataset exhibits highly skewed distributions and extreme outliers in traffic volume, which are suggestive of high-intensity attack bursts, according to statistical analysis using histograms and boxplots. While high-traffic anomalies and spikes in CPU utilization (often exceeding 60%) highlight the resource stress brought on by DDoS events, volumetric features like rx-packets and rx-bytes exhibit clustering at lower ranges for typical traffic. On the other hand, during the feature selection process, some features, like priority, table-id and idle-timeout, stay almost constant, indicating that they are redundant candidates for removal.

The relationships between these characteristics are further clarified by correlation analysis using heatmap matrices, which reveals strong positive correlations between packet counts and byte volumes, as would be expected in volumetric attacks. Notably, the analysis shows a surprisingly negative correlation between CPU utilization and traffic surges, possibly as a result of the controller's incapacity to process signals during severe flooding, while time and aggregate traffic volume show a strong positive correlation. Entropy is a crucial additional metric for assessing traffic randomness, even though it exhibits weaker direct correlations. Together, these findings demonstrate that the SDN-DDoS2025 dataset contains the unique signatures required for deep learning detection model training.

Comprehensive EDA is provided in the Supplementary Data Section. This supplementary analysis employs visual and statistical techniques, including histograms, boxplots and heatmap correlation matrices to validate the dataset's quality and identify the most discriminative features. The EDA highlights critical indicators of DDoS activity, such as heavily skewed packet distributions and significant CPU volatility, while also identifying redundant constant-value features like priority and table-id that are candidates for exclusion in the subsequent deep learning modeling phase.

4. The proposed CNN-LSTM hybrid model for DDoS detection

CNN and LSTM models are prominent deep learning architectures extensively employed in time-series analysis and network traffic detection tasks [26]. A typical CNN architecture comprises three main layers: convolutional, pooling and fully connected layers. The convolutional layer initially extracts significant features from the input data through multiple filtering operations. Subsequently, the pooling layer reduces spatial dimensions by down-sampling the resultant feature maps, preserving essential information while reducing computational complexity. Finally, the fully connected layer interprets these pooled features, facilitating classification or regression tasks [27].

LSTM networks, a specialized variant of recurrent neural networks, excel in processing sequential data due to their capability to capture long-term dependencies and effectively mitigate the vanishing gradient issue [28, 29]. LSTMs demonstrate considerable efficacy by analyzing network traffic patterns and identifying temporal anomalies. The internal gating mechanisms of LSTM units strategically regulate information flow, enabling the model to selectively retain pertinent historical data and discard irrelevant inputs. Such mechanisms balance long-term memory retention and short-term adaptivity, making LSTM models highly suitable for dynamic and time-sensitive tasks, including network anomaly detection and DDoS analysis [30].

4.1 The proposed CNN-based DDoS detection model

This study suggested a hybrid CNN model built on the best architecture for spotting harmful network traffic attacks, e.g. DDoS, in a normalized dataset. The suggested approach transforms input characteristics into a 3D format appropriate for one-dimensional Convolution layers (Conv-1D). The suggested design is based on dense layers for classification, pooling layers, dropout for regularization and successive layers of convolutional filters. It has an Inception-style block that captures spatial hierarchies at several resolutions by processing inputs with different convolutional kernel sizes, effectively capturing spatial hierarchies at multiple resolutions. This architecture improves the capacity of the model to draw several characteristics from data from network traffic.

The Proposed CNN-based DDoS Detection Model Architecture is as follows:

(1) *Initial Convolutional Layers:* Two Conv-1D layers with 32 filters each and a kernel size equal to 3.

(2) *MaxPooling and Dropout* (50%) to reduce dimensions and prevent overfitting.

(3) *Inception Block:* Three branches of Conv-1D layers with filters of size 1, 3 and 5. Concatenated outputs capture patterns at different resolutions, followed by Max-Pooling and Dropout (30%).

(4) *Global Feature Extraction:* Global Average Pooling 1D to summarize spatial features across time steps.

(5) *Dense Layers:* Two fully connected layers (256 and 128 neurons) with ReLU activation and Batch Normalization for stability and a Sigmoid output layer for binary classification.

(6) *Optimization:* Adam optimizer with a learning rate of 0.001 and Binary Cross-entropy as the loss function.

(7) *Callbacks: ReduceLROnPlateau* to dynamically reduce the learning rate and *ModelCheckpoint* to save the best-performing model during training.

Figure 5 shows the CNN-based model design. This architecture guarantees optimal speed and robustness by means of local and global feature extraction, hence classifying DDoS traffic efficiently.

