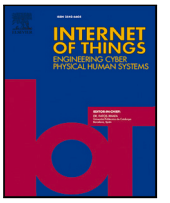
Internet of Things 28 (2024) 101336

Contents lists available at ScienceDirect

Internet of Things

journal homepage: www.elsevier.com/locate/iot



Research article

A novel deep learning-based intrusion detection system for IoT DDoS security

Selman Hizal [a,*], Unal Cavusoglu [b], Devrim Akgun [b]

[a] Sakarya University of Applied Sciences, Computer Engineering Department, Esentepe Campus, Serdivan, Sakarya, 54050, Turkey
[b] Sakarya University, Software Engineering Department, Esentepe Campus, Serdivan, Sakarya, 54050, Turkey

ARTICLE INFO

Dataset link: https://www.unb.ca/cic/datasets/iotdataset-2023.html

Keywords:
Intrusion detection system
Deep learning
IoT security
DDoS

ABSTRACT

Intrusion detection systems (IDS) for IoT devices are critical for protecting against a wide range of possible attacks when dealing with Distributed Denial of Service (DDoS) attacks. These attacks have become a primary concern for IoT networks. Intelligent decision-making techniques are required for DDoS attacks, which pose serious threats. The range of devices connected to the IoT ecosystem is growing, and the data traffic they generate is continually changing; the need for models more resistant to new attack types and existing attacks is of research interest. Motivated by this gap, this paper provides an effective IDS powered by deep learning models for IoT networks based on the recently published CICIoT2023 dataset. In this work, we improved the detection and mitigation of potential security threats in IoT networks. To increase performance, we performed preprocessing operations on the dataset, such as random subset selection, feature elimination, duplication removal, and normalization. A two-level IDS using deep-learning models containing binary and multiclass classifiers has been designed to identify DDoS attacks in IoT networks. The effectiveness of several deep-learning models in real-time and detection performance has been evaluated. We trained fully connected, convolutional, and LSTM-based deep learning models for detecting DDoS attacks and sub-classes. According to the results on a partially balanced sub-dataset, two staged models performed better than baseline models such as DNN (Deep Neural Networks), CNN (Convolutional Neural Networks), LSTM (Long Short Term Memory), RNN (Recurrent Neural Network).

1. Introduction

The Internet of Things (IoT) advancements have changed several sectors by improving communication and data sharing. However, this growth also introduces security challenges as IoT networks become susceptible to cyber threats. Ensuring the security and integrity of IoT ecosystems has emerged as a recent concern for organizations benefiting from interconnected devices [1,2]. An overview of the security framework employed to safeguard IoT devices from DDoS attacks is shown in Fig. 1. At its core, the taxonomy addresses multiple layers of defense, encompassing both cloud-based and local security measures. Cloud security and firewall policies serve as the initial lines of defense, preventing unauthorized access and ensuring the integrity of data flow. Including seven Raspberry Pi 4 devices as local attackers highlights the vulnerability at the edge, emphasizing the need for robust internal security mechanisms. The utilization of Gigamon Network G-TAP-ATX for network monitoring and integrating a Netgear unmanaged switch GS308 demonstrates a proactive approach to detecting and mitigating potential threats at the network level. Furthermore, Fig. 1 underscores the significance of intrusion detection through deep learning models, emphasizing the importance

* Corresponding author.
E-mail addresses: selmanhizal@subu.edu.tr (S. Hizal), unalc@sakarya.edu.tr (U. Cavusoglu), dakgun@sakarya.edu.tr (D. Akgun).

https://doi.org/10.1016/j.iot.2024.101336
Received 27 April 2024; Received in revised form 21 July 2024; Accepted 21 August 2024
Available online 29 August 2024

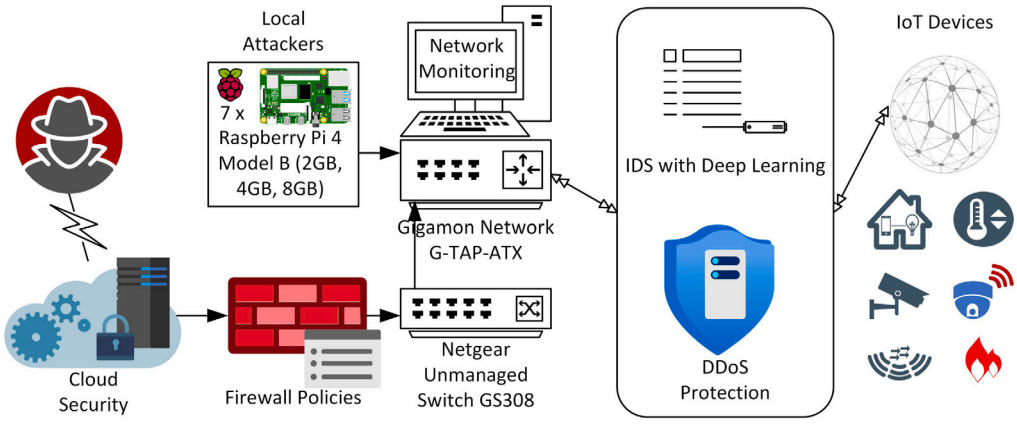


Fig. 1. DDoS attack taxonomy for IoT devices [3].

of advanced technologies in safeguarding several IoT devices from evolving cyber threats. This comprehensive security framework aims to create a resilient defense against DDoS attacks targeting IoT ecosystems.

This paper presents a comprehensive IoT security framework that addresses the need for robust security measures in IoT networks. Our proposed framework incorporates an IDS empowered by deep learning models, aiming to enhance the detection and mitigation of potential security threats. The foundation of our framework lies in collecting and preprocessing a diverse dataset of IoT network traffic data, encompassing both normal and abnormal activities. We ensure the dataset's consistency for further study by conducting preprocess steps such as data cleaning, standardization, and suitable formatting. We employ recent deep-learning models to achieve accurate and efficient IDS. Specifically, we consider Deep Neural Network (DNN), Long Short-Term Memory (LSTM), and Convolutional Neural Network (CNN). These models can capture important patterns in the IoT network traffic data, enabling effective identification of abnormal activities and possible security threats. The performance of the trained models is then assessed using an independent test dataset. Implementing our proposed IoT security design may help enterprises improve their security, protect IoT ecosystems, and minimize potential attacks. Integrating deep learning models within the IDS provides a robust system to detect and respond effectively for evaluating security threats in IoT networks. An intrusion detection system is proposed to provide IoT network security by using deep learning models on the CICIoT2023 dataset. Preprocessing, which eliminates unimportant features, removes the random selection process and removes duplication and log normalization, is performed on the dataset to improve the performance of the proposed model. After that, 3 different sub-datasets were generated for binary, 3-class, and 12-class classification. In the proposed two-stage detection model, it is first determined whether the traffic is an attack or not, and then the DDoS sub-attack type is determined. Performance and speed tests of the proposed models were performed.

Recent studies reveal that machine learning and deep learning models are popular in the design of IDS [4,5]. These systems have been developed using deep-learning models and extensively tested on IDS datasets. ML and DL techniques have become increasingly popular due to their demonstrated effectiveness in a wide range of applications [6–9]. Given the importance of computational efficiency in IoT devices, there is a growing interest in exploring more effective approaches and designing systems that can achieve efficient results. However, improved IDS systems need to be reconsidered with IoT data sets containing traffic data of new devices [10]. Additionally, these systems need to be updated to perform successful testing. Through this research, we aim to contribute to advancing IoT security and empower organizations to embrace the full potential of interconnected IoT devices while maintaining a secure environment. Performing performance tests with different deep learning algorithms on newly created and larger data sets generates new concepts for future study. The contributions of this work can be summarized below:

- An IDS model has been proposed for IoT networks, specifically targeting DDoS attacks and their subclasses for devices.
- A two-stage classification model has been proposed where the first stage is used for binary classification and the second for multiclass classification.
- The real-time detection performance of the proposed IDS has been measured using different deep-learning models.

