International Journal of Computer Information Systems and Industrial Management Applications
ISSN 2150-7988 Volume 18(21s) 2026 pp. 977-1002
https://doi.org/10.70917/ijcisim-2026-5374
Article

# A Systematic Review of Deep Learning Models for DDoS Attack Detection in Cybersecurity

**Divyashree S1*, Dr. Lakshmi B N[2]**

[1]Research Scholar, BMS Institute of Technology and Management, Bengaluru-560119, Assistant Professor , Department of Computer Science and Engineering, East Point college of Engineering and Technology , Bengaluru- 560049
[2]Associate Professor, Department of Computer Science and Engineering, BMS Institute of Technology and Management, Bangalore-560119

**Abstract:** Distributed Denial of Service (DDoS) attacks are still one of the biggest threats to cybersecurity. They overload network resources and cause problems for businesses, governments, and important infrastructure. Conventional detection systems frequently inadequately respond to the dynamic and extensive characteristics of these attacks. Deep learning (DL) has become a powerful method in recent years because it can automatically learn complex traffic patterns, adjust to new attack strategies, and cut down on false positives. This systematic literature review (SLR) evaluates the role of deep learning models in DDoS attack detection by synthesizing 97 peer-reviewed studies published from 2015 to 2024. The review looks at popular deep learning (DL) methods like Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), Long Short-Term Memory (LSTM), Autoencoders, Generative Adversarial Networks (GANs), and hybrid models. It also looks at benchmark datasets (like CICIDS2017, CICDDoS2019, and NSL-KDD) and evaluation metrics like accuracy, precision, recall, F1-score, and detection rate. Key findings show that DL methods are better at detecting things than traditional methods, but they have a lot of problems, such as dataset imbalance, high computational cost, lack of real-time adaptability, and being vulnerable to attacks from other systems. The study delineates research deficiencies in the creation of lightweight, interpretable, and scalable models appropriate for IoT and edge computing environments. This review synthesizes existing knowledge, offering insights into the advantages and drawbacks of DL-based DDoS detection while delineating avenues for future research aimed at developing resilient, adaptive, and reliable cybersecurity systems.
**Keywords:** Distributed Denial of Service (DDoS), Cybersecurity, Deep Learning, Hybrid Models, Intrusion Detection Systems (IDS), IoT Security, Adversarial Attacks

## 1. INTRODUCTION

Distributed Denial of Service (DDoS) assaults flood network resources and disrupt online services, posing a cybersecurity risk [1]. Complex and massive attacks make standard detection technologies less successful at spotting sophisticated traffic patterns. To overcome this limitation, researchers have increasingly employed deep learning to analyze large network data and find subtle irregularities. Deep learning models increase the accuracy of identifying fraudulent communications and adapt to changing attack methods. This systematic review weighs the pros and cons of deep learning for DDoS attack detection and suggests ways to improve cybersecurity resilience.

Cybersecurity is a critical issue for enterprises globally, as cyber assailants consistently aim at company data and information technology (IT) assets to generate profit or gain a geopolitical advantage. Recent projections show that the global cost of cybercrime would reach over USD 10.5 trillion annually by 2025, up from USD 3 trillion in 2015 [2]. Furthermore, the number of Internet of Things (IoT) devices is projected to exceed 30 billion by 2030, which dramatically increases the cyber-attack surface and exacerbates the difficulties of cybersecurity [3]. Cybersecurity is the protection of individual or corporate computer data from unwanted access [4]. A cyber-attack is an endeavor to get illegal access, potentially including the theft of private data, intellectual property, proprietary business strategies, and/or the interruption of essential IT systems [5]. Criminal syndicates and state-sponsored paramilitary cyber entities

have begun employing cyber-attacks as a tactical approach, resulting in the emergence of advanced persistent threats (APTs) that are increasingly challenging for organizations to mitigate, even with established cybersecurity frameworks [6],[7]. Cyber threat intelligence (CTI) has arisen as a viable tool for enterprises to manage the growing volume and intricacy of security incidents [8]. CTI denotes the anticipatory recognition and examination of cyber hazards. Nevertheless, subscribing to many threat intelligence sources may result in information overload. A Threat Intelligence-Sharing Platform (TISP) may convert cyber threat information into actionable intelligence, which can be integrated into many technologies to enhance incident response. Information security organizations and the ecosystem now provide TISP solutions in two categories: content aggregation, which supplies multiple threat data feeds, and threat intelligence management, which derives commercial value from the acquired data [9].

Multiple devices assault a server or network in a DDoS attack. This attack floods a server or network with fake requests to disrupt its flow. Legitimate traffic is disrupted because network resources are overloaded. These assaults use networks of internet-connected devices, including PCs and IoT devices, infected with malicious software and vulnerable to remote manipulation. These gadgets are bots. Botnets, or hacked computers, are the main source of DDoS assaults, making them effective. A botnet allows the attacker to control the assault by delivering remote orders to each bot. While targeting the victim's server, each bot in the botnet searches its IPs, which may overload the network and interrupt normal traffic. Each bot is a genuine internet device, making assaults and legitimate traffic difficult to distinguish. As said, DDoS attackers use botnets to launch their assaults, therefore its architecture includes an attacker, a botnet, and the target network or server. Botnet management creates different architectures. According to [10], DDoS attack architecture is centralized or decentralized. Figure 1 depicts these architectures. Centralized is shown in Fig. 1a. An attacker, target server, botnet, and C&C system are involved. C&C systems are called handlers in certain research [11],[12],[13]. This design prevents botnet machines from communicating. Every bot is coordinated by a C&C system. The botnet is operated by directly commanding these bots. In a decentralized design (Fig. 1b), bots form a P2P network. A bot receives an attack query to launch a DDoS assault [14].

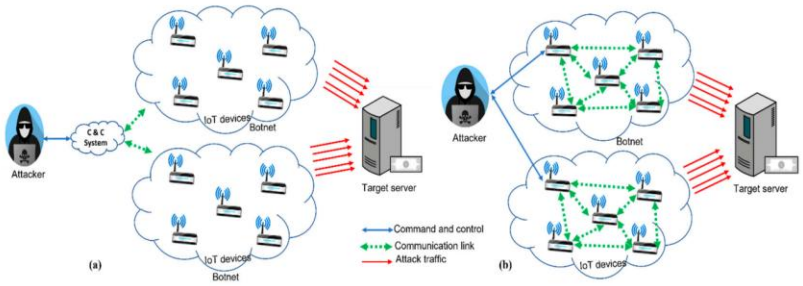


**Fig. 1.** Architecture of a DDoS attack: (a) centralized, (b) decentralized

The increasing scale of DDoS activity further emphasizes the seriousness of this threat. The frequency and intensity of global DDoS attacks are consistently increasing. According to NETSCOUT, there were approximately 10.09 million attacks in 2020 and 9.75 million attacks in 2021 [15], [16]. This trend continued in 2023 and 7.9 million attacks were reported in the first half of 2023, representing a 31% increase from the previous year [17]. Cloudflare shows a big spike in attacks in 2024 and 2025, with 21.3 million and 47.1 million mitigated attacks, respectively, and record-setting volumetric attacks of 5.6 Tbps and 31.4 Tbps [18], [19]. Although these statistics represent observations from NETSCOUT and Cloudflare rather than the complete global DDoS landscape, they demonstrate the increasing frequency, scale, and complexity of modern DDoS threats, particularly those associated with IoT-based botnets, as summarized in Table 1.

**Table 1.** Reported DDoS Attack Statistics from 2020–2025

| Year | Reported DDoS attack activity | Notable observation |
|---|---|---|
| 2020 | 10.09 million attacks | More than 10 million attacks recorded globally |

| | | |
|---|---|---|
| 2021 | 9.75 million attacks | Approximately 14% above pre-pandemic levels |
| 2023 | 7.9 million attacks | 31% year-over-year increase; approximately 44,000 attacks/day |
| 2024 | 21.3 million attacks | 53% increase over 2023; peak attack of 5.6 Tbps |
| 2025 | 47.1 million attacks | 121% increase over 2024; peak attack of 31.4 Tbps lasting 35 seconds |

The swift development of cyberthreats has made the use of increasingly advanced detection and prevention methods necessary. Among these, deep learning has become a potent cybersecurity tool because of its capacity to automatically identify intricate patterns in vast amounts of data. Deep learning models like Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and Long Short-Term Memory (LSTM) networks can learn hierarchical representations of network traffic, which allows them to detect subtle anomalies and novel attack patterns with high accuracy [20], [21]. This is in contrast to traditional machine learning methods that require manual feature engineering. Deep learning frequently outperforms traditional methods in terms of detection rate and false-positive reduction, as evidenced by numerous studies that show its efficacy in detecting malware, phishing attempts, DDoS attacks, and other cyberthreats [22]. Deep learning models can also adjust to changing attack tactics, which makes them appropriate for real-time cybersecurity applications. The incorporation of deep learning methods into cybersecurity frameworks offers a promising path for improving threat intelligence and automated defenses as network environments become more complex.

The use of deep learning models for cybersecurity DDoS attack detection is the only topic of this systematic review. Through synthesis and critical evaluation of existing research, the study seeks to offer a thorough understanding of contemporary approaches, architectures, and implementation strategies. In addition to providing a basis for further research and directing the creation of automated, adaptive, and efficient detection systems that can handle ever-more-complex DDoS attacks, it draws attention to the advantages, disadvantages, and practical applicability of these techniques.

The objectives of this systematic review are as follows:

- To review and synthesize existing research on the application of deep learning models for Distributed Denial of Service (DDoS) attack detection in cybersecurity.
- To categorize and compare different deep learning architectures (e.g., CNN, RNN, LSTM, GANs, autoencoders, hybrid models) used for DDoS detection.
- To examine the datasets (benchmark, real-world, synthetic) commonly employed for training and evaluating deep learning-based DDoS detection models.
- To analyze performance metrics such as accuracy, precision, recall, F1-score, detection rate, and false positive rate used in evaluating detection effectiveness.
- To identify challenges and limitations in applying deep learning for real-time DDoS detection, such as dataset imbalance, computational cost, and adversarial vulnerabilities.

The remaining part of this review is organized as follows: The systematic literature review (SLR) methodology is explained in Section 2, along with the review protocol, search strategy, and inclusion/exclusion criteria; the results are presented in Section 3, which highlighted the deep learning (DL) architectures, datasets, and performance metrics used for DDoS attack detection; the findings are summarized in Section 4, which also discusses current research, ML/DL techniques, hybrid approaches, datasets, performance evaluation, and challenges; and the review is concluded and future research directions are explored in Section 5.

## 2. Methodology

This section explains the method used for this systematic literature review. It goes into detail about the review protocol, the research questions that guided the study, the search strategy, the eligibility criteria, and the data extraction procedures used to look at how deep learning models are used to find DDoS attacks in cybersecurity.

## *2.1 Protocol for Systematic Literature Review (SLR)*

The present research uses a systematic literature review (SLR) methodology to examine the use of deep learning (DL) models for cybersecurity Distributed Denial of Service (DDoS) threat detection. In order to guarantee openness, consistency, and dependability, the review follows accepted SLR principles (Kitchenham & Charters, 2007; PRISMA, 2020) and synthesizes research published between 2016 and 2024. Very few evaluations have focused only on deep learning-based models for DDoS detection, despite earlier surveys looking at intrusion detection systems (IDS) and more general machine learning (ML)-based cybersecurity techniques. This study fills that gap by focusing on DL architectures, including Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), Long Short-Term Memory (LSTM), Autoencoders, Generative Adversarial Networks (GANs), and hybrid frameworks. It also offers an organized synthesis of their efficacy.