4.2 The proposed CNN-LSTM hybrid model for DDoS detection

The proposed CNN-LSTM hybrid model, as shown in Figure 6, is specifically designed for binary classification. The proposed model initially starts with the aforementioned CNN-based model, including the initial convolutional (Conv-1D) layer, then the architecture continues with its successive layers and the inception-inspired block. Additionally, an LSTM layer is added to mitigate overfitting. The LSTM layer is incorporated to model sequential dependencies and temporal relationships. Consequently, this architectural combination

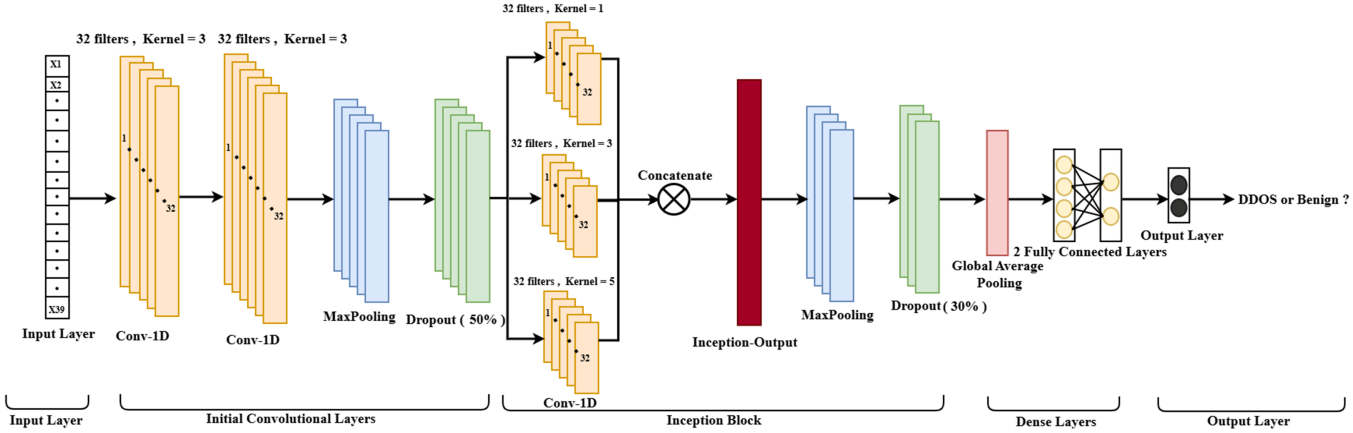

Figure 5. The architecture of the proposed CNN model. **Source**: Created by the authors

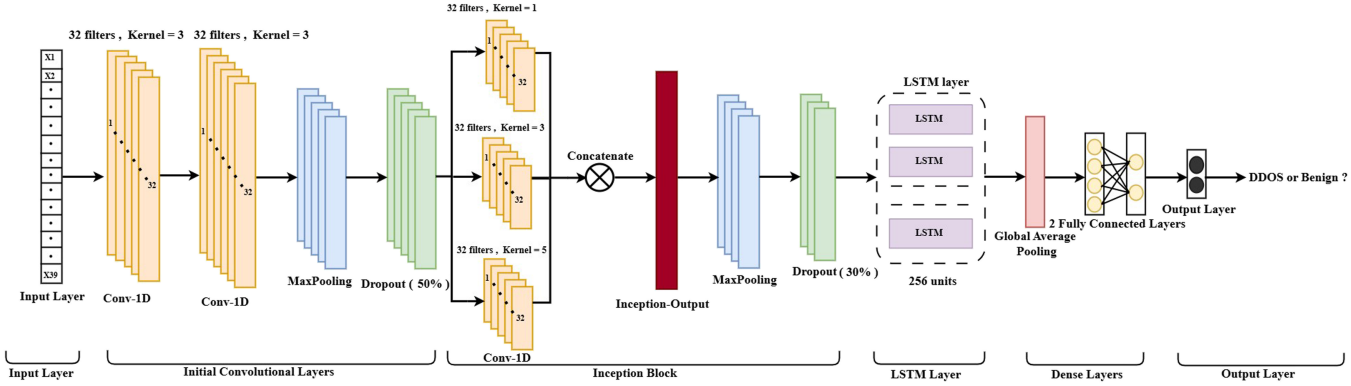

Figure 6. The architecture of the proposed CNN-LSTM hybrid model. **Source**: Created by the authors

strengthens the model's capability to extract rich, diverse features from sequential network traffic, significantly improving classification performance for DDoS detection tasks.

The proposed CNN-LSTM model architecture is as follows:

(1) *Input Layer*: Receives input data shaped for Conv-1D, where each feature is treated as a timestep.

(2) *Initial Convolutional Layers:*

- Two Conv-1D layers with 32 filters and kernel size 3.
- Batch normalization for stability.
- Max pooling for dimensionality reduction.
- Dropout with a 50% rate to prevent overfitting.

(3) *Inception Block:*

- Three branches with Conv-1D layers of varying kernel sizes (1, 3 and 5) for multi-scale feature extraction.
- The Concatenation layer combines these branches.
- Max pooling and (30%) dropout for regularization.

(4) *LSTM Layer:* A single-layer LSTM with 256 units for capturing temporal dependencies.

(5) *Dense Layers:* Two dense layers with 256 units, followed by batch normalization and an activation function.

(6) *Output Layer:* A single dense unit with sigmoid activation for binary classification.