The remaining of this paper can be outlined as follows: Section 2 summarizes an overview of the recent studies, while Section 3 presents the background of the study. Section 4 introduces the developed IDS, providing detailed insights into its functionalities. The system's performance tests were conducted, and the corresponding results are explicated in Section 5. The last section contains a thorough evaluation conducted to assess the overall efficiency of the proposed system.

2. Related work

In this section, studies carried out in the literature in recent years using machine learning methods and related data sets are examined. Studies using deep learning models are especially focused on this. The implementation details of some of these studies

Table 1
Recent literature about IDS for IoT networks.

Author	Year	Dataset	Method	Accuracy	Classification type
Sharma et al. [11]	2023	UNSW-NB15	DNN, GAN-DNN	84.00, 91.00	Multiclass
Bertoli et al. [12]	2023	Bot-IoT, UNSW-NB15, TON-IoT, CSE-CIC-IDS-2018	Deep Autoencoder	93.00, 74.00, 97.00, 98.00	Binary
Yao et al. [13]	2023	UNSW-NB15, CIC-IDS2017	BiGAN	80.10, 82.30	Multiclass
Thiyam and Dey [14]	2023	CIC-DDoS2019, Edge-IIoT	Hybrid model	99.86, 99.16	Binary
Nguyen and Le [15]	2023	BoT-IoT, CIC-IDS-2018, CIC-IDS-2017	SOCNN, LOF, INNE	98.94, 91.68, 96.07	Binary
Hnamte and Hussain [16]	2023	CICIDS2018, Edge-IIoT	DCNNBiLSTM	100, 99.64	Multiclass
Zhao et al. [17]	2023	SJTU-AN21, ISCXVPN2016, CIC-IoT2022	FlowTransformer	86.00, 95.20, 98.50	Multiclass
Vishwakarma and Kesswani [18]	2023	NF-UQ-NIDS	1D CNN	93.74	Multiclass
Alzughaibi and Khediri [19]	2023	CSE-CIC-IDS2018	MLP-BP, MLP-PSO	98.41, 95.32	Multiclass
Thakkar et al.[20]	2023	NSL-KDD, CIC-IDS_2017, UNSW_NB-15, BoT-IoT	DNN	98.90, 98.74, 96.70, 98.99	Binary
Jose et al.[21]	2023	CIC-IDS 2017	CNN-DNN-LSTM	94.61, 97.67, 98.61	Binary
Rizvi et al.[22]	2023	CIC-IDS2017, CSE-CIC-IDS2018	1D-DCNN	99.7, 99.98	Binary
Kamaldeep et al. [23]	2023	IoT-CIDDS	RF	98.6	Binary
Chaganti et al. [24]	2023	SDN-IoT, SDN-NF-TJ	DNN-CNN-LSTM-SVM	97.4, 97.7, 92.8, – 95.7, 96.0, 97.1, 85.77	Binary, Multiclass
Zhang et al. [25]	2023	CSE-CIC-IDS2018	ML-CNN	99.14, 99.89	Binary, Multiclass
Aravamudhan [26]	2023	NIDS	Fast R-CNN	99.5	Multiclass
Aldhyani et al. [27]	2023	CIC-DDoS2019	CNN–LSTM	100	Multiclass
Farhan et al. [28]	2022	CSE-CIC-IDS2018	LSTM	99	Multiclass
Bowen et al. [29]	2023	CIC-IDS2017, IoT-23, UNSW-NB15	CNN, BiLSTM	98.0, 76.34	Multiclass

are given comparatively in Table 1. At the end of the section, the motivation of the article is explained and an evaluation of related studies is presented.

Sharma et al. [11] used a filter-based feature selection approach to determine important features to detect attacks on IOT networks. They trained Generative Adversarial Networks (GANs) for synthetic data generation to increase the minority samples for balancing the classes in the dataset. They made comparisons using a network with fully connected layers and machine learning models. Bertoli et al. [12] presented an unsupervised federated learning approach for network intrusion detection. The central server has a global model, and the interacting devices use the global model for local training. Then, each device sends model parameters to the server for aggregation. The federated learning approach provides promising results for generalization among heterogeneous networks. Yao et al. [13] developed an unsupervised deep learning model based on Bidirectional Generative Adversarial Networks for anomaly detection in IoT networks. They used the Wasserstein distance in their model to boost detection performance and extensively tested the UNSW-NB15 and CIC-IDS17 datasets. Thiyam and Dey [14] proposed a hybrid method using TOMEK-Link and SMOTE to solve the class imbalance problem by resampling. The method computes the roc-values and selects important features. Using the refined features, they trained and tested their model on CICDDoS2019 and ML-EdgeIIoT datasets and presented comparative results. Nguyen and Le [15] proposed a staged detection model where the network traffic is classified as normal or attack. Their method first checks traffic using a soft-ordering CNN, and if the traffic is normal, the method uses isolation-based analysis and local outlier factor-based anomaly detection to check undefined attacks.

Hnamte and Hussain [16] developed a model that contains CNN and BiLSTM layers for intrusion detection. CNN layers extract distinctive features, and the BiLSTM layers capture associations between features. They evaluated their proposed approach with alternative models on the CICIDS2018 dataset. Zhao et al. [17] proposed a deep learning model called Flow Transformer to classify the flow sequences. Their design includes a multi-head attention mechanism and uses an RF-based optimization strategy to analyze

chosen features. They made performance evaluations on the CIC-IoT2022, ISCXVPN2016, and SJTU-AN21 datasets. Vishwakarma et al. [18] presented a novel approach for detecting anomalies in IoT networks. The authors proposed a 1D CNN model based on transfer learning to address security challenges in IoT devices. The real-time applicability of the model was validated by deploying it on a Raspberry Pi3 as an edge device. Alzughaibi and El Khediri [19] presented a cloud-based IDS that utilized DNNs with backpropagation and particle swarm optimization. The paper provided experimental details on the CSE-CIC-IDS2018 dataset and insights for further research in a cloud environment. Vigoya et al. [30] presented a new CoAP-IoT anomaly detection dataset (CIDAD) and validated it using five machine-learning algorithms. The CIDAD dataset proves to be effective for anomaly traffic detection in CoAP-IoT environments. Thakkar and Lohiya [20] addressed the challenge of class imbalance in datasets for IoT network intrusion detection by proposing a deep learning approach based on ensemble learning. The approach successfully improved the success of intrusion detection systems by utilizing the Bagging classifier and class weights.

Jose et al. [21] evaluated the effectiveness of deep learning algorithms in intrusion detection for IoT using the CIC-IDS 2017 dataset. Patel et al. [31] addressed cybersecurity challenges in IoT, analyzed research methodology and performance metrics, and noted the importance of advanced security measures. The findings provided details about the progress achieved with intrusion detection in IoT while recognizing challenges in matching the increasing rate of cyber attacks. Rizvi et al. [22] investigated an IDS model for environments where the resources are constrained. They employed a 1D-DCNN model with dilated convolution to increase the detection performance. Abed [32] explored the deep learning models for IoT intrusion detection by reviewing prior studies and evaluating model performance on new traffic data. The study demonstrated the effectiveness of deep learning in handling large datasets, identifying new threats, and improving intrusion detection accuracy, emphasizing the importance of integrating new datasets for model enhancement. Kamaldeep et al. [23] presented a feature engineering and machine learning approach for identifying DDoS attacks on IoT devices. The approach for designing network-specific characteristics for IoT devices is presented in the article. Chaganti et al. [24] proposed an LSTM model approach for detecting attacks in IoT environments using SDN-based IDS. The paper presented a comprehensive performance evaluation of deep learning models on two SDN-IoT datasets. Zhang et al. [25] proposed a new two-stage intrusion detection methodology that uses machine learning and deep learning algorithms to perform thorough penetration detection in network traffic data in the context of IoT on the CSE-CIC-IDS2018 dataset. In the first stage, the LightGBM algorithm compares six traditional machine learning algorithms with network traffic data divided into normal and abnormal. On samples anticipated to be abnormal in the first stage, the CNN conducts thorough attack class identification in the second step.