The study also points out recurring problems in the literature, such as imbalanced datasets, a lack of real-time datasets, computational expense, problems with scalability in IoT systems, and susceptibility to hostile assaults. The next sections describe the review's methodology, which is graphically shown in the flow diagram (Fig. 2) and includes research questions, a search strategy, inclusion and exclusion criteria, quality evaluation, and data synthesis techniques.

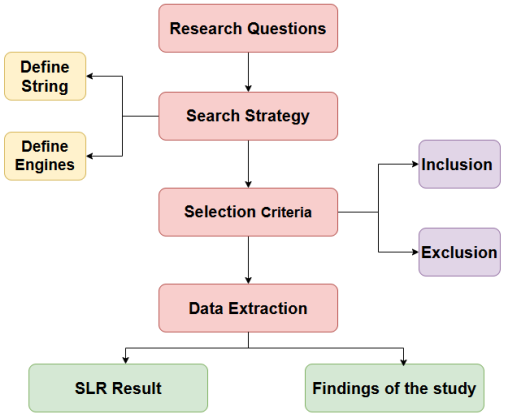


**Fig. 2.** SLR protocol overview

## *2.2 Review Research Questions*

The following research questions (Table 2) are formulated to lead this study by explicitly targeting the aims of comprehensively evaluating deep learning methodologies for DDoS attack detection. They concentrate on pinpointing the most prevalent deep learning approaches, scrutinizing the function of hybrid models, assessing frequently used datasets, evaluating performance indicators, and emphasizing significant obstacles and research deficiencies within the domain. These questions provide an organized way to look at what we know currently and help us decide where to go with future cybersecurity research.

**Table 2.** Research questions

| RQ No. | Research Question | Motivation |
|---|---|---|

| | | |
|---|---|---|
| RQ1 | What deep learning techniques are currently being used for DDoS attack detection? | To identify the most effective and widely adopted models (CNN, RNN, LSTM, hybrid, etc.). |
| RQ2 | How are hybrid approaches improving DDoS detection compared to single deep learning models? | To understand whether combining ML/DL techniques enhances accuracy and resilience. |
| RQ3 | What datasets are commonly used to train and evaluate deep learning-based DDoS detection systems? | To analyze dataset availability, limitations, and their impact on model reliability. |
| RQ4 | What performance metrics are used to evaluate the effectiveness of deep learning models in detecting DDoS attacks? | To determine consistent evaluation standards for fair model comparison. |
| RQ5 | What challenges and research gaps exist in applying deep learning for DDoS detection? | To highlight limitations such as dataset imbalance, high computational cost, and lack of real-time applicability. |

## *2.3 Search Strategy*

Systematic searches were done across key academic databases such IEEE Xplore, SpringerLink, ScienceDirect, ACM Digital Library, Wiley Online Library, Web of Science, Scopus, and Google Scholar to make sure that the review was thorough and methodologically sound. These databases were selected due to their comprehensive coverage of computer science, cybersecurity, and artificial intelligence research, rendering them especially pertinent to investigations into deep learning applications for DDoS detection.

The search approach included terms that had to do with deep learning, machine learning, cybersecurity, and finding DDoS attacks. This Boolean search string was used:

"Deep Learning," "Machine Learning," or "Hybrid Models" AND ("DDoS Detection," "Distributed Denial of Service," or "Intrusion Detection") AND ("IoT Security" OR "Cybersecurity" OR "Network Security")

The publishing period was determined from 2015 to 2024 to document the fast progress in deep learning methodologies and their use in cybersecurity during the last ten years. Strategically using Boolean operators helped cover as much ground as possible while keeping the review focused on the study's goals.

## *2.4 Criteria for Inclusion and Exclusion*

To ensure that the systematic review included only relevant and high-quality studies, specific inclusion and exclusion criteria were applied. Table 3 presents the criteria used for study selection.

**Table 3.** Inclusion and Exclusion Criteria for Study Selection

| **Criteria** | **Details** |
|---|---|
| Inclusion Criteria | Peer-reviewed journal articles, conference papers, and book chapters published between 2015–2024.<br>Studies focusing on deep learning models (CNN, RNN, LSTM, GANs, Autoencoders, Hybrid models) for DDoS attack detection.<br>Research using benchmark, real-world, or synthetic datasets relevant to cybersecurity.<br>Papers reporting quantitative performance metrics (accuracy, precision, recall, F1-score, detection rate, false positive rate).<br>Studies within the domain of cybersecurity, IoT, and network security. |

| | |
|---|---|
| Exclusion Criteria | Articles not written in English. |
| | Studies without use of deep learning or hybrid ML/DL methods. |
| | Works lacking experimental validation or sufficient methodological detail. |
| | Editorials, commentaries, non-peer-reviewed sources, or duplicate records. |
| | Studies focused only on traditional machine learning without deep learning integration. |

### *2.5 Data Extraction*

For this study, data were systematically extracted from each included paper using a structured approach to ensure consistency and relevance. The information that was taken out included bibliographic information (author, year, publication source), research goals, and the deep learning models used for DDoS detection, such as CNN, RNN, LSTM, autoencoders, GANs, and hybrid approaches. We kept track of dataset information like CICIDS2017, CICDDoS2019, NSL-KDD, and other benchmark or synthetic datasets to see how they could be used for training and testing. To make it possible to compare, key performance metrics like accuracy, precision, recall, F1-score, detection rate, and false positive rate were collected. Also, the results included information on how well the model works, how adaptable it is, and how efficient it is, as well as problems like dataset imbalance, computational overhead, and problems with real-time IoT or edge environments. This structured extraction was the basis for putting together the results, comparing methods, and finding areas where deep learning could be better used for DDoS detection.

## 3. SLR Results

### *3.1 Identification and Screening Process*

The first search through the chosen databases found a total of 446 records. After getting rid of 116 duplicate entries, 330 records were looked at based on their titles and abstracts. 121 of these were not included because they did not meet the initial relevance criteria. We tried to get the full text of the other 209 reports, but we couldn't get to 106 of them because they weren't fully available or had limitations. A thorough review of the other 103 reports determined their eligibility. 6 studies were not included: 2 because they did not clearly focus on DDoS detection using deep learning methods, 2 because they were published before 2015, and 2 because they were not in English. In the end, 97 studies met all of the criteria for inclusion and were included in the systematic review for a detailed synthesis. Figure 3 shows the PRISMA Flow Diagram of the Identification and Screening Process.

## 4. Findings of the study

### *4.1 Existing Research on Deep Learning for DDoS Detection*

Recent studies have concentrated on employing deep learning models to identify distributed denial of service (DDoS) attacks within contemporary communication infrastructures. For example, Reddy et al. [23] came up with a new activation function that worked with a recurrent neural network (RNN) in a software-defined network (SDN) environment using the Open Daylight platform. This made throughput, latency, and overall controller performance much better than other methods. In the same way, Hnamte et al. [24] came up with a way to use a deep neural network (DNN) to find DDoS attacks in SDN environments. They got detection accuracy rates of 99.98%, 100%, and 99.99% on the SDN, CICIDS2018, and Kaggle datasets, respectively. This shows how strong deep learning is in real-world mitigation. Shah et al. [25] underscored the gravity of DDoS attacks in depleting network resources and identified constraints concerning dataset regularization, computational expense, and feature reduction that require additional focus in machine learning-based solutions. Kumar D et al. [26] tackled this issue by creating a model based on LSTM that could update its own feature extraction and got 98% accuracy on the CIDDDoS2019 dataset, which was better than traditional machine learning methods. In a similar vein, S. Ahmed et al. [27] concentrated on application-level DDoS attacks that imitate authentic user behavior, utilizing a multilayer perceptron (MLP) classifier to examine packet attributes such as HTTP frame size, IP mapping, and proxy addresses, achieving a detection efficiency of 98.99% with fewer false positives than traditional classifiers. Mehmood et al. [28] offered a comprehensive examination of machine

learning and deep learning methodologies for DDoS detection, highlighting the escalating threats posed by botnet-driven assaults. Ali Mustapha et al. [29] made further progress by using an RNN short-term memory model to deal with adversarial DDoS traffic created by generative adversarial networks (GANs). They reported detection rates between 91.75% and 100%, but they also pointed out that their method had some weaknesses against more advanced GAN-based attacks. Ahmed et al. [30] built on hybrid approaches and came up with a deep learning-based intrusion detection system for IoT cloud/fog environments. This system combined CNN, LSTM, autoencoder, and DNN models and showed better accuracy, false alarm rate, and true positive rate on the CIC-DDoS2019 dataset than other ML/DL techniques.

Recent ML/DL Techniques for Detecting DDoS Attacks

ML is a branch of artificial intelligence (AI) that includes all methods and algorithms that enable computers to use mathematical models to automatically learn from large datasets. For DDoS detection in SDN (also known as Shallow Learning), the most often used machine learning techniques are Decision Tree (DT), K-Nearest Neighbor (KNN), Artificial Neural Network (ANN), Support Vector Machine (SVM), K-Means Clustering, Fast Learning Networks, Ensemble Methods, and others. Below are succinct descriptions of each category:

**Decision Tree (DT):** An elementary supervised ML approach that classifies and predicts data using regression principles. Every node in the model's tree represents a feature or trait. Each leaf represents a class label or probable result, while the branch represents a choice or rule. DT automatically chooses ideal tree characteristics and prunes superfluous branches to minimize overfitting [31]. The quality of a split is often judged by information gain based on entropy, as given in Eq. (1):

$$H(S) = -\sum_{i=1}^{c} \quad p_i og\ _2 p_i, \tag{1}$$

where $p_i$ is the probability of class $i$. $H(S)$ is the entropy of the dataset $S$.

The information gain of choosing feature $A$ can also be expressed as Eq. (2):

$$H(S) - IG(A) = \sum_{v \in Values(A)} \quad \frac{|\ S_v\ |}{|\ S\ |} H(S_v), \tag{2}$$

where $S_v$ denotes the subset of samples with value $v$ for feature $A$.

**K-Nearest Neighbor (KNN):** K-Nearest Neighbor (KNN) is a basic supervised ML technique that classifies data samples using “feature similarity”. KNN can identify a data sample's class based on its neighbor and distance from them. Too little or too high k values might cause overfitting or improper sample case classification in the KNN algorithm. Researchers using the latest benchmark dataset, CSE-CIC-IDS2018, used the Synthetic Minority Oversampling Technique (SMOTE) to overcome dataset imbalance when evaluating ML algorithms, including KNN, to improve minority class attack detection [32]. The Euclidean distance between two samples is computed using Eq. (3):

$$d(x, y) = \sqrt{\sum_{i=1}^{n} \quad (x_i - y_i)^2}, \tag{3}$$

where $x$ and $y$ denote two feature vectors.

KNN can determine the class label through majority voting among the $k$ nearest neighbors as Eq. (4):

$$\hat{y} = arg\ arg\ \max_{c} \sum_{x_i \in N_k(x)} I(y_i = c), \tag{4}$$

where $N_k(x)$ is the set of the $k$ nearest neighbors of $x$, $y_i$ is the class label of neighbor $x_i$, and $I(\cdot)$ is the indicator function.