The proposed CNN-LSTM hybrid model demonstrates excellent performance for sequential and time-series classification tasks, particularly in the context of SDN-based DDoS detection. Specifically, the Conv-1D layers effectively extract salient spatial and temporal features from network traffic data, capturing localized patterns, periodicities and correlations. The incorporation of inception blocks further enhances the model's ability to identify multiscale features indicative of traffic anomalies relative to normal network conditions. Complementing this, the LSTM layers excel in learning sequential dependencies and long-term temporal patterns inherent in network traffic, significantly aiding the detection of network anomalies and potential threats. To mitigate overfitting, the model integrates dropout layers and L2 regularization techniques into the architecture. Particularly, the Spatial Dropout 1D layer strategically positioned before the LSTM layer selectively eliminates entire feature maps, improving generalization capability when processing sparse or noisy data.

In conclusion, the proposed CNN-LSTM hybrid model combines LSTM's ability to model temporal dependencies with CNN's strength in spatial feature extraction, providing robust DDoS detection sensitive to both short-term fluctuations and long-term traffic trends.

5. Experimental results and analysis

5.1 Experimental work setup

The experimental work was conducted on a PC equipped with Windows 11(64-bit), an Intel Core i7 processor and 32 GB RAM. The proposed model was developed in Python 3.10.9 (64-bit) using Jupyter Notebook within an Anaconda environment. An SDN network environment was simulated using Mininet version 2.3.1b4 on VMware Workstation running Ubuntu 22.04.3, with Wireshark 3.6.2 employed for traffic capture. As shown, Table 2 illustrates the proposed CNN-based model training hyperparameters and Figure 7 illustrates the adopted hyperparameters in training the proposed CNN-LSTM hybrid model.

Table 2. Hyperparameters used in the CNN-based model

Hyper parameter	Value
Learning Rate	1×10^{-3}
Batch Size	64
Epochs	50
Dropout (CNN1)	0.5
Dropout (Inception Block)	0.3
Filters (Initial Conv-1D)	32
Kernel Size (Initial Conv-1D)	3
Filters (Inception Block)	32
Kernel Sizes (Inception Block)	1, 3, 5
Optimizer	Adam
Loss Function	Binary Cross entropy
Callback: ReduceLROnPlateau	Yes
Callback: ModelCheckpoint	Yes
Validation Split	20%

Parameter	Value
Conv-1D Layer	
Number of Filters	32
Activation Function	ReLU
Padding	Same
Regularization	L2 (0.001)
MaxPooling Pool Size	2
Inception Module	
Conv1D Filters	32
Conv1D Kernel Sizes	1, 3, 5
Concatenation	Yes
MaxPooling Pool Size	2
Dropout Rate	30%
LSTM Layer	
Number of Units	256
Return Sequences	False
Regularization	L2 (0.001)
Dense Layer	
Number of Units	256
Activation Function	ReLU
Regularization	L2 (0.001)
Batch Normalization	Yes
Output Layer	
Number of Units	1
Activation Function	Sigmoid
Training Hyperparameters	
Optimizer Type	Adam
Learning Rate	5e-4
Patience	20 epochs
Restore Best Weights	Yes
Batch Size	64
Epochs	200

Figure 7. The proposed CNN-LSTM hybrid model hyperparameters. **Source**: Created by the authors

5.2 Data preparation and preprocessing

Data preparation and preprocessing are critical phases in deep learning, significantly influencing model accuracy and reliability. The preprocessing steps included removing irrelevant features, null values and duplicate entries from the dataset, thereby reducing noise and inconsistencies to enhance the model's predictive capacity [16].

The following outlines the respective preprocessing steps conducted on both datasets, SDN-DDoS2025, CIC-DDoS2019 as follows [16–18]:

(1) *Data Loading:*

- Load the dataset from a CSV file.
- Display the initial shape of the Data Frame.

(2) *Data Cleaning:*

- Remove irrelevant or problematic columns.
- Identify and remove non-numeric columns (categorical or string data).
- Impute missing values (NaNs) in numerical columns using the median.
- Replace zero values with a small positive value to avoid division by zero or log transformation issues.

(3) *Further Data Cleaning and Feature Engineering:*

- Convert negative numbers to their absolute values.
- Remove constant features (features with a single unique value).
- Remove quasi-constant features (features with very low variance).
- Remove duplicated features.
- Remove highly correlated features.
- Apply one-hot encoding to categorical features.
- Save the preprocessed Data Frame to a new CSV file.

(4) *Feature Scaling:*

- Apply Min-Max scaling to bring features to the range [0, 1].
- Scale the data from [0, 1] to $[-1, 1]$.
- Combine scaling steps into a preprocessing pipeline.
- Apply the scaling pipeline to the feature data.
- Save the scaled Data Frame to a new CSV file.