Selvam and Velliangiri [33] explained new trends developing with IoT technology and Industry 4.0. The study examined deep learning models to detect and identify attacks on Bot-IoT and CSE-CIC-IDS2022 datasets. Aravamudhan [26] created an effective and flexible IDS, Fast Region-Based Convolution Neural Network. The study used a stacked model that combined the Singular Value Decomposition (SVD) and Principal Component Analysis (PCA) techniques. Aldhyani and Alkahtani [27] constructed network IDS using robust deep learning models that can secure Agriculture 4.0 networks. The combination of LSTM and CNN architectures can be utilized to create an efficient and flexible intrusion detection system for identifying DDoS attacks on the CIC-DDoS2019 dataset. Farhan and Jasim [28] presented cyber security research as an important demand for monitoring infrastructures. The study utilized deep learning to analyze the CSE-CIC-IDS2018 dataset, which contains both normal and malicious traffic, and assess the LSTM deep learning model. Bowen et al. [29] investigated BLoCNet, a DL model that uses convolutional and BiLSTM layers together. According the test results BLoCNet architecture worked effectively throughout the IDS datasets. Okey et al. [34] developed a transfer learning IDS using CNN architecture. The researchers trained their system on CSE-CICIDS2018 and CIC-IDS2017. They used transfer learning models and the model averaging technique and selected the best models, namely MobileNetV3Small, InceptionV3, and EfficientNetV2B0, to construct an ensemble model.

Termos et al. [35] present the Graph Deep Learning Framework based on Centrality Measures (GDLC), which contains an algorithm for dynamically selecting the best appropriate centrality measures based on the network's topological attributes derived from traffic data. They are incorporated with AI techniques, notably deep learning models such as CNN, LSTM, and GRU. Nie et al. [36] suggest M2VT-IDS, a new multi-task IoT intrusion detection system with good detection accuracy. It combines a packet-wise representation of traffic and a two-stage multi-task learning architecture that includes a multi-view sharing network and a task-specific attention network to simultaneously conduct anomaly detection, attack identification, and device identification tasks. In this [37], a unique multi-module framework called MULTIBLOCK is proposed, which makes use of a multi-controller architecture based on SDN and machine learning. MULTI-BLOCK has remarkable performance, as demonstrated by thorough evaluation utilizing well-known IoT datasets and a variety of test situations. Multi-block is an important development in IoT network security that provides a stable and flexible defense against dynamic attacks. Li et al. [38] offer a Hybrid DoS Attack IDS (HDA-IDS) that combines signature-based detection with anomaly-based detection to identify known and new DoS/botnet attacks efficiently. This research also presents CL-GAN, a unique anomaly-based detection approach. It combines GAN and CNN-LSTM to identify malicious traffic and provide a baseline for typical behavior. According to experimental analysis, the HDA-IDS performs better than other IDSs in identifying botnet and DoS attacks. Nandanwar et al. [39] Introduce AttackNet, a powerful deep-learning model based on an adaptive CNN-GRU model that detects and classifies numerous botnet attacks in IIoT. The model is thoroughly tested using the most recent dataset, demonstrating its capacity to protect IIoT networks from complex attacks. The proposed model outperforms state-of-the-art IIoT anomaly detection methods on a real-time IoT device dataset, particularly in the N_BaIoT dataset. Neto et al. [3] introduced a new and comprehensive IoT attack dataset aimed at assisting in building security analytics in real-world IoT operations. To create the dataset, they conducted 33 attacks on an IoT architecture of 105 devices. There are seven attack categories: DDoS, DoS, Web-based, Recon, Spoofing, Brute Force, and Mirai. All attacks targeted IoT devices and were designed to threaten other devices.

When reviewing current research, it is clear that there has been progress and a variety of approaches to IDS for IoT networks. Research-based on the intensive use of deep learning models, including GANs, CNNs, LSTMs, and hybrid, to address challenges like class imbalance, feature selection, and anomaly detection. There are potentially promising results of generalization and detection accuracies across heterogeneous networks by the Federated Learning and Multi-Task Learning frameworks. Comparative studies using public datasets show enhanced performance metrics like accuracy, precision, recall, and F1-score, proving the effectiveness. The establishment of new models and datasets, such as BLoCNet, M2VT-IDS, and HDA-IDS, highlights how IDS is always evolving to counter advanced cyber threats. Together, the evaluated works emphasize the need for more sophisticated machine learning methods and extensive dataset evaluations to provide a more resilient and flexible protection for IoT environments.

2.1. Motivation

Detecting DDoS attacks is crucial for real-world scenarios due to the increasing frequency and sophistication of these attacks. As highlighted in the Cloudflare DDoS Threat Report for 2024 Q2 [40], there was a significant 20% year-over-year increase in DDoS attacks, with a notable rise in attacks carried out by state-level actors and sophisticated cybercriminals leveraging advanced tools like generative AI. These attacks can severely disrupt online services, causing substantial financial and reputational damage to businesses. The report indicates that while most DDoS attacks are relatively small and short-lived, their potential impact on unprotected systems remains devastating. Therefore, implementing robust DDoS detection and mitigation strategies, such as those utilizing deep learning techniques, is essential for safeguarding digital infrastructure and ensuring the availability and reliability of online services.

Although the literature reviewed in the article generally focuses on cybersecurity threats in IoT networks, there are some research gaps and potential research areas. The evaluated studies on IDS employing deep learning models in IoT networks show the strategies researchers have used to support security. Each approach contributes uniquely to the evolving area of intrusion detection, from feature selection and different deep learning models to addressing noisy data and introducing novel optimization techniques. As the field advances, integration of machine learning approaches becomes essential to fortify IDS systems against the dynamic and evolving threats in IoT environments. The model's comparison with the recent research is shown in Table 1, including the data set, method, performance test results, and classification type used for IDS in IoT networks. As seen in the literature comparison, the data sets used in studies such as UNSW-NB15, Bot-IoT, TON-IoT, Edge-IIoT, CSE-CIC-IDS-2018 generally focused before 2023 or specific use cases [11,13,16,20,21]. Therefore, there is a need for new data sets that reflect the current traffic of IoT devices and include different types of threats. In our study, CiCIoT2023, which contains up-to-date and comprehensive traffic for IoT networks, was preferred.

Due to the large amount of traffic in IoT networks, it is essential to develop systems that require low resource consumption, can work in real-time, and can react quickly. In the studies examined in Table 1, binary [12,14,15,20,21,23] or multiclass [11,13,16–19,29] structure was preferred as classification. Unlike many studies in the literature, binary and multiclass structures are used together in our research. The proposed two-stage detection model aims to detect quickly and at a lower cost. First, performing binary classification makes the proposed model lighter and more scalable than other models. In our study, prediction times were analyzed considering different package sizes for binary and multiclass classification.

In the literature review, it was seen that machine learning algorithms [3,12,23] and some of the deep learning methods [11,20,21,24,27,28] were generally used in the models used in systems developed for attack detection on IoT networks. In our study, DNN, CNN, and LSTM deep learning methods, which are widely used in the literature, were used, and binary and multiclass classification can be performed simultaneously by using two-stage architecture.

3. Background

IoT device security is ensured by IDS, which monitors network traffic and device behavior to detect and respond to threats. IDS for IoT devices employ different techniques to identify malicious activities and unauthorized access attempts, providing an additional defense against intrusions. One approach used in IDS for IoT devices is signature-based detection. This technique compares network traffic patterns or device behavior against known attack signatures or patterns. The IDS raises an alert to signal a potential intrusion when a match is found. Signature-based detection effectively identifies known attacks and quickly responds to well-documented threats. Another technique employed by IDS for IoT devices is anomaly-based detection. This method focuses on identifying deviations from normal or expected behavior. Anomaly-based detection is particularly useful for detecting novel or previously unseen attacks, as it does not rely on known signatures.