**Support Vector Machine (SVM):** The max-margin separation hyperplane in n-dimensional feature space underpins SVM, a supervised ML algorithm. It solves linear and nonlinear problems using kernel functions. SVM uses the kernel function to transfer a low-dimensional input vector into a high-dimensional feature space, then the support vectors to build an optimum maximum marginal hyperplane as a decision boundary. SVM can detect DDoS assaults more efficiently and accurately by properly classifying benign and dangerous classes [33]. The decision boundary is given by

$$f(x) = w^T \phi(x) + b, \tag{5}$$

where $w$ is the weight vector, $\phi(x)$ is the kernel transformed feature vector, and $b$ is the bias term.

The parameters of soft-margin SVM are obtained by minimizing

$$\min_{w,b,\xi} \frac{1}{2} \| w \|^2 + C \sum_{i=1}^{N} \xi_i \tag{6}$$

subject to

$$y_i(w^{\top} \phi(x_i) + b) \geq 1 - \xi_i,$$

$$\xi_i \geq 0,$$

where $C$ is a parameter controlling the trade-off between maximizing the margin and minimizing the classification errors, and $\xi_i$ is the slack variable.

**K-Mean Clustering:** Clustering organizes related data into useful clusters or groupings. K-Mean clustering is a prominent unsupervised iterative ML method. K represents a dataset's total centroids. Allocating data points to clusters usually involves distance. Reducing the distance between data points and cluster centroids is the key aim [34]. Its objective function is

$$J = \sum_{i=1}^{K} \sum_{x_j \in C_i} \| x_j - \mu_i \|^2 . \tag{7}$$

where $J$ denotes the clustering goal, $C_i$ denotes the $i$-th cluster, $x_j$ represents a data point allocated to cluster $C_i$, and $\mu_i$ is the centroid of the $i$-th cluster.

The centroid of each cluster is then updated as

$$\mu_i = \frac{1}{| C_i |} \sum_{x_j \in C_i} x_j, \tag{8}$$

where $| C_i |$ is the number of data points assigned to cluster $C_i$.

**Artificial Neural Network (ANN):** ANN is a supervised ML algorithm inspired by the human nervous system. It has processing units, neurons (nodes), and their connections. The nodes have an input layer, hidden layers, and an

output layer. Backpropagation is how ANNs learn. The main advantage of ANN is its ability to learn from bigger datasets for nonlinear modeling. Training ANN models is challenging because to its length, which might hamper learning and provide suboptimal results [35]. The output of a neuron is

$$y = f(b), \tag{9}$$

where $x_i$ are the input features, $w_i$ are the connection weights, $b$is the bias term, and $f(\cdot)$ is the activation function.

For the popular ReLU activation, the non-linearity is defined as

$$f(z) = max(0, z). \tag{10}$$

**Ensemble methods:** Ensemble approaches aim to profit from several classifiers by learning together. Each classifier has pros and cons; some can detect specific attacks and others. Ensemble approaches train numerous weak classifiers to generate a stronger classifier, which is usually voted on [36]. For a classification task, the ensemble prediction can be expressed as

$$\hat{y} = arg\ arg\ max \quad {}_{c} \sum_{m=1}^{M} I(h_m(x) = c), \tag{11}$$

where $M$ is the total number of base classifiers, $h_m(x)$ is the prediction of the $m$-th classifier for input $x$, $I(\cdot)$ is the indicator function, and $\hat{y}$ is the final predicted class obtained by majority voting.

DL methods include hybrid, semi-supervised, supervised instance, and sequence learning. Short summaries of each category follow:

**Supervised instance learning:** Learning employs instance flow [37]. Labelled instances are used for training. Popular methods in this field are:

- **Deep Neural Networks (DNN**): A key DL structure that lets the model learn at several levels. It has input, output, and hidden layers. DNNs simulate complicated nonlinear functions. More buried layers increase model abstraction and potential. For classification, the output layer has one fully connected layer and a softmax classifier. Hidden layer activation usually uses the Rectified Linear Unit (ReLU) function [38],[39]. The output of a neuron is computed as

$$Z = \sum_{i=1}^{N} x_i w_i + b, \tag{12}$$

$$A = f(Z), \tag{13}$$

where $x_i$ are the input features, $w_i$ are the associated weights, $b$ is the bias term, $Z$ is the weighted sum, $f(\cdot)$ is the activation function (e.g., ReLU), and $A$ is the activated output.

- **Convolutional Neural Network (CNN):** CNNs are ideal for image and signal data. All CNNs have an input layer, a stack of convolutional and pooling layers for feature extraction, a fully connected layer, and a softmax classifier in the classification layer. Computer vision has advanced, and CNNs can conduct supervised feature extraction and classification for DDoS detection [40]. The convolution operation is defined by

$$x_j^{(l)} = f\left(\sum_{i} x_i^{(1)} * k_{ij}^{(l)} + b_j^{(l)}\right), \tag{14}$$

where $x_i^{(1)}$ is the input feature map from the preceding layer, $k_{ij}^{(l)}$ is the convolution kernel, $b_j^{(l)}$ is the bias term, $f(\cdot)$ is the activation function, and $x_j^{(l)}$ is the output feature map.

**Supervised sequence learning:** Supervised sequence learning uses flows to remember earlier input states while learning from a collection of inputs. These models are most popular:

- **Recurrent Neural Networks (RNN):** RNNs enhance feed-forward neural networks and sequence data modeling. Input, hidden, and output units make form an RNN, with hidden units providing memory. RNN units evaluate current and past inputs to make decisions. Semantic understanding, handwriting prediction, speech processing, and human activity identification employ RNNs [41]. In DDoS detection, RNNs may extract features and classify. Before short-term memory issues, RNNs can only handle sequences up to a specific length [42]. The hidden state of an RNN is calculated as

$$h_t = f(W_{xh}x_t + W_{hh}h_{t-1} + b_h), \tag{15}$$

where $x_t$ is the input at time step $t$, $h_{t-1}$ is the preceding hidden state, $W_{xh}$ and $W_{hh}$ are the weight matrices, $b_h$ is the bias term, and $f(\cdot)$ is the activation function.

- **Long Short-Term Memory (LSTM):** One DL structure that has effectively tackled the difficulties with RNNs is LSTM. Several memory cells or blocks make up an LSTM network. By use of three processes called gates—forget, input, and output gates—the subsequent cell obtains both the hidden state and the cell state [43]. The memory blocks have the ability to decide which information to remember or disregard. Information that is no longer needed by the LSTM is removed from the current input by a forget gate [44]. Extracting relevant information from the current input and processing it as an output is the output gate's responsibility. Lastly, the input gate is in charge of adding inputs to the cell state [45]. The cell state is updated by

$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t, \tag{16}$$

where $C_{t-1}$ is the previous cell state, $f_t$ is the forget gate, $i_t$ is the input gate, $\tilde{C}_t$ is the candidate cell state, and $\odot$ is the element-wise multiplication operator.

The forget, input, and output gates of the LSTM are defined as

$$f_t = \sigma\big(W_f[h_{t-1}] + b_f\big), \tag{17}$$

$$i_t = \sigma(W_i[h_{t-1}] + b_i), \tag{18}$$

$$\tilde{C}_t = tanh(W_C[h_{t-1}] + b_C), \tag{19}$$

And

$$o_t = \sigma(W_o[h_{t-1}] + b_o), \tag{20}$$

where $f_t$, $i_t$, and $o_t$ denote the forget, input, and output gates, respectively, $\sigma(\cdot)$ is the sigmoid function, and $\tilde{C}_t$ is the candidate cell state. The hidden state is then obtained as

$$h_t = o_t \odot \quad tanh(C_t), \tag{21}$$

where $h_t$ is the current hidden state and $o_t$ is the output gate.

**Semi-supervised learning:** Semi-supervised learning uses unlabeled data for algorithm pre-training. This method trains a model using labeled and unlabeled data. An autoencoder extracts features, and deep or shallow machine learning models classify them. A popular unsupervised neural network-based deep-learning approach is autoencoding (AE). AE learns the optimal characteristics to get the closest output-input match. An autoencoder has

the same-sized input and output layers, even though buried layers are generally smaller. Symmetric encoder–decoder functioning is crucial to AE. AE comes in stacked, sparse, and variable forms [46]. The encoding and decoding operations are given by

$$h = f(Wx + b), \tag{22}$$

$$\hat{x} = g(W'h + b'), \tag{23}$$

where $x$ is the input vector, $h$ is the encoded latent representation, $W$ and $W'$ are the encoder and decoder weight matrices, $b$and $b'$are the bias terms, $f(\cdot)$ and $g(\cdot)$ are the encoder and decoder activation functions, and $\hat{x}$ is the reconstructed output.

**Hybrid learning:** A hybrid learning approach is a combination of any two additional techniques, such as supervised deep learning, unsupervised deep learning, or shallow machine learning. CNN–LSTM [47],[48],[49], LSTM–Bayes [50], RNN–AE [51], and other hybrid models have been often used by researchers. The output of a hybrid architecture is usually a weighted combination of the outputs or learned representations of their constituent models:

$$y_{hybrid} = \sigma\left(\sum_{k=1}^{M} \alpha_k f_k(x)\right), \tag{24}$$

where $f_k(x)$ is the output or learned representation of the $k$-th sub-model, $\alpha_k$ is its weight learned or empirically determined, $M$ is the number of combined models, and $\sigma(\cdot)$ is the final activation or combination function.

In a CNN-LSTM architecture, CNN is used to extract spatial features from network traffic, which are processed by LSTM to extract temporal dependencies:

$$h_t = LSTM\big((x_t)h_{t-1}\big), \tag{25}$$

where $CNN(x_t)$ is the extracted spatial features from input $x_t$ and $h_t$ is the hidden state of LSTM that stores the learned temporal information.

In an LSTM-Bayes hybrid, the features obtained from LSTM can be classified using a Bayesian decision rule:

$$\hat{y} = arg\ arg\ \max_{y}\left[P(y)\prod_{i} P\big(h_{t,i} \mid y\big)\right], \tag{26}$$

where $h_{t,i}$ is the $i$-th LSTM feature at time $t$, $P(y)$ is the prior class probability and $P\big(h_{t,i} \mid y\big)$ is the corresponding class conditional likelihood.

For an RNN-AE hybrid, the RNN captures the temporal traffic characteristics and the autoencoder reconstructs the input to identify anomalous traffic. The reconstruction error is defined as

$$e_t = \| x_t - \hat{x}_t \|^2, \tag{27}$$

with the traffic treated as an attack when

$$Attack\ if\ e_t > \tau,\ Otherwise\ Normal, \tag{28}$$

where $x_t$ and $\hat{x}_t$ are representations of the input and reconstructed traffic, $e_t$ is the anomaly score, and $\tau$ is the detection threshold set on the validation data.

**Other learning methods:** Transfer learning uses pre-trained models from a repository [46]. Researchers utilize deep learning to train models on one attack domain before applying them to another. The transfer learning approach is described by

$$\theta_T^* = arg\ arg\ \min_{\theta_T} L\left(\left(y_T\right)\theta_T\right),\ initialized\ with\ \theta_S, \tag{29}$$

where $\theta_S$ and $\theta_T$ are the parameters pre-trained in the source domain and the target domain, respectively, $x_T$ denotes the traffic samples in the target domain, $y_T$ denotes the corresponding labels, and $L(\cdot)$ is the loss function minimized for fine-tuning. It enables the transferred learning of knowledge learned in an attack domain to another target domain.

This section has provided a thorough summary of the most popular ML and DL algorithms for DDoS detection systems. Figure 4 illustrates the taxonomy of current ML/DL-based DDoS detection approaches.