(5) *Data Inspection and Diversity Metrics:*

- Display the shape of the processed Data Frame.
- Calculate and print the Shannon Diversity Index (H), Maximum Diversity (Hmax) and Pielou Index (J) to assess feature distribution diversity

5.3 Evaluation metrics

The following Evaluation metrics have been used in the conducted analysis:

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + FN + TN} \tag{1}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{2}$$

$$\text{Precision} = \frac{TP}{TP + FP} \tag{3}$$

$$\text{F1} - \text{Score} = 2\times\left(\frac{\text{Precision}\times\text{Recall}}{\text{Precision} + \text{Recall}}\right) \tag{4}$$

$$MCC = \frac{TP\cdot TN - FP\cdot FN}{\sqrt{P\cdot N\cdot TP_{sum}\cdot TN_{sum}}} \tag{5}$$

5.4 The evaluation of the proposed CNN-LSTM hybrid model

This section provides the achieved results and performance of the CNN-based model as well as the proposed CNN-LSTM hybrid model, both of which were trained and assessed using each of the CIC-DDOS2019 and the generated SDN-DDoS2025 datasets.

5.4.1 Results on the typical CIC-DDoS2019 dataset. The performance evaluation results represented in Table 3 show that the CNN-based model has state-of-the-art detection capabilities, with a near-perfect accuracy of 99.98% and consistent scores of 99.98% for precision, recall and F1. When compared to existing architectures, the model maintains superior reliability, significantly outperforming CNN-WRS, which had an accuracy of 95.76% and a significantly lower Matthews Correlation Coefficient (MCC) of 29.64%, while matching the top-tier performance of Cybernet and CNN-BRS (both 99.99%). Notably, the proposed model's MCC of 99.70% confirms its ability to correctly classify both benign and malicious traffic, as compared to Bays-CNN's 98.59%. These results highlight that our CNN model is a good candidate for more research and use in DDoS detection, hence requiring a priority emphasis in the following evaluations of our newly developed SDN dataset.

Furthermore, Table 4 demonstrates the superior capabilities of the proposed CNN-LSTM hybrid model against the other four benchmark models on the CIC-DDoS2019 dataset. It is worth noting that the proposed CNN-LSTM hybrid model achieves 99.98% accuracy, recall and precision of 99.98% for both, and an F1-score and MCC of 99.98% and 99.70%, respectively. CNN-BRS and Cybernet outperform the other models in some metrics, particularly with a perfect recall and precision 99.99% for each and a slightly higher MCC (99.93% for CNN-BRS and 99.98% for Cybernet). However, the overall performance of the CNN-LSTM model remains exceptional. In contrast, the Bays-CNN exhibits solid performance, with 99.90% accuracy and decent values for precision, F1-score and MCC, but it is slightly behind the proposed CNN-LSTM hybrid model in terms of recall, precision and F1-score. CNN-WRS performs the worst across all metrics. With 95.76% accuracy, a low precision of 95.87% and a poor MCC of 29.64%, CNN-WRS is not as effective as the other models for DDoS detection.

5.4.2 Results on the generated SDN-DDoS2025 dataset. Regarding the CNN-based model, Table 5 presents the performance evaluation conducted on the newly generated SDN-DDoS2025 dataset. The results indicate that the suggested CNN model performs better in

Table 3. CNN-based model results on CIC-DDoS2019 dataset

Metrics	CNN-BRS [16]	Bays-CNN [17]	CNN-WRS [18]	Cybernet [19]	CNN
Accuracy	99.99%	99.90%	95.76%	99.99%	*99.98%*
Recall	99.99%	98.90%	99.84%	99.99%	*99.98%*
Precision	99.99%	99.70%	95.87%	99.99%	*99.98%*
F1-Score	99.99%	99.29%	95.95%	99.99%	*99.98%*
MCC	99.93%	98.59%	29.64%	99.98%	*99.70%*

Table 4. Hybrid CNN-LSTM model results on the typical CIC-DDoS2019 dataset

Metrics	CNN-BRS [16]	Bays-CNN [17]	CNN-WRS [18]	Cybernet [19]	CNN-LSTM
Accuracy	99.99%	99.90%	95.76%	99.99%	*99.98%*
Recall	99.99%	98.90%	99.84%	99.99%	*99.98%*
Precision	99.99%	99.70%	95.87%	99.99%	*99.98%*
F1-Score	99.99%	99.29%	95.95%	99.99%	*99.98%*
MCC	99.93%	98.59%	29.64%	99.98%	*99.70%*

Table 5. Results of our proposed CNN on SDN-DDoS2025 dataset

Metrics	CNN-BRS [16]	Bays-CNN [17]	CNN-WRS [18]	Cybernet [19]	CNN
Accuracy	91.87%	88.26%	84.00%	93.14%	*96.11%*
Recall	91.86%	79.57%	93.00%	92.80%	*96.11%*
Precision	93.96%	84.21%	87.00%	92.50%	*96.46%*
F1-Score	92.25%	81.54%	90.00%	92.60%	*96.18%*
MCC	94.83%	63.61%	52.11%	83.18%	*89.79%*

terms of accuracy, precision, recall, F1-score and MCC versus benchmarks. This suggests that it is more efficient in spotting DDoS attacks with fewer false positives and negatives.

The proposed CNN model (96.11%) outperforms other benchmark models. The CNN-Based model is significantly better than CNN-BRS (91.87%) and Bays-CNN (88.26%). CNN-WRS (84%) has the lowest test accuracy, and Cybernet (93.14%) is slightly better than CNN-BRS but still falls behind CNN models. The CNN model (96.11%) has a higher recall, which indicates that it is the best model for identifying true positives among other models.