To enhance detection capabilities, some IDS for IoT devices utilize machine learning algorithms. These algorithms can analyze large volumes of network traffic and device data to detect patterns and anomalies. Machine learning-based IDS can be trained with updated datasets with new attacks to adapt and learn recent attack types. In addition to detection, IDS for IoT devices often incorporate response mechanisms to mitigate threats. They can trigger actions such as blocking suspicious network traffic, isolating compromised devices from the network, or alerting security administrators for further investigation and response. It is important to note that IDS for IoT devices should be designed with resource-constrained environments in mind. IoT devices usually come with restricted processing capability, memory, and energy resources. Therefore, IDS solutions for IoT devices should be lightweight, efficient, and optimized to minimize resource consumption while maintaining effective threat detection. IDS are critical for safeguarding the security of IoT devices. IDS can effectively identify and respond to potential intrusions by employing signature-based and anomaly-based detection techniques and machine learning algorithms. As IoT expands, further research and development in IDS technologies are needed to address the evolving threat landscape and protect IoT ecosystems.

Table 2
CICIoT2023 number of instances for 46 features.

Attack type	Number of instances	Attack type	Number of instances
DDoS-ICMP_Flood	7,200,504	DDoS-ACK_Fragmentation	285,104
DDoS-UDP_Flood	5,412,287	DNS_Spoofing	178,911
DDoS-TCP_Flood	4,497,667	Recon-HostDiscovery	134,378
DDoS-PSHACK_Flood	4,094,755	Recon-OSScan	98,259
DDoS-SYN_Flood	4,059,190	Recon-PortScan	82,284
DDoS-RSTFINFlood	4,045,285	DoS-HTTP_Flood	71,864
DDoS-SynonymousIP_Flood	3,598,138	VulnerabilityScan	37,382
DoS-UDP_Flood	3,318,595	DDoS-HTTP_Flood	28,790
DoS-TCP_Flood	2,671,445	DDoS-SlowLoris	23,426
DoS-SYN_Flood	2,028,834	DictionaryBruteForce	13,064
BenignTraffic	1,098,195	BrowserHijacking	5,859
Mirai-greeth_flood	991,866	CommandInjection	5,409
Mirai-udpplain	890,576	SqlInjection	5,245
Mirai-greip_flood	751,682	XSS	3,846
DDoS-ICMP_Fragmentation	452,489	Backdoor_Malware	3,218
MITM-ArpSpoofing	307,593	Recon-PingSweep	2,262
DDoS-UDP_Fragmentation	286,925	Uploading_Attack	1,252
		Total	46,686,579

3.1. Distributed Denial of Service (DDoS)

DDoS attacks in CICIoT2023 are classified under flooding, fragmentation, application layer, and mirai attacks. Flooding attacks are commonly used to target IoT devices as part of DDoS attacks. Flooding attacks overwhelm the network or resources of a target device, rendering it unable to function correctly. DDoS attacks that we used to classify with our two-staged model are explained as follows.

ICMP Flood is a cyber attack that inundates a target device with excessive ICMP Echo Request packets, overwhelming its resources and potentially causing a denial of service. UDP Flood is an attack using the User Datagram Protocol (UDP), which includes flooding a target system with UDP packets. A TCP flood attack is a cyber-attack where an assailant overwhelms a target system with excessive TCP connection requests. A PSH-ACK flood is a cyber-attack where a system is inundated with TCP packets bearing the PSH (Push) and ACK (Acknowledge) flags, overloading its resources. A SYN flood is a cyber-attack where an assailant inundates a target system with a barrage of TCP SYN packets, overwhelming its resources and preventing legitimate connections. A RST-FIN flood is a cyber attack that floods a target system with TCP packets bearing RST (reset) and FIN (finish) flags, disrupting connections and network stability. A Synonymous IP Flood is a cyber-attack in which an attacker floods a target server with modified TCP-SYN packets that all appear to come from the same source and destination IP address, consuming the server's resources. An ICMP fragmentation attack is a cyber-attack where the attacker sends manipulated ICMP packets with modified fragment offsets, exploiting target systems' fragment reassembly processes. A UDP fragmentation attack exploits weaknesses in a target system's handling of fragmented User Datagram Protocol (UDP) packets. An ACK fragmentation attack manipulates acknowledgment (ACK) packets to exploit vulnerabilities in a target system's communication process.

3.2. Deep learning models

Deep learning models are a large type of neural network with several layers that can extract features automatically from data, typically for classification or regression. There are parameters, such as problem type, dataset properties, and real-time considerations, to determine the architecture of the model. In practice, fully connected layers, convolution layers, recurrent layers, and derived models from these layers are combined to form a deep learning architecture. A typical deep neural network includes layers such as fully connected, convolutions, and recurrent layers and some non-trainable layers like activation functions, pooling operations, and flattening. The layers before the classification extract the essential features from the input data, and then the classification layer, which is a fully connected layer, classifies the elaborated features to determine the output.

3.3. IoT DDoS evaluation dataset (CiCIoT2023)

The Canadian Institute for Cybersecurity (CIC) has valuable contributions to cybersecurity. Through dataset provision, industry partnerships, and an IoT lab, they actively support research and initiatives to advance IoT security. The CICIoT2023 dataset directory contains subdirectories with original traffic captures in .pcap format, extracted features in .csv files for ML evaluation, and supplementary materials, including source code and tool descriptions. It provides a comprehensive resource for training and evaluating ML models in attack detection and classification. The dataset has 33 attacks, as shown in Table 2, and divided into 7 classes DDoS (12), DoS (4), Recon (5), Web-Based (6), Brute Force (1), Spoofing (2), Mirai (3) collected from an IoT environment consisting of 105 devices. We preferred the CICIoT2023 dataset because it contains a large amount of traffic belonging to different attack types from very different IoT devices compared to previous datasets, and training, testing, and verification operations were carried out on this dataset. Our focus is only on the recently released CICIoT2023 dataset that includes inputs for various IoT devices.

Table 3
IoT datasets and features.

Dataset name	Year	Number of devices	Attack types	Number of samples	Attack categories
CICIoT2023 [3]	2023	105	33	46,686,579	DDoS, DoS, Recon, Web-Based, Brute Force, Spoofing, Mirai
SDN-IoT [24]	2023	15	5	367,500	DDoS, DoS, Port Scanning, Fuzzing, OS Fingerprinting
Edge-IIoT [41]	2022	+10	14	2,219,210	Backdoor, DDoS_HTTP, DDoS_ICMP, DDoS_UDP, DDoS_TCP, Fingerprinting, MITM, Password, Port_Scanning, Ransomware, SQL_injection, Uploading, Vulnerability_scanner, XSS
TON-IoT [42]	2020	5	9	64,342	Backdoor, DDoS, Ransomware, Injection, XSS, Password, Scanning, DoS, MITM
IoT-23 [43]	2020	23	9	325,307,990	Attack, C&C, DDoS, FileDownload, HeartBeat, Mirai, Okiru, PartOfAHorizontalPortScan, Torii
Bot-IoT [44]	2019	9	6	73,360,990	Service scanning, OS Fingerprinting, DDoS, DoS, Keylogging, Data Theft

Table 3 shows the number of devices, attack types, and the total number of samples included in the IoT datasets created to date. Among the datasets, the most comprehensive and up-to-date CICIoT2023 [3] was preferred in our study. This dataset includes 105 devices and contains 33 different attack types and 46,686,579 samples under main categories such as DDoS, DoS, Recon, Web-Based, Brute Force, Spoofing, and Mirai. Therefore, it is the largest dataset in terms of both attack diversity and data volume. The CICIoT2023 dataset is particularly valuable for researchers and cybersecurity professionals, providing a wide range of scenarios for in-depth analysis and testing.