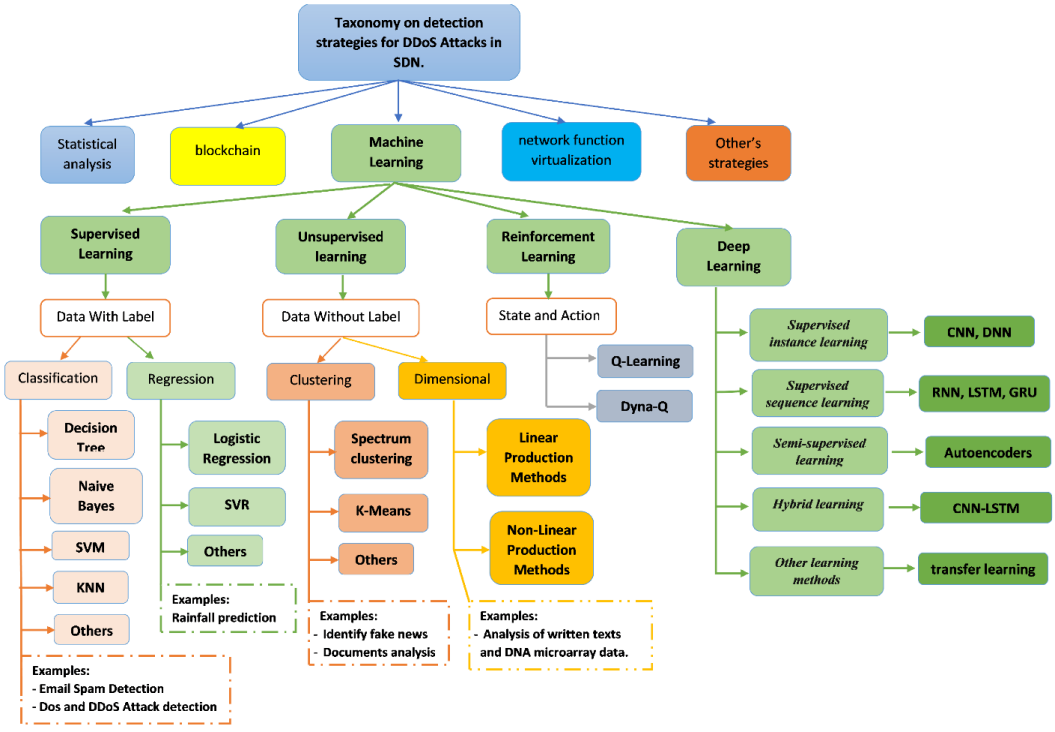


**Fig. 4.** Taxonomy of machine learning-based and deep learning-based DDoS detection systems.

## *4.2 Hybrid-based approaches to detect DDoS attacks in cybersecurity*

The development of hybrid models that combine Deep Learning (DL) and Machine Learning (ML) has shown great potential in raising the precision and effectiveness of intrusion detection systems (IDS). By combining the advantages of machine learning (ML) with deep learning (DL), hybrid models enable more advanced detection of sophisticated cyberattacks such Advanced Persistent Threats (APT) and Distributed Denial of Service (DDoS) [52]. For instance, deep learning models like Convolutional Neural Networks (CNN) or Recurrent Neural Networks (RNN) handle the classification of network traffic, while machine learning algorithms like Random Forest (RF) or Support Vector Machine (SVM) are used for feature extraction and selection [53]. By combining these methods, a greater variety of characteristics and behaviors may be captured, improving the identification of known and unexpected dangers. [52] claim that DL models are excellent at automatically identifying complex patterns in data, particularly when paired with ML-based feature selection, which increases the adaptability of hybrid models in large-scale and real-time settings.

Studies on the advantages of ensemble approaches, which integrate many classifiers to improve detection performance, has grown. A popular strategy is to mix Random Forest (RF) with Support Vector Machine (SVM), taking use of SVM's accuracy in binary classification tasks and RF's capacity to handle high-dimensional data [54]. According to [53], ensemble models that include predictions from SVM, DT, and RF provide noticeably better accuracy and less false-positive rates than solo models. For intrusion detection, in particular, an ensemble approach that included RF and SVM produced an accuracy of over 99%, surpassing individual models that had issues with processing time (SVM) or dimensionality (RF) [55]. Furthermore, it was highlighted by [56] that ensemble techniques such as bagging and boosting increase generalization and detection rates by lowering the variance and bias of individual classifiers, hence strengthening the resilience of intrusion detection systems. All things considered, hybrid and ensemble approaches—which combine the advantages of several models while reducing their drawbacks—have emerged as crucial to improving the status of IDS.

A hybrid CNN-LSTM model with XGBoost feature selection is presented by Zainudin et al. [57] for the categorization of DDoS attacks, with improved accuracy. The hybrid models' potential efficacy in preventing DDoS assaults in IoT networks is shown by the suggested model. The prediction in XGBoost is given as sum of prediction of many boosted decision tree models as:

$$\hat{y}(x) = \sum_{k=1}^{K} \quad f_k(x), \tag{30}$$

where $f_k(x)$ is the prediction of the k-th decision tree, and $K$ is the total number of trees. The obtained feature importance can be used to keep the most relevant traffic features for the subsequent classification.

By putting out a hybrid detection strategy, the authors in [58] address the shortcomings of supervised learning models in identifying unknown DoS/DDoS assaults on the Internet of Things.

A soft-ordering CNN (SO-CNN) with a local outlier factor (LOF) and nearest-neighbor ensembles (NNEs) for isolation-based anomaly identification are combined in this technique. LOF measures how different a sample is from the density of its local neighborhood, and is defined as follows:

$$LOF_k(x) = \frac{1}{\mid N_k(x) \mid} \sum_{o \in N_k(x)} \quad \frac{lrd_k(o)}{lrd_k(x)}, \tag{31}$$

where $N_k$ is the -nearest-neighbor set of samples $x$, and $lrd_k(x)$ denotes its local reachability density. A higher LOF value indicates a greater likelihood that the sample is anomalous. When tested on the BoT-IoT, CIC-IDS-2017, and CIC-IDS-2018 datasets, the suggested model performs better in terms of accuracy.

The authors of a different research [30] provide a brand-new deep learning-based intrusion detection solution for cloud or fog deployment in Internet of Things settings. In a two-level architecture, the hybrid model combines a CNN, LSTM, Deep Autoencoder, and DNN. When tested on the CIC-DDoS2019 dataset, the model outperformed both deep learning and conventional machine learning techniques, obtaining low false alarm rates, high accuracy, and true-positive rates. A hybrid intrusion detection system for the fog computing layer was suggested by the authors in [59],[60]. DNN-KNN is a method that combines DNN and KNN to classify assaults or normal in a binary fashion. In the DNN-KNN method, the DNN is used to map the input traffic vector $x$ into a learned feature representation $z$, where $z$ is given by

$$z = f_\theta(x), \tag{32}$$

then, KNN determines the class using the distance between the learned representations:

$$d(z_i, z) = \sqrt{\sum_{j=1}^{d} \quad \left(z_j - z_{ij}\right)^2}, \tag{33}$$

where $f_\theta(\cdot)$ is the trained DNN mapping, $z_i$ represents the learned feature representation of the $i$-th training sample, $z$ represents the learned feature representation of the input sample, and $d$ is the dimensionality of the learned feature space. The CICIDS2017 and NSL-KDD datasets are used to assess the suggested model.

The hybrid approach shows notable advantages over conventional intrusion detection techniques, providing excellent accuracy and recall rates together with minimal memory and processor overhead. A deep learning-based DDoS detection model that integrates RNN, LSTM, and a CNN into a bidirectional CNN-BiLSTM architecture was suggested by Aswad et al. [61] in order to overcome the difficulties associated with attack detection. The bidirectional CNN-BiLSTM architecture can be expressed as

$$h_t^{BiLSTM} = \left[\left(CNN(h_{t-1})\right)LSTM\left(CNN(h_{t+1})\right)\right]. \tag{34}$$

where $CNN(x_t)$ denotes the spatial features extracted from the traffic input, and forward and backward LSTM units capture the temporal dependencies in both directions. The CICIDS2017 dataset is used to train and evaluate the model.

An intrusion detection method for SDN using a Gated Recurrent Unit Recurrent Neural Network (GRU-RNN) was suggested by the authors in [62]. The update of the GRU hidden state is given by:

$$h_t = (1 - z_t) \odot \quad h_{t-1} + z_t \odot \quad \tilde{h}_t, \tag{35}$$

where $z_t$ is the update gate, $h_{t-1}$ is the previous hidden state, $\tilde{h}_t$ is the candidate hidden state, and $\odot$ denotes element-wise multiplication. There is an improvement in accuracy outcomes.

In order to identify network attacks, Hossain et al. [63] investigate an LSTM-based method. The research uses the CICIDS 2017 labeled dataset to find the best settings for network attack detection by adjusting different LSTM hyperparameters, optimizers, loss functions, learning rates, and activation functions. The TensorFlow framework's LSTM and RNN are used by the authors in [64] to create an intrusion detection system that can identify DDoS assaults. A deep learning-powered intrusion detection system created especially for identifying and thwarting DDoS assaults in Internet of Things networks is presented by the authors in [65]. To improve intrusion detection performance, several deep learning models, such as CNN, CNN, and LSTM, are evaluated using the CICIoT2023 dataset. The quality of the dataset is enhanced via features extraction, normalization, duplicate removal, and random subset selection. In order to improve model generalization, balanced datasets are designed to lessen class imbalance. Nkoro et al. [66] addressed the lack of transparency and reliability in conventional intrusion detection systems by proposing a zero-trust architecture for maritime cybersecurity that makes use of Explainable Artificial Intelligence (XAI). The framework seeks to provide highly accurate and interpretable cyberattack detection and mitigation in marine contexts. The framework improves the interpretability of black-box NIDS models using SHAP and LIME. The SHAP contribution of feature $i$ is given by:

$$\phi_i = \sum_{S \subseteq N \setminus \{i\}} \frac{|S|!\,(M-|S|-1)!}{M!}[f(S \cup \{i\}) - f(S)], \tag{36}$$

where $\phi_i$ represents the contribution of feature $i$, $S$ is a feature subset, $M$ is the total number of features, and $f(\cdot)$ is the model prediction. LIME produces a local explanation using:

$$\xi(x) = arg\ arg \min_{g \in G} [L(\pi_x) + \Omega(g)], \tag{37}$$

where $f$ is the black-box model, $g$ is the interpretable surrogate model, $L$ measures the local approximation error, and $\Omega(g)$ controls the model complexity. The 2023 EdgeIIoTset and CICIoT2023 datasets, which include a variety of IoT-based attack scenarios, are used to assess the framework.

In [67], the authors presented DL-BiLSTM, a lightweight deep learning-based intrusion detection system that combines Bi-LSTM and DNN for improved feature extraction. The model is appropriate for real-time IoT security as it also uses dynamic quantization to reduce computational complexity and incremental principal component analysis (IPCA) for feature reduction. The CICIDS2017, N-BaIoT, and CICIoT2023 datasets are used for training and assessment in the suggested model. Numerical encoding, feature reduction, and normalization are examples of data preparation. The use of deep learning models, such as DNN, CNN, and RNN, for cyberattack detection in IoT networks is examined by Abbas et al. [68]. The models are evaluated using the CICIoT2023 dataset.