While Bays-CNN (79.57%) displays much poorer performance than others, Cybernet (92.80%) also does well. With high precision, the suggested CNN model (96.46%) is effective in accurately forecasting positive classifications. With lower precision scores, Bays-CNN (84.21%) and CNN-WRS (87%) indicate they might forecast more false positive results. With an F1-Score of 98.03%, with a lower F1-Score, Bays-CNN (81.54%) and CNN-WRS (90%) show less balanced performance.

Moreover, Table 6 presents the performance evaluation conducted on the newly generated SDN-DDoS2025 dataset regarding the proposed CNN-LSTM hybrid model, highlighting that the proposed hybrid model consistently outperforms all the benchmarks in all key metrics, including accuracy, precision, recall, F1-score and MCC. The CNN-LSTM model outperforms all the benchmarks with the highest accuracy (98.02%), markedly exceeding CNN-BRS (91.87%), Bays-CNN (88.26%), CNN-WRS (84%) and Cybernet 93.14%). CNN-LSTM exhibits superior recall that improves true positive detection capability (98.01%) compared to Cybernet's (92.80%) and particularly Bays-CNN (79.57%). Regarding precision, CNN-LSTM attains an impressive score (98.04%), significantly outperforming Bays-CNN (84.21%) and CNN-WRS (87%), both of which exhibit higher false-positive rates. The F1-score, which reflects a balanced measure of precision and recall, further emphasizes CNN-LSTM's robust performance (98.03), contrasting notably with CNN-WRS (90%) and Bays-CNN (81.54%). Additionally, CNN-LSTM achieves a strong MCC value (94.49%), effectively accounting for false positives and negatives, whereas CNN-WRS (52.11%) and Bays-CNN (63.61%) present considerably weaker correlation metrics.

Table 6. Results on the typical SDN-DDoS2025 dataset

Metrics	CNN-BRS [16]	Bays-CNN [17]	CNN-WRS [18]	Cybernet [19]	CNN-LSTM
Accuracy	91.87%	88.26%	84.00%	93.14%	*98.02%*
Recall	91.86%	79.57%	93.00%	92.80%	*98.01%*
Precision	93.96%	84.21%	87.00%	92.50%	*98.04%*
F1-Score	92.25%	81.54%	90.00%	92.60%	*98.03%*
MCC	94.83%	63.61%	52.11%	83.18%	*94.49%*

Overall, the proposed CNN-LSTM model delivers superior performance improvements across all metrics, demonstrating its efficacy compared to other benchmark models, making it the preferred choice for accurate and reliable DDoS detection within an SDN environment.

5.5 Results analysis and discussion

This section demonstrates a detailed comparative analysis, including cross-evaluation experiments between the CIC-DDoS2019 and SDN-DDoS2025 datasets, further validating the robustness and generalization of the CNN-LSTM model.

In this study, the CNN-LSTM model was purposefully designed and presented after the CNN model as a natural architectural extension, rather than as an independent contribution. The CNN model serves as a baseline for capturing spatial characteristics of network traffic, while the CNN-LSTM model expands on this capability by including temporal dependency modeling. Presenting both models within the same study allows for a controlled and fair comparison using identical datasets, preprocessing steps and experimental conditions.

The comparative experimental results show that the proposed CNN-LSTM model and the CNN model both obtain a 99.98% accuracy on the CIC-DDoS2019 dataset, which is 4.20% higher than CNN-WRS and 0.01% lower than CNN-BRS and Cybernet (all of which achieve 99.99%). This demonstrates that both the CNN-LSTM and CNN models perform exceptionally well on the CIC-DDoS2019 dataset, achieving near-perfect accuracy.

The comparative experimental results show that the proposed CNN-LSTM model obtains a 99.98% accuracy on the CIC-DDoS2019 dataset, surpassing CNN-WRS by 4.20% and closely matching CNN-BRS and Cybernet (99.99%). More significantly, the proposed CNN-LSTM model achieves 98.02% accuracy on the SDN-DDoS2025 dataset, outperforming CNN-BRS, Bays-CNN, CNN-WRS and Cybernet by margins of 6.69%, 11.04%, 16.69% and 5.24%, respectively.

Figure 8 presents the improvements in classification accuracy of the proposed CNN, CNN-LSTM hybrid models relative to benchmarks on both CIC-DDoS2019 and SDN-DDoS2025 datasets. On the CIC-DDoS2019 dataset, both the proposed CNN-LSTM and CNN models achieve a high detection accuracy, surpassing CNN-WRS and Bays-CNN by 4.22% and 0.08%, respectively, while being slightly lower than Cybernet and CNN-BRS by 0.01%. On the SDN-DDoS2025 dataset, the CNN-LSTM model demonstrates a significant improvement, outperforming CNN-BRS by 6.69%, Bays-CNN by 11.04%, CNN-WRS by 16.69% and Cybernet by 5.24%. The CNN model on the SDN-DDoS2025 dataset also shows notable improvements, though less than the CNN-LSTM, exceeding CNN-BRS by 4.24%, Bays-CNN by 7.85%, CNN-WRS by 12.11% and Cybernet by 2.97%.