Comparing the CICIoT2023 dataset with other datasets in Table 3, the SDN-IoT [24] dataset includes 15 devices, 5 attack types and 367,500 samples. Although it covers basic attacks such as DDoS, DoS, Port Scanning, Fuzzing, OS Fingerprinting, it lacks the breadth and depth of CICIoT2023. The Edge-IIoT [41] dataset contains over 10 devices and 2,219,210 samples. The dataset covers 14 attack types, including DDoS types, SQL Injection, MITM, Malware, and Scanning as the main categories. However, despite this diversity, it is quite small compared to the sample size of CICIoT2023. The TON-IoT [42] dataset contains 64,342 samples and 9 attack types. It covers major attack types such as Backdoor, DDoS, Ransomware, Injection, XSS, Password, Scanning, DoS, and MITM. Its smaller scale and sample size limits a comprehensive analysis compared to CICIoT2023. With 23 devices and 9 attack types, the IoT-23 [43] dataset contains 325,307,990 samples and a range of attack types. However, it does not cover as many attack types as CICIoT2023. The Bot-IoT [44] dataset consists of 9 devices and 6 attack types and contains a total of 73,360,990 samples. Although the number of samples in this dataset is quite high, it is less than CICIoT2023 regarding device and attack diversity. CICIoT2023 is the most comprehensive dataset due to its coverage of 105 devices, the largest number of attack types 33, and a large sample size of 46,686,579. This makes it the most versatile and valuable dataset for IoT security research, surpassing all others in breadth and depth.

4. Proposed IDS for DDoS attacks

This section presents deep learning architecture for detecting DDoS attacks in IoT networks. First of all, preprocessing operations were carried out on the CICIoT2023 dataset and 3 different datasets were obtained to perform different classification processes. These obtained datasets were used for training and testing of the models developed to detect DDoS attacks. After obtaining the datasets, 5 different proposed models were explained in detail. The CICIoT2023 dataset preprocessing methods to improve performance are described, followed by a detailed explanation of the proposed model.

4.1. Preprocessing operations

In this section, the pre-processing steps used to obtain the data sets to be used in the training and testing of the proposed model are explained in detail. The sub-datasets required for binary and multiclass training from the CICIoT2023 raw data set are shown in Fig. 2. In order for the training and testing processes to produce more successful results, the data set was made more efficient by first deleting the less important attributes and repetitive records. As seen in Fig. 2, since the volume of the dataset is very large, random data selection was made from all attack types and normal traffic to create three different datasets that can represent the entire dataset for training and testing processes. Log normalization processes were applied to eliminate imbalances on the data set. Thus, more powerful and balanced datasets were obtained after preprocessing. The following are the preprocessing operations that were performed on the dataset, listed in order:

- **Eliminating unimportant features:** The presence of unimportant data in deep learning models causes performance to decrease and processing requirements to increase. Therefore, features that contain a very small number of samples or null values are eliminated. First, features containing zero or null values in the dataset were removed. Thus, 37 useful features were selected among 46 features. Table 4 shows the minimum, maximum, standard deviation, and mean values of the 37 selected features.

Table 4
Dataset description.

Features	Min-Max	Mean	Standard Dev
1	0–9905359.25	489 429.928	1 080 466.974
2	0–47	9.804	9.031
3	0–255	87.397	41.983
4	0–8388608	4157.466	51 540.253
5	0–8388608	4157.466	51 540.253
6	0–13368	0	0.008
7	0–1	0.026	0.16
8	0–1	0.107	0.309
9	0–1	0.041	0.199
10	0–1	0.042	0.2
11	0–1	0.446	0.497
12	0–7.7	0.058	0.201
13	0–12.52	0.569	0.776
14	0–82.4	0.085	0.515
15	0–4401.7	60.283	175.877
16	0–9586.5	443.533	971.371
17	0–1	0.053	0.224
18	0–1	0.312	0.463
19	0–1	0.001	0.034
20	0–1	0.001	0.026
21	0–1	0.655	0.475
22	0–1	0.141	0.348
23	0–1	0	0.022
24	0–1	0.061	0.239
25	0–1	0.999	0.031
26	0–1	0.999	0.031
27	50–120 892.1	4622.142	6073.947
28	42–13 583	176.957	267.898
29	42–41 814	986.247	1476.134
30	42–13 583	440.418	542.79
31	0–10 996.261	281.691	439.548
32	42–13 583	440.846	546.034
33	1–15	9.498	2.819
34	9.165–164.821	24.532	16.074
35	0–15 551.061	398.04	621.946
36	0–1	0.567	0.436
37	1–244.6	141.504	72.603

Table 5
CiC_IoT2023 number of instances for benign, Non-DDoS and DDoS.

Class (Label)	Number of instances	Number of random	Number of instances (Duplicates removed)
BenignTraffic	1,098,195	1,000,000	35
Non-DDoS	11,603,824	1,000,000	30,473
DDoS	33,932,344	1,000,000	94,974
Total	46,634,363	3,000,000	125,482

- **Random selection process:** It is important to decrease dataset size when working with large-scale datasets that are computationally costly. In large datasets, choosing closely balanced examples across each class helps reduce imbalances between classes during training. This operation is essential for reducing class imbalances, mainly when dealing with datasets containing numerous records. Randomly selecting samples contributes to a more balanced representation of classes and better model training by addressing potential biases. Then, since the dataset has 46,686,579 sample data and an imbalance between classes, sub-datasets containing 3M samples were randomly determined. The deep learning model was trained using a cleaned dataset. Table 5 shows the total number of Benign, Non-DDoS and DDoS attack types in the data set and the deleted duplicate removal numbers for 1 million randomly selected samples. In the study, a balanced data set was created by selecting 1 million random samples for each different class. In total, a balanced data set of 3 million was created from Benign, Non-DDoS and DDoS traffic types. Table 6 shows the sample numbers in the data set of DDoS sub-attacks and the 100,000 samples taken from each sub-attack type to create a balanced data set and the numbers of deleted duplicate records. Since duplicate records in the data set affect the accuracy of the results obtained in performance, this process was applied to all classes, and 1 million balanced and duplicate removed attack classes were obtained from DDoS attack types. Fig. 3 shows the distribution of the original dataset and the balanced dataset.
- **Remove duplication:** The full dataset consists of 46 features. Duplicate records may appear when these attributes are analyzed over 37 features. Therefore, the proposed deep learning modeling incorrectly using more than one identical record for training may negatively affect performance. By deleting the repetitive samples in this dataset, a sub-dataset containing 2,874,518 samples was obtained. Training was carried out in binary and multiclass formats on this data set.

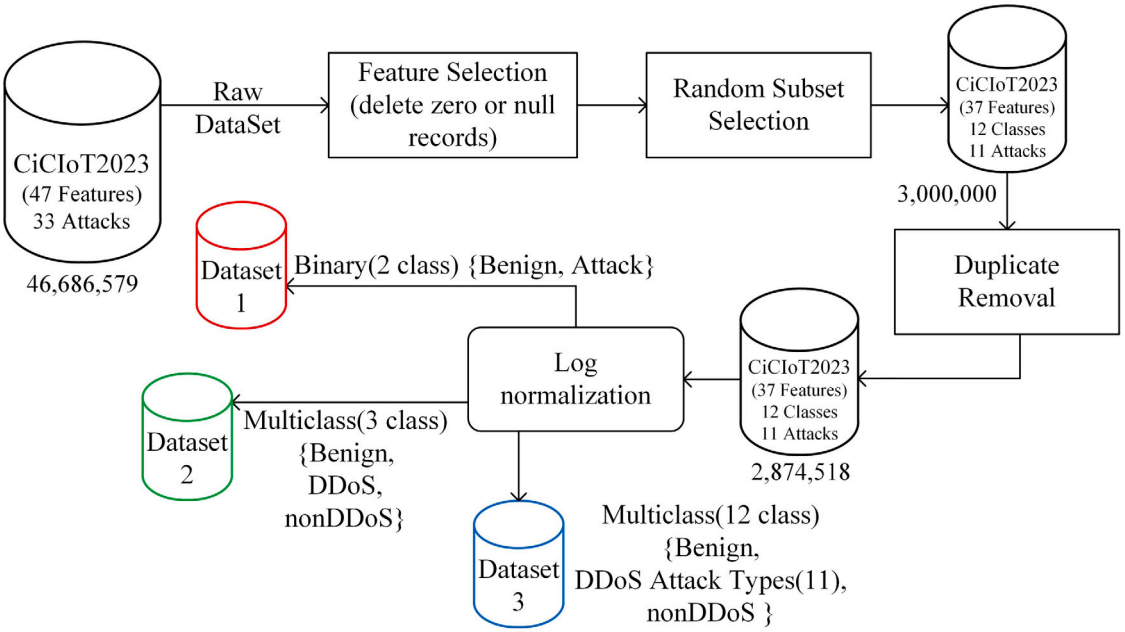


Fig. 2. CICIoT2023 dataset preprocessing stages.