Detecting DDoS assaults in IoT security necessitates improved intrusion detection systems (IDSs) since existing signature-based and anomaly-based approaches are inadequate for developing threats. Machine learning approaches improved outcomes by using optimization techniques [69],[70],[71]. Results have improved using hybrid approaches. Deep neural networks (DNN), LSTM, and CNN have improved DDoS attack detection [72],[73],[74]. Further research examined federated transfer learning and protocol-based deep intrusion detection for enhanced IoT security [75]. CNNs and LSTM networks excel in analyzing time and geographic network traffic, making them helpful in detecting complex attack patterns. Using hybrid deep learning architectures like CNN-LSTM and bidirectional CNN-BiLSTM improves detection accuracy and reduces false alarm rates by integrating distinct model strengths. In DDoS attack detection for IoT networks, hybrid learning techniques have gained popularity for their ability to incorporate strengths of several methodologies, leading to enhanced accuracy and resilience. These models combine machine learning and deep learning approaches to take advantage of factors like feature selection efficiency and superior pattern recognition capabilities. Hybrid approaches, incorporating anomaly detection techniques like Isolation Forests or LOF, improve detection of unknown or zero-day attacks in IoT scenarios. These models adapt to dynamic network traffic by combining supervised and unsupervised learning methods.

### *4.3 Datasets for Deep Learning-Based DDoS Detection in cybersecurity*

The datasets and attack class types utilized by the articles that were examined for DDoS attack detection are shown in Table 4. The CICIDS2017, CICDDoS2019, ISCX2012, KDDCUP 1999, NSL-KDD, CSECICIDS2018, and UNSWNB15 datasets are the seven datasets that were utilized in the majority of the articles, it was found.

**Table 4.** Summary of Approaches, Datasets, and Attack Classes in Deep Learning-Based DDoS Detection Studies

| **References** | **Approach Used** | **Dataset Used** | **Classes of Attacks in the Studies** |
|---|---|---|---|
| Virupakshar et al. [76] | DT, KNN, NB, DNN | KDDCUP, LAN, Cloud | KDDCUP99: Normal, DoS, R2L, U2R, Probe. Cloud: ICMP flooding, TCP flooding, HTTP flooding |
| Haider et al. [77] | Ensemble RNN, LSTM, CNN, Hybrid RL | CICIDS2017 | Slowloris, Slowhttptest, Hulk, GoldenEye, Heartbleed, DDoS |
| Wang and Liu [78] | Information entropy and CNN | CICIDS2017 | Benign, BForce, SFTP, SSH, Slowloris, Slowhttptest, Heartbleed, Web BForce, Hulk, GoldenEye, XSS, SQL Injection, Infiltration Dropbox Download, Botnet ARES, Cool disk, DDoS LOIT, PortScans |

| | | | |
|---|---|---|---|
| Kim et al. [79] | CNN | KDDCUP99 and CSE-CIC-IDS2018 | KDDCUP99: Benign, Neptune, Smurf. CSE-CIC-IDS2018: Benign, DoS-SlowHTTPTest, DoS-Hulk, DoS-GoldenEye, DoS-Slowloris, DDoS-HOIC, DDoS-LOIC-HTTP |
| Doriguzzi-Corin et al. [80] | CNN | ISCX2012, CIC2017, CSECIC2018 | ISCX2012: DDoS via IRC botnet. CIC2017: HTTP DDoS via LOIC. CSECIC2018: HTTP DDoS via HOIC |
| Asad et al. [80] | DNN | CICIDS2017 | Benign, DoS Slowloris, DoS Hulk, DoS SlowHTTPTest, DoS GoldenEye |
| Muraleedharan and Janet [81] | DNN | CICIDS2017 | Benign, Slowloris, SlowHTTP, Hulk, GoldenEye |
| Sbai and El Boukhari [82] | DNN | CICDDoS2019 | Data flooding, UDP flooding attack |
| de Assis et al. [83] | CNN | Simulated SDN data, CICDDoS2019 | SDN: DDoS attack. CICDDoS2019: 12 DDoS attacks (training), 7 attacks (testing) |
| Hussain et al. [84] | CNN (ResNet) | CICDDoS2019 | SYN, TFTP, DNS, LDAP, UDP Lag, MSSQL, NetBIOS, SNMP, SSDP, NTP, UDP, Normal traffic |
| Amaizu et al. [85] | DNN | CICDDoS2019 | UDP LAG, SYN, DNS, MSSQL, NTP, SSDP, TFTP, NetBIOS, LDAP, UDP, Benign |
| Cil et al. [86] | DNN | CICDDoS2019 | 12 DDoS attacks (training), 7 attacks (testing) |
| Liang and Znati [87] | LSTM | CICIDS2017 | Slowloris, Hulk, Slowhttptest, GoldenEye, LOIC |
| Shurman et al. [88] | Hybrid IDS, LSTM | Reflection-based CICDDoS2019 | MSSQL, SSDP, CharGen, LDAP, NTP, TFTP, DNS, SNMP, NETBIOS, PORTMAP |
| Assis et al. [45] | GRU | CICDDoS2019, CICIDS2018 | CICDDoS2019: 12 attacks (training), 7 attacks (testing). CICIDS2018: Infiltration, HTTP DoS, web app |

| | | | |
|---|---|---|---|
| | | | attacks, brute force, latest attacks |
| Yang et al. [89] | AE | Synthetic Dataset, UNB2017, MAWI | Synthetic: Excessive GET post-attack, Recursive GET, SlowLoris, Slow POST. UNB2017: Slow HTTP, Hulk, Slowloris, GoldenEye. MAWI: Normal |
| Kasim [90] | AE-SVM | CICIDS2017, NSL-KDD, 6957 DDoS | CICIDS2017: Slowloris, Slowhttptest, Hulk, GoldenEye, DDoS LOIT. NSL-KDD: Back, Land, Pod, Smurf, Neptune, Teardrop, Processtable, UDPStorm, Apache2, Mailbomb, Worm. 6957 DDoS dataset |
| Bhardwaj et al. [91] | AE with DNN | NSL-KDD, CICIDS2017 | NSL-KDD: Back, Land, Teardrop, Mailbomb, Processtable, UDPStorm, Neptune, Pod, Smurf, Apache2, Worm. CICIDS2017: Slowloris, Hulk, Slowhttptest, GoldenEye, DDoS LOIT |
| Premkumar and Sundararajan [92] | RBF | Generated dataset | Data Flooding, Jamming, Exhaustion, Sinkhole, Eavesdropping, Packet dropping |
| Roopak et al. [48] | CNN with LSTM | CICIDS2017 | DDoS |
| Elsayed et al. [51] | RNN-AE | CICDDoS2019 | 12 DDoS attacks (training), 7 attacks (testing) |
| He et al. [93] | 6LANN, 7LANN, 8LANN, 9LANN | – | SYN-type and LDAP-type DDoS attacks |

### *4.4 Performance Metrics for Evaluating Detection Effectiveness*

Researchers have assessed the efficacy of attack detection technologies using several criteria. In this part, we talk about some of the ways that people usually use to find out how well these strategies work.

**Detection Accuracy**

Any method used for attack detection should be able to accurately detect attacks. This metric determines the percentage of accurate predictions across all the cases considered. The detection accuracy (DA) is expressed using Eq. (38):

$$D_A = \frac{T_P + T_N}{T_P + F_P + T_N + F_N} \tag{38}$$

In the equation above, TN represents true negative traffic, which is the number of instances of normal traffic that the detection method correctly classifies as belonging to the normal class; FN represents false negative traffic,

which is the number of attack traffic instances that are recognized as normal traffic; TP represents the number of attack instances that are correctly classified; and FP represents the number of normal traffic instances that are incorrectly classified as attack instances.

**Error Rate**

The error rate (ER) is also known as misclassification or misidentification error. It calculates the proportion of inaccurate predictions across all instances. It is written as shown in Eq. (39):

$$E_R = \frac{F_N + F_P}{F_N + F_P + T_N + T_P} \tag{39}$$

**Specificity**

It is a measure of the normal traffic that is accurately predicted. The specificity ($S_P$) is expressed as Eq. (40):

$$S_P = \frac{T_N}{T_P + F_P} \tag{40}$$

**Precision**

It is an estimate of the proportion of positive patterns in a positive class that are successfully predicted out of all the anticipated patterns. The precision (P) is expressed as shown:

$$P = \frac{T_P}{T_P + F_P} \tag{41}$$

**Sensitivity/Recall(s)**

It measures the proportion of positive patterns that are classified properly. It is expressed as Eq. (42):

$$S = \frac{T_P}{T_P + F_P} \tag{42}$$

**F-Measure/F1-Score**

This metric denotes the harmonic mean between s and P. It is expressed using Eq. (43) as:

$$F - measure = \frac{2 \times s \times P}{s + P} \tag{43}$$

The above performance metrics are widely adopted in the literature to quantify the effectiveness of machine learning and deep learning–based attack detection models. Rather than relying solely on theoretical definitions, prior studies evaluate detection frameworks by reporting these metrics under different datasets and application scenarios. In particular, detection accuracy, error rate, precision, recall, and F-measure are commonly used to compare algorithm robustness, false alarm behavior, and classification reliability. A consolidated comparison of reported performance metrics across representative deep learning–based DDoS detection studies is summarized in Table 5, highlighting how different algorithms perform under identical evaluation criteria and real-world security environments.

**Table 5.** Comparative Analysis of Deep Learning Models for DDoS Detection

| Authors | Dataset | Algorithm(s) | Performance Metrics | Application | Key Findings |
|---|---|---|---|---|---|
| Kona et al. [94] | CICIDS2017 | Time-series ensemble (RF + NN) | DR, ERR, TPR, TNR, Precision, FPR | Cloud security | Ensemble RF–NN model achieved 95.2% accuracy. |

| | | | | | |
|---|---|---|---|---|---|
| Tang et al. [95] | NS2 simulation, test-bed | CNN, BP, LSTM | DR, FN, FP | IoT security | Multifeature CNN fusion demonstrated high detection accuracy and stability for LDoS detection. |
| Shen et al. [96] | SDN flow data | LSTM | Accuracy, Precision, Recall, F1, FPR | Industrial internet | Achieved 99% accuracy for flooding and 98% for stealthy attacks |
| Krishnan et al. [97] | CICIDS2017 | LSTM, CNN | Accuracy, Precision, Recall, F1, ROC, Kappa | Network security | Optimized 1-layer and 2-layer LSTM models improved DoS detection performance. |
| Boukhalfa et al. [98] | NSL-KDD | LSTM | Accuracy, Sensitivity, FPR, Precision, Recall, F-measure | Network security | LSTM demonstrated better performance than SVM and KNN. |
| Abdallah et al. [99] | SDN dataset | CNN–LSTM | Accuracy, Precision, Recall, F1 | SDN | Hybrid CNN–LSTM achieved 96.32% accuracy, surpassing individual models. |
| Imrana et al. [100] | NSL-KDD | BiDLSTM | Accuracy, Precision, Recall, F-score | Internet security | Improved detection but higher training time and complexity |
| Sai Sindhu Theja et al. [101] | KDD Cup 99 | OCSA + RNN | Accuracy, Precision, Recall, F-measure | Cloud security | OCSA-RNN demonstrated improved performance compared with other metaheuristic RNN models. |
| Li et al. [102] | DARPA 1999, DARPA 2009, CICDDoS2019 | QuinDC | TPR, FPR, TFOR, FOR | IoT security | Real-time volumetric detection with low latency |
| Yang et al. [103] | UNB 2017, MAWI | AE-D3F | DR, FPR, FNR | Network security | Achieved 82% DR with 0 FPR |

| | | | | | |
|---|---|---|---|---|---|
| Saharkhizan et al. [104] | Modbus traffic | Ensemble LSTM | Accuracy, Precision, Recall, F-measure | IoT cyberattacks | Achieved 98.99% accuracy and 92.32% DR |
| Parra et al. [105] | N-BaIoT | CNN, LSTM | Accuracy, Precision, Recall, TPR, TNR | IoT security | CNN and LSTM achieved 94.3% and 94.8% accuracy |
| Ahmed et al. [106] | CTU-13 | ANN, DL | Accuracy | Cyber security | Effective zero-day botnet detection compared with SVM and NB |
| Lam et al. [107] | IDS-2018 | CNN-NAS | Precision, Recall, F1 | IoT security | Achieved 96.4% detection rate |
| Agarwal et al. [108] | KDD99 | KNN, CNN, LSTM | Accuracy | Cyber security | SAAT framework reduced false feedback |
| Asad et al. [109] | CICIDS2017 | DeepDetect NN | F1, ROC | Internet security | Achieved 98% accuracy and 0.99 F1 |
| Sumathi et al. [110] | NSL-KDD | C4.5, SVM, KNN | Accuracy, Precision, F1 | Cloud security | SVM achieved high detection performance using minimal features. |
| Novaes et al. [111] | CICDDoS2019 | DNN, GAN | Accuracy, Precision, Recall | SDN | GAN-based DNN demonstrated improved performance compared with CNN and LSTM. |
| Bovenzi et al. [112] | Bot-IoT | Hierarchical AE | F1, ROC, TPR | IoT security | H2ID achieved a high F1 score. |