Figure 9 demonstrates the comparative improvements in detection accuracy achieved by the proposed CNN-LSTM hybrid model relative to benchmark approaches on both the CIC-DDoS2019 and SDN-DDoS2025 datasets. For the CIC-DDoS2019 dataset, the proposed CNN-LSTM hybrid model attains a remarkably high accuracy, surpassing CNN-WRS and Bays-CNN by margins of 4.41% and 0.08%, respectively, and closely aligning with the

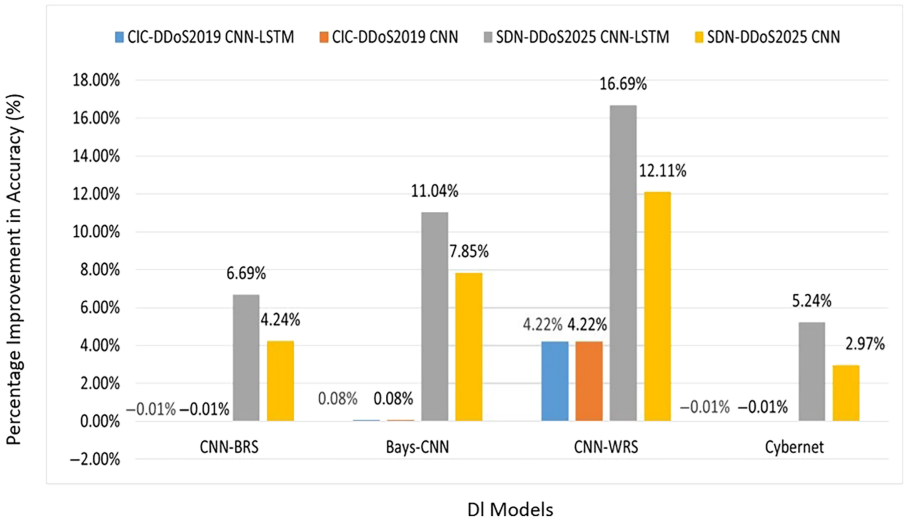

Figure 8. Percentage improvements in classification accuracy of the proposed CNN, CNN-LSTM models relative to benchmark architectures across the CIC-DDoS2019 and SDN-DDoS2025 datasets

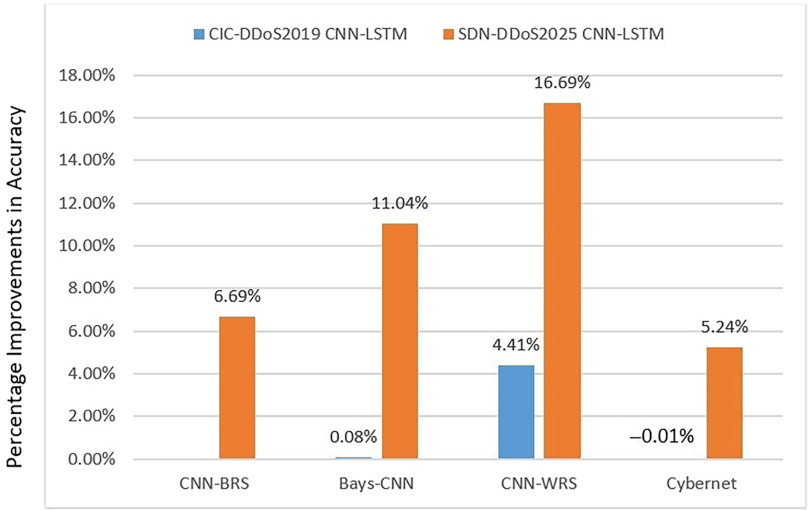

Figure 9. Improvements in the detection accuracy of the proposed CNN-LSTM hybrid model relative to benchmarks on CIC-DDoS2019 and SDN-DDoS2025 datasets. **Source**: Created by the authors

highest-performing models, Cybernet and CNN-BRS, with a marginal difference of only 0.01%. Notably, the CNN-LSTM hybrid model exhibits even more substantial performance gains on the SDN-DDoS2025 dataset, outperforming CNN-BRS by 6.69%, Bays-CNN by 11.04%, CNN-WRS by 16.69% and Cybernet by 5.24%

Additionally, cross-evaluation experiments reveal the robustness of publicly available DL models against the newly introduced SDN-DDoS2025 dataset. As depicted in Figure 10, the CNN-LSTM model exhibits minimal accuracy degradation (1.96%) when transitioning from