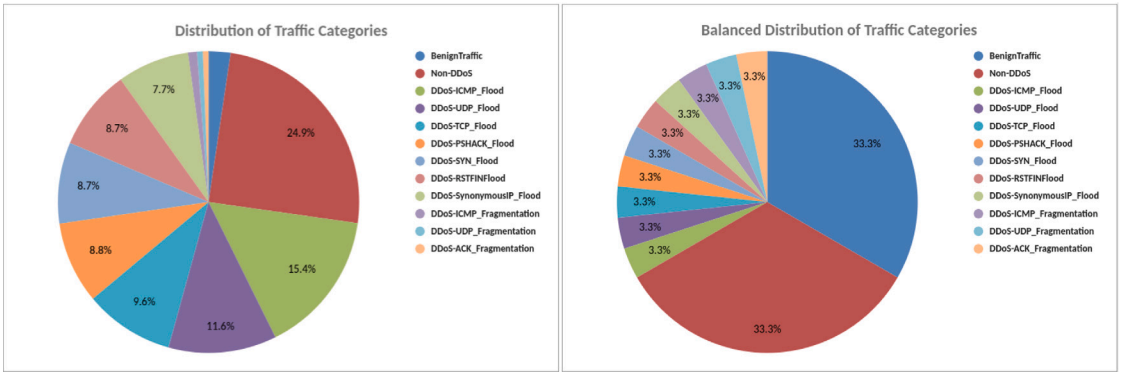


Fig. 3. CICIoT_2023 dataset distribution before and after balancing.

Table 6
CiC_IoT2023 number of instances for DDoS attacks.

Class (Label)	Number of instances	Number of random	Number of instances (Duplicates removed)
DDoS-ICMP_Flood	7,200,504	100,000	21,474
DDoS-UDP_Flood	5,412,287	100,000	0
DDoS-TCP_Flood	4,497,667	100,000	14,189
DDoS-PSHACK_Flood	4,094,755	100,000	16,511
DDoS-SYN_Flood	4,059,190	100,000	9975
DDoS-RSTFINFlood	4,045,285	100,000	24,468
DDoS-SynonymousIP_Flood	3,598,138	100,000	4910
DDoS-ICMP_Fragmentation	452,489	100,000	987
DDoS-UDP_Fragmentation	286,925	100,000	0
DDoS-ACK_Fragmentation	285,104	100,000	2460
Total	33,932,344	1,000,000	125,482

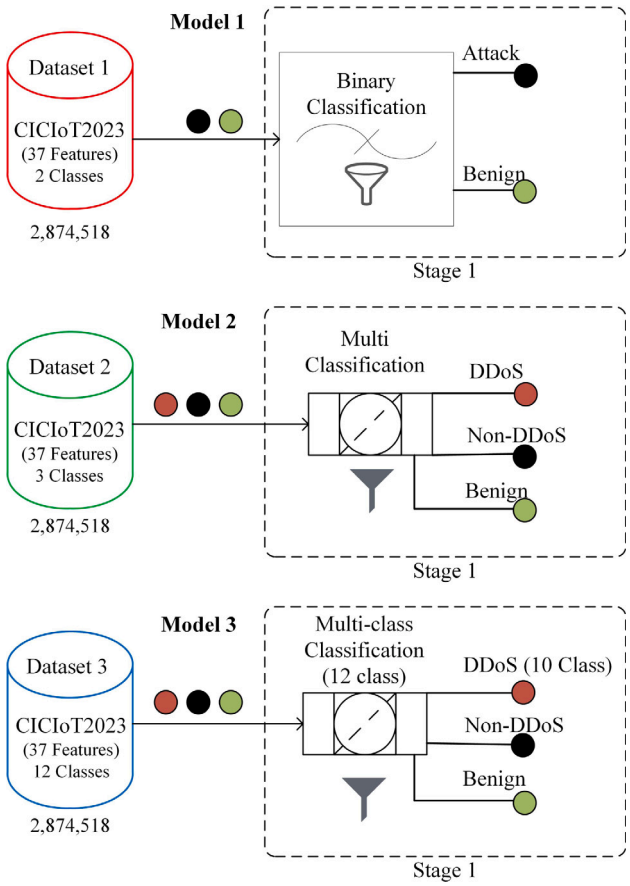


Fig. 4. Proposed one-stage system model for IoT security.

- **Log normalization:** The process of logarithmic normalization is a mathematical technique used to transform data that has a wide range of values into a normalized scale. This technique is particularly useful when dealing with data that follows a logarithmic distribution, where the magnitude of values increases exponentially. It is important to note that while the logarithmic scale technique is widely used in data preprocessing and normalization, it may not always generate new data with less variability or a more normal distribution.

4.2. Proposed IDS models

In the proposed study, we developed two-stage models for first detecting the attacks and then classifying them as DDoS and non-DDoS attacks. To establish the performance level, we tested alternative deep learning model structures with benign, DDoS, and nonDDoS attack types. We first implemented one-stage models for binary, three-class, and multiclass models. For the two-stage models, the three-class and multiclass models were implemented where, in both cases, the first stage uses a binary classifier, and the second stage differentiates between DDoS and non-DDoS attacks. Figs. 4 and 5 show the five different types of networks built in this work. Deep learning layers such as fully connected, convolutional, and LSTM layers were employed in each model for comparison. Therefore, there are totally five different models examined, and these are as follows:

- **Model 1**: works in binary mode to classify the network traffic as benign and attack. Dataset 1 was used to train and test Model 1 and binary classifiers of the two stage models.
- **Model 2**: implements the three-class classification, which separates the network traffic as benign, DDoS, and nonDDoS. Therefore, packages identified as attacks can be classified into DDoS and nonDDoS. Dataset 2, where attack traffic is defined as DDoS and nonDDoS, was used to train and test Model 2.
- **Model 3**: classifies the network traffic into 12 classes, with ten subclasses of DDoS attacks, one for non-DDoS and one for benign packets. Dataset 3, where DDoS attacks in the traffic are classified into ten subcategories, was used to train and test Model 3.

Table 7
Training parameters.

Parameter	Value
Optimizer	Adam
Learning rate	2e−4
Batch size	256
Epochs	200
Train ratio	60%
Test ratio	20%
Validation ratio	20%

Table 8
Summary of the deep learning model for 12-class classification using LSTM-based model.

Layer (type)	Output shape	Param #
in1 (InputLayer)	(None, 37, 1)	0
conv1d_3 (Conv1D)	(None, 35, 128)	512
lstm (LSTM)	(None, 35, 256)	394 240
lstm_1 (LSTM)	(None, 512)	1 574 912
batch_normalization_1 (BatchNormalization)	(None, 512)	2048
flatten_1 (Flatten)	(None, 512)	0
dense_9 (Dense)	(None, 512)	262 656
dropout_2 (Dropout)	(None, 512)	0
dense_10 (Dense)	(None, 512)	262 656
dense_11 (Dense)	(None, 12)	6156

- **Model 4**: is implemented in two stages: in the first stage, it classifies traffic as benign and attack, and in the second stage, it makes another binary classification according to the attack class for nonDDoS and DDoS. In the first stage, traffic was classified as benign and attack. If an attack is identified, the outputs of stage 1 are used and sent to the second stage to determine the attack type. In the second stage, the traffic detected as an attack was determined as DDoS or Non-DDos.
- **Model 5**: also classifies the DDoS attack traffic for its ten sub-attack types. Fully connected, convolutional, and LSTM-based deep learning methods were used to implement all models.
  In the last proposed model, training and testing were conducted on benign, non-DDoS, and DDoS (10 class) on dataset 3. As a result of the classification process, it is determined whether the traffic is benign, non-DDoS, or which DDoS attack class. DNN, CNN, and LSTM deep learning methods were used in the model. This model provides us with a more detailed output about traffic.