### *4.5 Challenges and Research Gaps in Deep Learning-Based DDoS Detection*

**1. Detection Speed and Computational Overhead**

Some detection systems that utilize entropy and statistics are quick and don't need a lot of processing power, but they typically aren't accurate enough and have a lot of false alarms, which makes them hard to apply in real life. Machine learning and deep learning make things more accurate, but they also take longer to process since they have more layers. This means that there is still a trade-off between speed and accuracy that hasn't been addressed yet [113].

**2. Real-Time Realization for IoT Devices with Limited Resources**

Most current detection approaches don't work for IoT situations that happen in real time. Deep learning needs strong technology, such GPUs, which use a lot of energy. This makes them unsuitable for IoT and UAV contexts

where resources are restricted. Long processing periods and extra work for the computer might drain the device's battery, and even GPU acceleration has trouble cutting down on training time [114].

**3. Selecting Features and Setting Adaptive Thresholds**

Detection performance relies significantly on thresholds and chosen characteristics, which fluctuate with traffic dynamics. Because attack structures change quickly, thresholds need to be able to adjust to new characteristics on their own. Adaptive mechanisms, on the other hand, add to the computational load, thus there has to be a balance between adaptability, detection accuracy, and speed, which is not yet found in the literature.

**4. Self-Learning and Adaptation**

DDoS attack paths are always changing, particularly now that 5G networks are connecting more devices. Current models may not work for all forms of attacks. Frameworks that quickly learn and adapt to changing attack patterns while keeping low computational time and good accuracy are needed.

**5. Data Quality Issues**

The amount and quality of the dataset are very important for deep learning models. Some of the problems include that there aren't enough high-quality datasets that are easy to get to, that the data classes are unbalanced (which lowers accuracy), that the datasets are old, and that there isn't enough variety in the traffic. Some recent research look at class imbalance [115],[116],[117],[118], but more effort is required to make sure that datasets are reliable for detecting contemporary attacks.

**6. Lack of Real-Time Datasets**

Ethical limits make it hard to get real-time assault datasets. Commonly used datasets like Kyoto, UNIBS, Kent2016, and ISCX2012 are no longer useful. On the other hand, CICIDS2017 and CICDDoS2019 are simulated and do not show how smart-device networks are changing or how assaults are changing. This constraint also applies to IoFT and drone networks, which have little datasets. Researchers commonly employ datasets from other fields [119], although these may not be a good representation of real-world traffic. There is an urgent demand for realistic, balanced, and dynamic datasets that accurately reflect authentic DDoS attack behavior across many contexts.

## 5. Conclusion

This systematic review has offered a thorough synthesis of research regarding the application of deep learning (DL) models for the detection of DDoS attacks in the realm of cybersecurity. The results demonstrate that deep learning architectures, including CNNs, RNNs, LSTMs, autoencoders, and hybrid frameworks, substantially improve detection accuracy, decrease false positives, and adapt more efficiently to changing attack patterns in comparison to conventional machine learning techniques. Hybrid models, in particular, show great promise because they use the best parts of both DL and ML methods to make systems more resilient and scalable in the real world.

The review also points out a number of problems that keep coming up. Dataset limitations—like imbalance, old traffic, and not having real-time or domain-specific data—are still a big problem for reliable evaluation. High computational costs, delays in real-time detection, and limited resources in IoT/edge environments make practical deployment even harder. Also, problems with interpretability, vulnerability to attacks, and inconsistent use of performance metrics still make it hard to use at scale. To fill these gaps, we need to create standardized, diverse, and realistic benchmark datasets; lightweight and energy-efficient deep learning models that work on limited devices; and explainable AI methods that can boost trust and openness. The future looks bright for the combination of federated learning, transfer learning, and adversarially robust architectures, as well as hybrid frameworks that combine anomaly detection with predictive intelligence.

Deep learning is a strong way to improve cybersecurity defenses against DDoS attacks in general. However, progress depends on filling in the gaps in current research and turning laboratory-level breakthroughs into useful, scalable, and flexible solutions that can handle the ever-changing and complex nature of modern cyber threats.

**Author Contributions**.

## 6. Acknowledgments

Acknowledgments may be made to individuals or institutions that have made an important contribution.

**Conflict of Interest**

The author(s) declared no potential conflicts of interest concerning the research, authorship, and publication of this article.

**Funding**

The author(s) received no financial support for the research, authorship, and publication of this article.

**Data Availability Statements**

The datasets generated and/or analyzed during the current study are available from the corresponding author on reasonable request.

## REFERENCES

1. M. Shrestha, C. Johansen, J. Noll, and D. Roverso, "A methodology for security classification applied to smart grid infrastructures," *Int. J. Crit. Infrastruct. Prot.*, vol. 28, p. 100342, 2020.
2. J. Kramer, "Cybercrime To Cost The World $10.5 Trillion Annually By 2025." [Online]. Available: https://www.sec.gov/Archives/edgar/data/736012/000168316820004004/intrusion_ex9901.htm
3. Statista, "Number of Internet of Things (IoT) connections worldwide from 2022 to 2023, with forecasts from 2024 to 2034," 2024. [Online]. Available: https://www.statista.com/statistics/1183457/iot-connected-devices-worldwide/
4. V.-D. Ngo, T.-C. Vuong, T. Van Luong, and H. Tran, "Machine learning-based intrusion detection: feature selection versus feature extraction," *Cluster Comput.*, vol. 27, no. 3, pp. 2365–2379, 2024.
5. A. Lenka, M. Goswami, H. Singh, and H. Baskaran, "Cybersecurity disclosure and corporate reputation: Rising popularity of cybersecurity in the business world," in *Effective Cybersecurity Operations for Enterprise-Wide Systems*, IGI Global, 2023, pp. 169–183.
6. J. Kotsias, A. Ahmad, and R. Scheepers, "Adopting and integrating cyber-threat intelligence in a commercial organisation," *Eur. J. Inf. Syst.*, vol. 32, no. 1, pp. 35–51, 2023.
7. H. Gately, "Russian organised crime and Ransomware as a Service: state cultivated cybercrime," 2023, *Macquarie University*.
8. S. Tufail, I. Parvez, S. Batool, and A. Sarwat, "A survey on cybersecurity challenges, detection, and mitigation techniques for the smart grid," *Energies*, vol. 14, no. 18, p. 5894, 2021.
9. M. S. Abu, S. R. Selamat, A. Ariffin, and R. Yusof, "Cyber threat intelligence–issue and challenges," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 10, no. 1, pp. 371–379, 2018.
10. K. Huang, L.-X. Yang, X. Yang, Y. Xiang, and Y. Y. Tang, "A low-cost distributed denial-of-service attack architecture," *IEEE access*, vol. 8, pp. 42111–42119, 2020.
11. K. B. Adedeji, A. M. Abu-Mahfouz, and A. M. Kurien, "DDoS attack and detection methods in internet-enabled networks: Concept, research perspectives, and challenges," *J. Sens. Actuator Networks*, vol. 12, no. 4, p. 51, 2023.
12. T. Shorey, D. Subbaiah, A. Goyal, A. Sakxena, and A. K. Mishra, "Performance comparison and analysis of slowloris, goldeneye and xerxes ddos attack tools," in *2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, IEEE, 2018, pp. 318–322.
13. C. Douligeris and A. Mitrokotsa, "DDoS attacks and defense mechanisms: classification and state-of-the-art," *Comput. networks*, vol. 44, no. 5, pp. 643–666, 2004.
14. M. Cirillo, M. Di Mauro, V. Matta, and M. Tambasco, "Botnet identification in DDoS attacks with multiple emulation dictionaries," *IEEE Trans. Inf. Forensics Secur.*, vol. 16, pp. 3554–3569, 2021.
15. Carol Hildebrand, "Crossing the 10 Million Mark: DDoS Attacks in 2020." [Online]. Available: https://www.netscout.com/blog/asert/crossing-10-million-mark-ddos-attacks-2020
16. C. Hildebrand, "The Beat Goes On Record-breaking DDoS activity surged into the first quarter of 2021." [Online]. Available: https://www.netscout.com/blog/asert/beat-goes
17. NETSCOUT, "NETSCOUT Identified Nearly 7.9 Million DDoS Attacks in 1H2023 According to Its Latest DDoS Threat Intelligence Report." [Online]. Available: https://www.netscout.com/press-releases/netscout-identified-nearly-79-million-ddos-attacks-1h2023
18. O. Yoachimik and J. Pacheco, "Record-breaking 5.6 Tbps DDoS attack and global DDoS trends for 2024 Q4." [Online]. Available: https://blog.cloudflare.com/ddos-threat-report-for-2024-q4/#emerging-attack-vectors