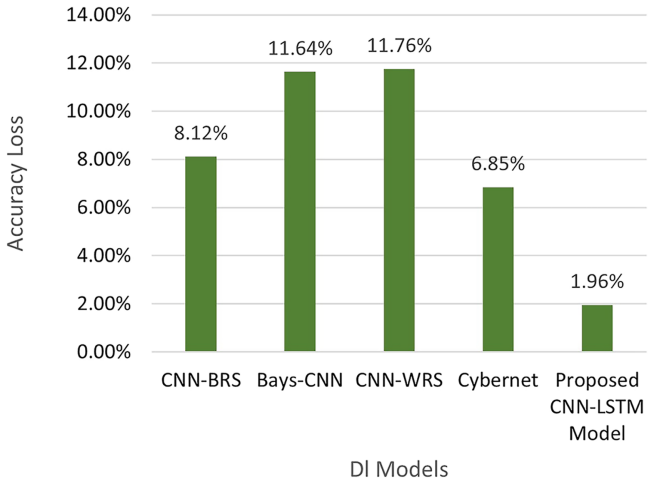

Figure 10. Comparative decrease in classification accuracy across DL models on SDN-DDoS2025 relative to CIC-DDoS2019. **Source**: Created by the authors

CIC-DDoS2019 to SDN-DDoS2025, signifying enhanced stability and robustness compared to benchmark models. Conversely, Bays-CNN (11.64%) and CNN-WRS (11.76%) experience significant performance deterioration, suggesting vulnerability to overfitting or inadequate generalization, while Cybernet maintains consistent performance across both datasets. These results underscore the superior resilience and generalization capacity of the proposed CNN-LSTM hybrid approach.

6. Conclusion

This research highlights the critical need for developing a realistic dataset tailored for DDoS detection in an SDN environment. It presents the methodology for generating a novel DDoS attack dataset in SDN named SDN-DDoS2025 based on the CICDDoS2019 dataset. Furthermore, a comprehensive EDA is conducted on SDN-DDoS2025 to characterize feature correlations, skewness, central tendency and outlier analysis.

Additionally, the research proposes an enhanced hybrid deep learning model that combines CNN and LSTM for improved DDoS detection accuracy. The proposed CNN-LSTM model achieves an impressive accuracy of 99.98% and 98.02% on CIC-DDoS2019 and SDN-DDoS2025, respectively. On the CIC-DDoS2019 dataset, the proposed model performs almost as well as the best benchmark in accuracy on SDN-DDoS2025.

The notable slight accuracy reduction of the proposed CNN-LSTM hybrid model (1.94%) compared to the accuracy drops in the benchmarks (ranging from 6.84% to 11.55%), demonstrates its superior robustness on both CIC-DDoS2019 and SDN-DDoS2025 datasets.

The comprehensive EDA of the SDN-DDoS2025 dataset also reveals important challenges and opportunities for future research, particularly in data preprocessing strategies, advanced feature engineering and the development of more effective DDoS detection methodologies.

Plain language summary

Our dataset and model help protect SDN networks from DDoS attacks, ensuring stable and secure internet services.

Ethics statement
This study involved no human or animal subjects and complies with all relevant ethical standards.

Data access statement
The dataset generated and analyzed during the current study is available at: https://github.com/Mohamed-A-Elshafey/SDNDDoS2025.

Supplementary material
The supplementary material for this article can be found online.

References

1. Liu F, Kibalya G, Kumar SVNS, Zhang P. Challenges of traditional networks and development of programmable networks. Cham: Springer; 2022. p. 37-61. doi: 10.1007/978-3-030-89328-6_3.
2. Gupta N, Maashi M, Tanwar S, Badotra S, Aljebreen M, Bharany S. A comparative study of software defined networking controllers using Mininet. Electronics. 2022; 11(17): 2715. doi: 10.3390/electronics11172715.
3. Fan C, Kaliyamurthy NM, Chen S, Jiang H, Zhou Y, Campbell C. Detection of DDoS attacks in software defined networking using entropy. Appl Sci. 2022; 12(1): 370. doi: 10.3390/app12010370.
4. Malliga S, Nandhini PS, Kogilavani SV. A comprehensive review of deep learning techniques for the detection of (distributed) denial of service attacks. Inform Technol Control. 2022; 51(1): 180-215. doi: 10.5755/j01.itc.
5. Clinton UB, Hoque N, Robindro Singh K. Classification of DDoS attack traffic on SDN network environment using deep learning. Cybersecurity. 2024; 7(23): 23. doi: 10.1186/s42400-024-00219-7.
6. Moustafa N, Slay J. UNSW-NB15: a comprehensive data set for network intrusion detection systems. In: Proceedings of the 2015 Military Communications and Information Systems Conference (MilCIS), 2015 Nov 3-5, Canberra. IEEE; 2015. p. 1-6.
7. Sharafaldin I, Lashkari AH, Ghorbani AA. Toward generating a new intrusion detection dataset and intrusion traffic characterization. In: International Conference on Information Systems Security and Privacy; 2018. Available from: https://api.semanticscholar.org/CorpusID:4707749
8. Sharafaldin I, Lashkari AH, Sahib I, Ghorbani A. Developing realistic distributed denial of service (DDoS) attack dataset and taxonomy. In: Proceedings of the 2nd International Conference on Computing, Communication, and Security (CCS'19), 2019 Oct 10-12, New York, NY. ACM; 2019. p. 1-8.
9. Buzzio-Garca J, Vergara J, Rios-Guiral S, Garzn C, Gutierrez S, Botero JF, Quiroz-Arroyo JL, Perez-Diaz JA. SDNFlow dataset. 2023. doi: 10.21227/40v2-hh58.
10. Elsayed M, Le-Khac NA, Jurcut AD. InSDN: a novel intrusion dataset for software-defined networks. IEEE Access. 2020; 8: 165363-84.
11. Koroniotis N, Moustafa N, Sitnikova E, Turnbull B. Towards the development of a realistic botnet dataset in the Internet of Things for network forensic analytics: BoT-IoT dataset. Future Generation Comput Syst. 2019; 100: 779-96. doi: 10.1016/j.future.2019.05.041.
12. Moustafa N, Turnbull B, Choo KKR. An ensemble intrusion detection technique based on statistical flow features for protecting internet of things networks: TON-IoT dataset. IEEE Internet Things J. 2019; 6(3): 4815-30. doi: 10.1109/jiot.2018.2871719.
13. Kim Y, Hakak S, Ghorbani A. DDoS attack dataset (CICEV2023) against EV authentication in charging infrastructure. In: 2023 20th Annual International Conference on Privacy, Security and Trust (PST). Los Alamitos: IEEE Computer Society; 2023. p. 1-9. doi: 10.1109/PST58708.2023.10320202.