Table 7 shows the hyperparameters for the training procedure. We selected Adam [45] as the optimization method to train network weights. Although the number of epochs is fixed, we used $ModelCheckpoint$ callback in Keras to select the model with the highest validation accuracy, which is 20% of the samples. At the end of each model training, we use a test dataset, which is also 20% of the samples, for evaluating the model performance. In the original set, the number of samples per class is imbalanced and some classes have large numbers of samples as was described in Table 2. The effect of the classes with a relatively small number of samples on the accuracy is therefore small. The drop in the accuracy results for the partially balanced dataset is mostly because of the increased inequality of the number of samples per class in the dataset. We used the same dataset to evaluate the proposed models to increase the performance. Tables 8–10 show the architectural details such as the layer type, the number of trainable parameters, and the number of units per layer of the LSTM, CNN, and DNN models for the evaluation of the dataset. Table 10 a shows the DNN-based model, which just contains fully connected layers. DNN-based model is the basic implementation for most of the practical solutions in deep learning and machine learning. There are also more complicated layers like CNN and LSTM for better feature extractions. As shown by Tables 8 and 9, these layers are usually located previous to the classifiers with fully connected layers. In the LSTM-based layer, we also used a CNN layer for preprocessing. Note that these models implement the multiclass classifiers with softmax output layers and implement the binary classifier by just replacing the softmax layer with sigmoid for the output layer. These models were used to implement both single-stage and multistage models given by Figs. 4 and 5, respectively.

5. Experimental results

In the experiments, we used the CICIoT2023 raw data set and the sub-dataset given in Fig. 2 with increased class balance. We trained and evaluated the deep learning models using TensorFlow 2.13 [46] and Python 3.11 on the Ubuntu 20.04 operating system. The computational hardware includes a GeForce RTX 3060 12 GB NVIDIA® graphics card, and 32 GB of main memory. First, we repeat the results for binary and 34-class classification evaluations using the Random Forest algorithm, which provided the best results in the paper where Neto et al. introduced the CICIoT2023 dataset [3]. In this experiment, we also used Random Forest and all of the 46,686,579 samples as in the reference paper and applied the preprocessing operations mentioned before. To train the dataset which has a large number of samples, we allocated about 150 GB of additional swap memory. Therefore, it slows down the training procedure and takes tens of hours to train the network. Following the training, Random Forest produced an accuracy of

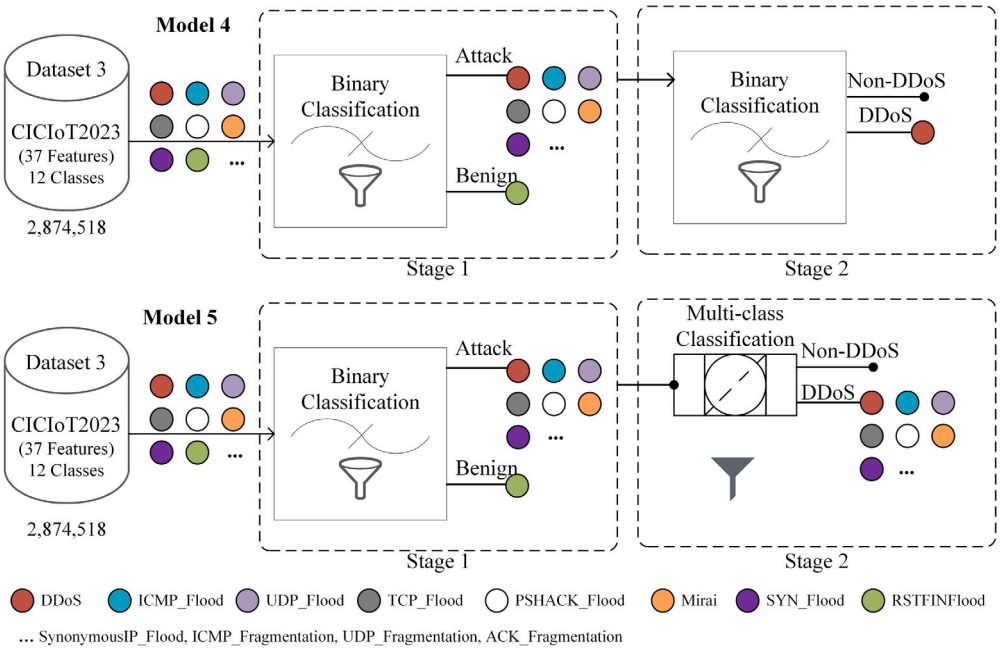


Fig. 5. Proposed two-stage system model for IoT security.

Table 9
Summary of the deep learning model for 12-class classification using CNN-based model.

Layer (type)	Output shape	Param #
in1 (InputLayer)	(None, 37, 1)	0
conv1d (Conv1D)	(None, 35, 128)	512
conv1d_1 (Conv1D)	(None, 33, 256)	98 560
conv1d_2 (Conv1D)	(None, 31, 512)	393 728
batch_normalization (BatchNormalization)	(None, 31, 512)	2048
flatten (Flatten)	(None, 15 872)	0
dense_6 (Dense)	(None, 512)	8 126 976
dropout_1 (Dropout)	(None, 512)	0
dense_7 (Dense)	(None, 512)	262 656
dense_8 (Dense)	(None, 12)	6156

Table 10
Summary of the deep learning model for 12-class classification using DNN-based model.

Layer (type)	Output Shape	Param #
in1 (InputLayer)	(None, 37)	0
dense (Dense)	(None, 64)	2432
dense_1 (Dense)	(None, 128)	8320
dense_2 (Dense)	(None, 256)	33 024
dense_3 (Dense)	(None, 512)	131 584
dropout (Dropout)	(None, 512)	0
dense_4 (Dense)	(None, 512)	262 656
dense_5 (Dense)	(None, 12)	6156

99.73% and 99.76% for the depths of 15 and 25, respectively. Random Forest also produced an accuracy of 99.16% for 34 class classification models. In the second experiment, we used the processed dataset as described previously in Fig. 2. In this case, Random Forest produced 94.03% accuracy for the binary classification model and 89.01% accuracy for the 34-class classification model.

Table 11 shows the performance of single-stage and multi-stage models using DNN, CNN, and LSTM network types. The performance of the Model 1 which makes binary classification shows accuracy over 94.5% and LSTM-based implementation reaches to 94.96% accuracy. The CNN-based model also obtains close results to the LSTM model and for True Positive Rate (TPR) and False Positive Rate (FPR) it performs better. The single stage three class classifier implemented with Model 2 performs about 90% accuracy for all models and it performs best with the CNN-based model providing 90.85% accuracy. When the number of classes is increased to 12 for including subclasses of DDoS attacks as in Model 3, the LSTM-based model performed best accuracy and other metrics for the same model. In the case of two-stage classifiers, the first classifier works as a binary classifier for both models. If any attack is