19. O. Yoachimik, J. Pacheco, and C. One, “2025 Q4 DDoS threat report: A record-setting 31.4 Tbps attack caps a year of massive DDoS assaults.” [Online]. Available: https://blog.cloudflare.com/ddos-threat-report-2025-q4/
20. Y. Kim, “Convolutional neural networks for sentence classification,” *arXiv Prepr. arXiv1408.5882*, 2014.
21. Y. Zhou *et al.*, “Hybrid genetic algorithm method for efficient and robust evaluation of remaining useful life of supercapacitors,” *Appl. Energy*, vol. 260, no. November 2019, 2020, doi: 10.1016/j.apenergy.2019.114169.
22. T. A. Tang, L. Mhamdi, D. McLernon, S. A. R. Zaidi, and M. Ghogho, “Deep learning approach for network intrusion detection in software defined networking,” in *2016 international conference on wireless networks and mobile communications (WINCOM)*, IEEE, 2016, pp. 258–263.
23. P. Reddy, Y. Adetuwo, and A. K. Jakkani, “Implementation of machine learning techniques for cloud security in detection of ddos attacks,” *Int. J. Comput. Eng. Technol.*, vol. 15, no. 2, pp. 25–34, 2024.
24. V. Hnamte, A. A. Najar, H. Nhung-Nguyen, J. Hussain, and M. N. Sugali, “DDoS attack detection and mitigation using deep neural network in SDN environment,” *Comput. Secur.*, vol. 138, p. 103661, 2024.
25. A. Shah, D. Rathod, and D. Dave, “DDoS attack detection using artificial neural network,” in *International Conference on Computing Science, Communication and Security*, Springer, 2021, pp. 46–66.
26. D. Kumar, R. K. Pateriya, R. K. Gupta, V. Dehalwar, and A. Sharma, “DDoS detection using deep learning,” *Procedia Comput. Sci.*, vol. 218, pp. 2420–2429, 2023.
27. S. Ahmed *et al.*, “Effective and efficient DDoS attack detection using deep learning algorithm, multi-layer perceptron,” *Futur. Internet*, vol. 15, no. 2, p. 76, 2023.
28. M. A. Al-Shareeda, S. Manickam, and M. A. Saare, “DDoS attacks detection using machine learning and deep learning techniques: analysis and comparison,” *Bull. Electr. Eng. Informatics*, vol. 12, no. 2, pp. 930–939, 2023.
29. A. Mustapha *et al.*, “Detecting DDoS attacks using adversarial neural network,” *Comput. Secur.*, vol. 127, p. 103117, 2023.
30. A. Ahmim, F. Maazouzi, M. Ahmim, S. Namane, and I. Ben Dhaou, “Distributed denial of service attack detection for the Internet of Things using hybrid deep learning model,” *IEEE Access*, vol. 11, pp. 119862–119875, 2023.
31. V. G. Costa and C. E. Pedreira, “Recent advances in decision trees: an updated survey,” *Artif. Intell. Rev.*, vol. 56, no. 5, pp. 4765–4800, 2023.
32. Y. Zhang, G. Cao, B. Wang, and X. Li, “A novel ensemble method for k-nearest neighbor,” *Pattern Recognit.*, vol. 85, pp. 13–25, 2019.
33. A. Yılmaz, A. Küçüker, G. Bayrak, D. Ertekin, M. Shafie-Khah, and J. M. Guerrero, “An improved automated PQD classification method for distributed generators with hybrid SVM-based approach using un-decimated wavelet transform,” *Int. J. Electr. Power Energy Syst.*, vol. 136, p. 107763, 2022.
34. H. Ren and H. Lu, “Compositional coding capsule network with k-means routing for text classification,” *Pattern Recognit. Lett.*, vol. 160, pp. 1–8, 2022.
35. R. Gopi *et al.*, “Enhanced method of ANN based model for detection of DDoS attacks on multimedia internet of things,” *Multimed. Tools Appl.*, vol. 81, no. 19, pp. 26739–26757, 2022.
36. A. Zeinalpour and H. A. Ahmed, “Addressing the effectiveness of DDoS-attack detection methods based on the clustering method using an ensemble method,” *Electronics*, vol. 11, no. 17, p. 2736, 2022.
37. S. Gamage and J. Samarabandu, “Deep learning methods in network intrusion detection: A survey and an objective comparison,” *J. Netw. Comput. Appl.*, vol. 169, p. 102767, 2020.
38. A. Sellami and S. Tabbone, “Deep neural networks-based relevant latent representation learning for hyperspectral image classification,” *Pattern Recognit.*, vol. 121, p. 108224, 2022.
39. J. Brownlee, “A Gentle Introduction to the Rectified Linear Unit (ReLU). 2019,” *URL https//machinelearningmastery. com/rectified-linear-activation-function-for-deep-learning-neur*, 2022.
40. C. F. G. Dos Santos and J. P. Papa, “Avoiding overfitting: A survey on regularization methods for convolutional neural networks,” *ACM Comput. Surv.*, vol. 54, no. 10s, pp. 1–25, 2022.
41. S. P. Yadav, S. Zaidi, A. Mishra, and V. Yadav, “Survey on machine learning in speech emotion recognition and vision systems using a recurrent neural network (RNN),” *Arch. Comput. Methods Eng.*, vol. 29, no. 3, pp. 1753–1770, 2022.
42. T. E. Ali, Y.-W. Chong, and S. Manickam, “Machine learning techniques to detect a DDoS attack in SDN: A systematic review,” *Appl. Sci.*, vol. 13, no. 5, p. 3183, 2023.
43. M. A. Al Mehedi, M. Khosravi, M. M. S. Yazdan, and H. Shabanian, “Exploring temporal dynamics of river discharge using univariate long short-term memory (LSTM) recurrent neural network at East Branch of Delaware River,” *Hydrology*, vol. 9, no. 11, p. 202, 2022.
44. M. Mittal, K. Kumar, and S. Behal, “Deep learning approaches for detecting DDoS attacks: A systematic review,” *Soft Comput.*, vol. 27, no. 18, pp. 13039–13075, 2023.
45. M. V. O. Assis, L. F. Carvalho, J. Lloret, and M. L. Proença Jr, “A GRU deep learning system against attacks in software defined networks,” *J. Netw. Comput. Appl.*, vol. 177, p. 102942, 2021.
46. A. Aldweesh, A. Derhab, and A. Z. Emam, “Deep learning approaches for anomaly-based intrusion detection systems: A survey, taxonomy, and open issues,” *Knowledge-Based Syst.*, vol. 189, p. 105124, 2020.
47. M. Roopak, G. Y. Tian, and J. Chambers, “Deep learning models for cyber security in IoT networks,” in *2019 IEEE 9th annual computing and communication workshop and conference (CCWC)*, IEEE, 2019, pp. 452–457.
48. M. Roopak, G. Y. Tian, and J. Chambers, “An intrusion detection system against DDoS attacks in IoT networks,” in *2020 10th annual computing and communication workshop and conference (CCWC)*, IEEE, 2020, pp. 562–567.

49. B. Nugraha and R. N. Murthy, “Deep learning-based slow DDoS attack detection in SDN-based networks,” in *2020 IEEE conference on network function virtualization and software defined networks (NFV-SDN)*, IEEE, 2020, pp. 51–56.
50. H. Mohammadnia and S. Ben Slimane, “IoT-NETZ: Practical spoofing attack mitigation approach in SDWN network,” in *2020 Seventh International Conference on Software Defined Systems (SDS)*, IEEE, 2020, pp. 5–13.
51. M. S. Elsayed, N.-A. Le-Khac, S. Dev, and A. D. Jurcut, “DDoSNet: A deep-learning model for detecting network attacks,” in *2020 IEEE 21st International Symposium on" A World of Wireless, Mobile and Multimedia Networks"(WoWMoM)*, IEEE, 2020, pp. 391–396.
52. H. Al-Sahaf *et al.*, “A survey on evolutionary machine learning,” *J. R. Soc. New Zeal.*, vol. 49, no. 2, pp. 205–228, 2019.
53. C. Lee, P. Panda, G. Srinivasan, and K. Roy, “Training deep spiking convolutional neural networks with stdp-based unsupervised pre-training followed by supervised fine-tuning,” *Front. Neurosci.*, vol. 12, p. 435, 2018.
54. O. Samuel *et al.*, “Towards modified entropy mutual information feature selection to forecast medium-term load using a deep learning model in smart homes,” *Entropy*, vol. 22, no. 1, p. 68, 2020.
55. H. H. Pajouh, R. Javidan, R. Khayami, A. Dehghantanha, and K.-K. R. Choo, “A two-layer dimension reduction and two-tier classification model for anomaly-based intrusion detection in IoT backbone networks,” *IEEE Trans. Emerg. Top. Comput.*, vol. 7, no. 2, pp. 314–323, 2016.
56. B. Appasani and D. K. Mohanta, “A review on synchrophasor communication system: communication technologies, standards and applications,” *Prot. Control Mod. power Syst.*, vol. 3, no. 4, pp. 1–17, 2018.
57. A. Zainudin, L. A. C. Ahakonye, R. Akter, D.-S. Kim, and J.-M. Lee, “An efficient hybrid-dnn for ddos detection and classification in software-defined iiot networks,” *IEEE Internet Things J.*, vol. 10, no. 10, pp. 8491–8504, 2022.
58. X.-H. Nguyen and K.-H. Le, “Robust detection of unknown DoS/DDoS attacks in IoT networks using a hybrid learning model,” *Internet of Things*, vol. 23, p. 100851, 2023.
59. D. Mohamed and O. Ismael, “Enhancement of an IoT hybrid intrusion detection system based on fog-to-cloud computing,” *J. Cloud Comput.*, vol. 12, no. 1, p. 41, 2023.
60. K. Kalaivani and M. Chinnadurai, “A Hybrid Deep Learning Intrusion Detection Model for Fog Computing Environment.,” *Intell. Autom. Soft Comput.*, vol. 30, no. 1, 2021.
61. F. M. Aswad, A. M. S. Ahmed, N. A. M. Alhammadi, B. A. Khalaf, and S. A. Mostafa, “Deep learning in distributed denial-of-service attacks detection method for Internet of Things networks,” *J. Intell. Syst.*, vol. 32, no. 1, p. 20220155, 2023.
62. T. A. Tang, D. McLernon, L. Mhamdi, S. A. R. Zaidi, and M. Ghogho, “Intrusion detection in sdn-based networks: Deep recurrent neural network approach,” in *Deep Learning Applications for Cyber Security*, Springer, 2019, pp. 175–195.
63. M. D. Hossain, H. Ochiai, D. Fall, and Y. Kadobayashi, “LSTM-based network attack detection: performance comparison by hyper-parameter values tuning,” in *2020 7th IEEE International Conference on Cyber Security and Cloud Computing (CSCloud)/2020 6th IEEE International Conference on Edge Computing and Scalable Cloud (EdgeCom)*, IEEE, 2020, pp. 62–69.
64. P. K. Bediako, “Long Short-Term Memory Recurrent Neural Network for detecting DDoS flooding attacks within TensorFlow Implementation framework.,” 2017.
65. S. Hizal, U. Cavusoglu, and D. Akgun, “A novel deep learning-based intrusion detection system for IoT DDoS security,” *Internet of Things*, vol. 28, p. 101336, 2024.
66. E. C. Nkoro, J. N. Njoku, C. I. Nwakanma, J.-M. Lee, and D.-S. Kim, “Zero-trust marine cyberdefense for iot-based communications: An explainable approach,” *Electronics*, vol. 13, no. 2, p. 276, 2024.
67. Z. Wang, H. Chen, S. Yang, X. Luo, D. Li, and J. Wang, “A lightweight intrusion detection method for IoT based on deep learning and dynamic quantization,” *PeerJ Comput. Sci.*, vol. 9, p. e1569, 2023.
68. S. Abbas *et al.*, “Evaluating deep learning variants for cyber-attacks detection and multi-class classification in IoT networks,” *PeerJ Comput. Sci.*, vol. 10, p. e1793, 2024.
69. D. C. Nguyen, M. Ding, P. N. Pathirana, A. Seneviratne, J. Li, and H. V. Poor, “Federated learning for internet of things: A comprehensive survey,” *IEEE Commun. Surv. tutorials*, vol. 23, no. 3, pp. 1622–1658, 2021.
70. X. Li, J. Cheng, B. Zhang, X. Tang, and M. Sun, “An Adaptive DDoS Detection and Classification Method in Blockchain Using an Integrated Multi-Models.,” *Comput. Mater. Contin.*, vol. 77, no. 3, 2023.
71. D. C. Attota, V. Mothukuri, R. M. Parizi, and S. Pouriyeh, “An ensemble multi-view federated learning intrusion detection for IoT,” *IEEE Access*, vol. 9, pp. 117734–117745, 2021.
72. A. Mihoub, O. Ben Fredj, O. Cheikhrouhou, A. Derhab, and M. Krichen, “Denial of service attack detection and mitigation for internet of things using looking-back-enabled machine learning techniques,” *Comput. Electr. Eng.*, vol. 98, p. 107716, 2022.
73. S. Ullah, Z. Mahmood, N. Ali, T. Ahmad, and A. Buriro, “Machine learning-based dynamic attribute selection technique for DDoS attack classification in IoT networks,” *Computers*, vol. 12, no. 6, p. 115, 2023.
74. T. Khempetch and P. Wuttidittachotti, “DDoS attack detection using deep learning,” *IAES Int. J. Artif. Intell.*, vol. 10, no. 2, p. 382, 2021.
75. F. Yan, G. Zhang, D. Zhang, X. Sun, B. Hou, and N. Yu, “TL-CNN-IDS: Transfer learning-based intrusion detection system using convolutional neural network.,” *J. Supercomput.*, vol. 79, no. 15, 2023.
76. K. B. Virupakshar, M. Asundi, K. Channal, P. Shettar, S. Patil, and D. G. Narayan, “Distributed denial of service (DDoS) attacks detection system for OpenStack-based private cloud,” *Procedia Comput. Sci.*, vol. 167, pp. 2297–2307, 2020.