14. Karnani S, Agrawal N, Kumar R. A comprehensive survey on low-rate and high-rate DDoS defense approaches in SDN: taxonomy, research challenges, and opportunities. Multimed Tools Appl. 2024; 83(12): 35253-306. doi: 10.1007/s11042-023-16781-0.
15. Elubeyd H, Yiltas-Kaplan D. Hybrid deep learning approach for automatic DoS/DDoS attacks detection in software-defined networks. Appl Sci. 2023; 13(6): 3828. doi: 10.3390/app13063828. Available from: https://www.mdpi.com/2076-3417/13/6/3828
16. Najar AA, Naik SM. Cyber-secure SDN: a CNN-based approach for efficient detection and mitigation of DDoS attacks. Comput Sec. 2024; 139: 103716.
17. AlSaleh I, Al-Samawi A, Nissirat L. Novel machine learning approach for DDoS cloud detection: Bayesian-based CNN and data fusion enhancements. Sensors. 2024; 24(5): 1418. doi: 10.3390/s24051418. Available from: https://www.mdpi.com/1424-8220/24/5/1418
18. Najar A, Naik SM. A robust DDoS intrusion detection system using convolutional neural network. Comput Electr Eng. 2024; 117: 104698.
19. Azar ASMB. Cybernet model: a new deep learning model for cyber DDoS attacks detection and recognition. Comput Mater Cont. 2024; 78(1): 1275-95. Available from: http://www.techscience.com/cmc/v78n1/55407
20. Srivastava D. An introduction to data visualization tools and techniques in various domains. Int J Comput Trends Technol. 2023; 71(4): 125-30. doi: 10.14445/22312803/ijctt-v71i4p116.
21. Mousavi SM, St-Hilaire M. Early detection of DDoS attacks against SDN controllers. J Netw Comp Appl. 2018; 113: 64-79.
22. Mininet Team Mininet: an instant virtual network on your laptop; 2024. Available from: https://mininet.org/
23. Ryu Team Ryu: a component-based software defined networking framework; 2024. Available from: https://ryu-sdn.org/
24. Wireshark Team Wireshark: the world's premier network protocol analyzer; 2024. Available from: https://www.wireshark.org/
25. GitHub GitHub REST API documentation; 2022. Available from: https://docs.github.com/en/rest?apiVersion=2022-11-28
26. Setitra MA, Fan M, Agbley BLY, Bensalem ZEA. Optimized MLP-CNN model to enhance detecting DDoS attacks in SDN environment. Network. 2023; 3(4): 538-62. doi: 10.3390/network3040024. Available from: https://www.mdpi.com/2673-8732/3/4/24
27. El-Gabri AR, Aly HA, Ghoniemy TS, Elshafey MA. DLRA-net: deep local residual attention network with contextual refinement for spectral super-resolution. Int J Comput Vis. 2024; 133(4): 1-33. doi: 10.1007/s11263-024-02238-w.
28. Aydn H, Orman Z, Aydn MA. A long short-term memory (LSTM)-based distributed denial of service (DDoS) detection and defense system design in public cloud network environment. Comput Sec. 2022; 118: 102725. Available from: https://www.sciencedirect.com/science/article/pii/S0167404822001201
29. Badr MA, Elrewainy AF, Elshafey MAT. Hybrid spatial-spectral autoencoder models for lossy satellite image compression. J Aerosp Inf Syst. 2025; 22(5): 336-57. doi: 10.2514/1.I011445.
30. Bashaiwth A, Binsalleeh H, AsSadhan B. An explanation of the LSTM model used for DDoS attacks classification. Appl Sci. 2023; 13(15): 8820. doi: 10.3390/app13158820. Available from: https://www.mdpi.com/2076-3417/13/15/8820

Corresponding author
Khaled Metwally can be contacted at: k.metwally@mtc.edu.eg