77. S. Haider *et al.*, “A deep CNN ensemble framework for efficient DDoS attack detection in software defined networks,” *Ieee Access*, vol. 8, pp. 53972–53983, 2020.
78. L. Wang and Y. Liu, “A DDoS attack detection method based on information entropy and deep learning in SDN,” in *2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)*, IEEE, 2020, pp. 1084–1088.
79. J. Kim, J. Kim, H. Kim, M. Shim, and E. Choi, “CNN-Based Network Intrusion Detection against Denial-of-Service Attacks,” 2020. doi: 10.3390/electronics9060916.
80. R. Doriguzzi-Corin, S. Millar, S. Scott-Hayward, J. Martinez-del-Rincon, and D. Siracusa, “LUCID: A practical, lightweight deep learning solution for DDoS attack detection,” *IEEE Trans. Netw. Serv. Manag.*, vol. 17, no. 2, pp. 876–889, 2020.
81. N. Muraleedharan and B. Janet, “A deep learning based HTTP slow DoS classification approach using flow data,” *ICT Express*, vol. 7, no. 2, pp. 210–214, 2021.
82. O. Sbai and M. El boukhari, “Data flooding intrusion detection system for manets using deep learning approach,” in *Proceedings of the 13th international conference on intelligent systems: theories and applications*, 2020, pp. 1–5.
83. M. V. O. de Assis, L. F. Carvalho, J. J. P. C. Rodrigues, J. Lloret, and M. L. Proença Jr, “Near real-time security system applied to SDN environments in IoT networks using convolutional neural network,” *Comput. Electr. Eng.*, vol. 86, p. 106738, 2020.
84. F. Hussain, S. G. Abbas, M. Husnain, U. U. Fayyaz, F. Shahzad, and G. A. Shah, “IoT DoS and DDoS attack detection using ResNet,” in *2020 IEEE 23rd International Multitopic Conference (INMIC)*, IEEE, 2020, pp. 1–6.
85. G. C. Amaizu, C. I. Nwakanma, S. Bhardwaj, J.-M. Lee, and D.-S. Kim, “Composite and efficient DDoS attack detection framework for B5G networks,” *Comput. Networks*, vol. 188, p. 107871, 2021.
86. A. E. Cil, K. Yildiz, and A. Buldu, “Detection of DDoS attacks with feed forward based deep neural network model,” *Expert Syst. Appl.*, vol. 169, p. 114520, 2021.
87. X. Liang and T. Znati, “A long short-term memory enabled framework for DDoS detection,” in *2019 IEEE global communications conference (GLOBECOM)*, IEEE, 2019, pp. 1–6.
88. M. Shurman, R. Khrais, and A. Yateem, “DoS and DDoS attack detection using deep learning and IDS,” *Int. Arab J. Inf. Technol*, vol. 17, no. 4A, pp. 655–661, 2020.
89. K. Yang, J. Zhang, Y. Xu, and J. Chao, “Ddos attacks detection with autoencoder,” in *NOMS 2020-2020 IEEE/IFIP network operations and management symposium*, IEEE, 2020, pp. 1–9.
90. Ö. Kasim, “An efficient and robust deep learning based network anomaly detection against distributed denial of service attacks,” *Comput. Networks*, vol. 180, p. 107390, 2020.
91. A. Bhardwaj, V. Mangat, and R. Vig, “Hyperband tuned deep neural network with well posed stacked sparse autoencoder for detection of DDoS attacks in cloud,” *Ieee Access*, vol. 8, pp. 181916–181929, 2020.
92. M. Premkumar and T. V. P. Sundararajan, “DLDM: Deep learning-based defense mechanism for denial of service attacks in wireless sensor networks,” *Microprocess. Microsyst.*, vol. 79, p. 103278, 2020.
93. J. He, Y. Tan, W. Guo, and M. Xian, “A small sample DDoS attack detection method based on deep transfer learning,” in *2020 international conference on computer communication and network security (CCNS)*, IEEE, 2020, pp. 47–50.
94. S. S. Kona, “Detection of DDoS attacks using RNN-LSTM and Hybrid model ensemble,” 2020, *Dublin, National College of Ireland*.
95. D. Tang, L. Tang, W. Shi, S. Zhan, and Q. Yang, “MF-CNN: a new approach for LDoS attack detection based on multi-feature fusion and CNN,” *Mob. Networks Appl.*, vol. 26, no. 4, pp. 1705–1722, 2021.
96. C. Shen, G. Xiao, S. Yao, B. Zhou, Z. Pan, and H. Zhang, “An LSTM based malicious traffic attack detection in industrial internet,” in *2021 International Conference on Security, Pattern Analysis, and Cybernetics (SPAC)*, IEEE, 2021, pp. 60–65.
97. D. Krishnan, “Detection of denial-of-service attacks using stacked LSTM networks,” in *Proceedings of Data Analytics and Management: ICDAM 2021, Volume 2*, Springer, 2021, pp. 229–239.
98. A. S. Mirkhail and Z. Xinyou, “Deep Learning for Anomaly Detection in IoT Healthcare Systems,” *Int. Res. J. Multidiscip. Scope*, vol. 6, no. 2, pp. 1480–1494, 2025, doi: 10.47857/irjms.2025.v06i02.03768.
99. M. Abdallah, N. An Le Khac, H. Jahromi, and A. Delia Jurcut, “A hybrid CNN-LSTM based approach for anomaly detection systems in SDNs,” in *Proceedings of the 16th International Conference on Availability, Reliability and Security*, 2021, pp. 1–7.
100. Y. Imrana, Y. Xiang, L. Ali, and Z. Abdul-Rauf, “A bidirectional LSTM deep learning approach for intrusion detection,” *Expert Syst. Appl.*, vol. 185, p. 115524, 2021.
101. R. SaiSindhuTheja and G. K. Shyam, “An efficient metaheuristic algorithm based feature selection and recurrent neural network for DoS attack detection in cloud computing environment,” *Appl. Soft Comput.*, vol. 100, p. 106997, 2021.
102. J. Li, M. Liu, Z. Xue, X. Fan, and X. He, “RTVD: A real-time volumetric detection scheme for DDoS in the Internet of Things,” *IEEE Access*, vol. 8, pp. 36191–36201, 2020.
103. E. M. Bårli, A. Yazidi, E. H. Viedma, and H. Haugerud, “DoS and DDoS mitigation using Variational Autoencoders,” *Comput. Networks*, vol. 199, p. 108399, 2021, doi: https://doi.org/10.1016/j.comnet.2021.108399.
104. M. Saharkhizan, A. Azmoodeh, A. Dehghantanha, K.-K. R. Choo, and R. M. Parizi, “An ensemble of deep recurrent neural networks for detecting IoT cyber attacks using network traffic,” *IEEE Internet Things J.*, vol. 7, no. 9, pp. 8852–8859, 2020.

105. G. D. L. T. Parra, P. Rad, K.-K. R. Choo, and N. Beebe, “Detecting Internet of Things attacks using distributed deep learning,” *J. Netw. Comput. Appl.*, vol. 163, p. 102662, 2020.
106. A. A. Ahmed, W. A. Jabbar, A. S. Sadiq, and H. Patel, “Deep learning-based classification model for botnet attack detection,” *J. Ambient Intell. Humaniz. Comput.*, vol. 13, no. 7, pp. 3457–3466, 2022.
107. J. Lam and R. Abbas, “Machine learning based anomaly detection for 5g networks,” *arXiv Prepr. arXiv2003.03474*, 2020.
108. S. Agarwal, A. Tyagi, and G. Usha, “A deep neural network strategy to distinguish and avoid cyber-attacks,” in *Artificial intelligence and evolutionary computations in engineering systems*, Springer, 2020, pp. 673–681.
109. M. Asad, M. Asim, T. Javed, M. O. Beg, H. Mujtaba, and S. Abbas, “Deepdetect: detection of distributed denial of service attacks using deep learning,” *Comput. J.*, vol. 63, no. 7, pp. 983–994, 2020.
110. S. Sumathi, R. Rajesh, and N. Karthikeyan, “DDoS attack detection using hybrid machine learning based IDS models,” *J. Sci. Ind. Res.*, vol. 81, no. 03, pp. 276–286, 2022.
111. M. P. Novaes, L. F. Carvalho, J. Lloret, and M. L. Proença Jr, “Adversarial Deep Learning approach detection and defense against DDoS attacks in SDN environments,” *Futur. Gener. Comput. Syst.*, vol. 125, pp. 156–167, 2021.
112. G. Bovenzi, G. Aceto, D. Ciuonzo, V. Persico, and A. Pescapé, “A hierarchical hybrid intrusion detection approach in IoT scenarios,” in *GLOBECOM 2020-2020 IEEE global communications conference*, IEEE, 2020, pp. 1–7.
113. K. A. Alissa *et al.*, “Crystal structure optimization with deep-autoencoder-based intrusioan detection for secure internet of drones environment,” *Drones*, vol. 6, no. 10, p. 297, 2022.
114. S. Alzahrani and L. Hong, “Detection of distributed denial of service (DDoS) attacks using artificial intelligence on cloud,” in *2018 IEEE world congress on services (SERVICES)*, IEEE, 2018, pp. 35–36.
115. H. Ding, L. Chen, L. Dong, Z. Fu, and X. Cui, “Imbalanced data classification: A KNN and generative adversarial networks-based hybrid approach for intrusion detection,” *Futur. Gener. Comput. Syst.*, vol. 131, pp. 240–254, 2022.
116. R. K. Batchu and H. Seetha, “On improving the performance of DDoS attack detection system,” *Microprocess. Microsyst.*, vol. 93, p. 104571, 2022.
117. S. Khanam, I. Ahmedy, M. Y. I. Idris, and M. H. Jaward, “Towards an effective intrusion detection model using focal loss variational autoencoder for internet of things (IoT),” *Sensors*, vol. 22, no. 15, p. 5822, 2022.
118. L. Riddell, M. Ahmed, and P. Haskell-Dowland, “Establishment and mapping of heterogeneous anomalies in network intrusion datasets,” *Conn. Sci.*, vol. 34, no. 1, pp. 2755–2783, 2022.
119. R. A. Ramadan, A.-H. Emara, M. Al-Sarem, and M. Elhamahmy, “Internet of drones intrusion detection using deep learning,” *Electronics*, vol. 10, no. 21, p. 2633, 2021.