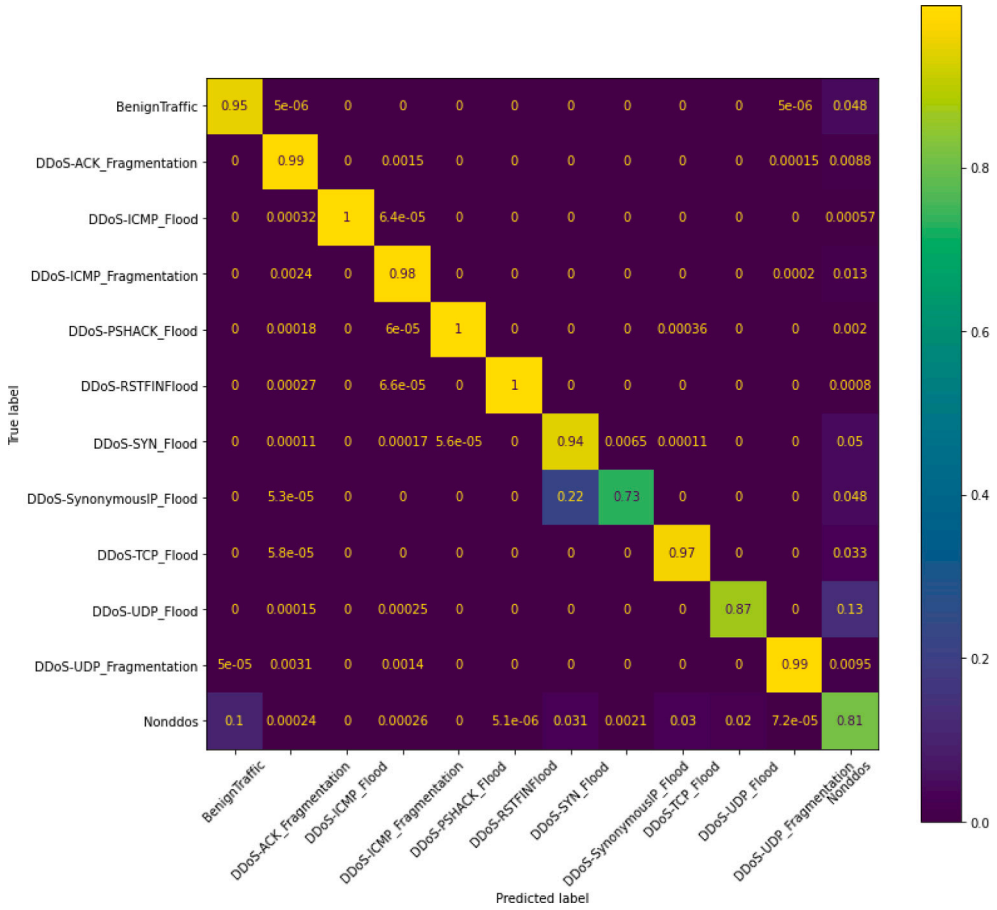


Fig. 6. Normalized confusion matrix for the two-stage LSTM-based model. The two-stage model includes 12 classes: a 10-class for DDoS attacks, 1-class for non-DDoS attacks, and a Benign class.

detected the second model in Model 4 makes another binary classification for detecting the attack type as DDoS and nonDDoS. This structure provided 91.22% accuracy when both stages are LSTM-based classifiers. In Model 5, the subclasses of DDoS attacks are also detected using a multiclass classifier. Therefore the second stage is only used in the case of attacks in both Models 4 and 5.

Tables 12–14 show the class-based performance metrics for the two-stage classifiers. While LSTM-based classifiers in the two-stage classifiers experiments perform best, there are differences among models when the class-based performances are examined. For example, for the class of DDoS-ICMP_Flood, the performance rates are 0.9991, 0.9989, and 0.9985 for the models CNN, DNN, and LSTM respectively. The class of DDoS-SynonymousIP_Flood is the worst-performing class for all classifiers. When compared to the single-stage models, the two-stage implementation provided an increase in all measured metrics. In general, the LSTM-based model performs best among two-stage models for all evaluated metrics. Fig. 6 shows the normalized confusion matrix for the two-stage model with the LSTM layers (see Fig. 6).

The performance metric for correctly classifying attacks provides us with useful information about the success of the developed model. In practice the inference times of the model for the attack detection models is another concern for real-time implementations. Table 15 shows the inference times for binary and multiclass classifiers used in the first and second stage of the two-stage classifiers respectively. The best-performing model for the single package and 10-package analysis is the CNN model for binary and multiclass classifiers respectively. But when the number of packages increased to 100 or more DNN produced shorter inference times for the experimented cases. In general, LSTM performance degrades as the number of packages to be processed grows. This is due to its computationally expensive recurrent structure. Note that the performance results may differ according to hardware; especially the version of the GPU device.

6. Discussion

The experimental results show that the AI-based IDS can accurately detect possible attacks in IoT networks. The system's capacity to adapt and learn from patterns helps to ensure effective intrusion detection, demonstrating its potential as a traditional system. One of the main problems with IoT networks is their vulnerability to DDoS attacks. The proposed IDS recognizes and categorizes

Table 11
Performance of DL models using CiCIoT2023, undersampled with random selection.

Model name	Model Type	ACC	TPR	FPR	Precision	F_score
Model 1: Binary classifier	DNN	0.94598	0.94546	0.05454	0.93715	0.94104
	CNN	0.94878	**0.94947**	**0.05053**	0.93966	0.94419
	LSTM	**0.94966**	0.94869	0.05131	**0.94163**	**0.94497**
Model 2: 3-class classifier	DNN	0.90326	0.90437	0.04849	0.90319	0.90260
	CNN	**0.90855**	**0.90960**	**0.04580**	**0.90883**	**0.90768**
	LSTM	0.90281	0.90373	0.04876	0.90351	0.90177
Model 3: 12-class classifier	DNN	0.89888	0.93399	0.01134	0.91042	0.91667
	CNN	0.90135	0.93493	0.01114	0.91090	0.91752
	LSTM	**0.90832**	**0.94255**	**0.01010**	**0.91108**	**0.92121**
Model 4: 2-stage 3-class classifier	DNN	0.90764	0.90869	0.04626	0.90760	0.90695
	CNN	0.91072	0.91172	0.04471	0.91096	0.90987
	LSTM	**0.91221**	**0.91332**	**0.04393**	**0.91218**	**0.91147**
Model 5: 2-stage 12-class classifier	DNN	0.90053	0.93493	0.01114	0.91090	0.91752
	CNN	0.90644	0.93493	0.01114	0.91090	0.91752
	LSTM	**0.91273**	**0.94304**	**0.00959**	**0.91458**	**0.92329**

Table 12
Class-based performance results for 12-class two-stage DNN model.

Traffic type	TPR	FPR	Precision	F-score
BenignTraffic	0.94374	0.05283	0.90502	0.92397
DDoS-ACK_Fragmentation	0.98370	0.00006	0.99813	0.99086
DDoS-ICMP_Flood	0.99898	0.00000	0.99987	0.99943
DDoS-ICMP_Fragmentation	0.98111	0.00006	0.99830	0.98963
DDoS-PSHACK_Flood	0.99742	0.00000	0.99988	0.99865
DDoS-RSTFINFlood	0.99841	0.00000	1.00000	0.99920
DDoS-SYN_Flood	0.94209	0.01828	0.62453	0.75112
DDoS-SynonymousIP_Flood	0.72608	0.00108	0.95821	0.82615
DDoS-TCP_Flood	0.95714	0.01005	0.74521	0.83799
DDoS-UDP_Flood	0.88230	0.00815	0.79598	0.83692
DDoS-UDP_Fragmentation	0.98480	0.00004	0.99888	0.99179
nonDDoS	0.81217	0.04550	0.90101	0.85429

Table 13
Class-based performance results for 12-class two-stage CNN model.

Traffic type	TPR	FPR	Precision	F-score
BenignTraffic	0.95191	0.05788	0.89767	0.92399
DDoS-ACK_Fragmentation	0.98877	0.00031	0.99126	0.99002
DDoS-ICMP_Flood	0.99911	0.00000	0.99987	0.99949
DDoS-ICMP_Fragmentation	0.98232	0.00010	0.99713	0.98967
DDoS-PSHACK_Flood	0.99772	0.00001	0.99982	0.99877
DDoS-RSTFINFlood	0.99887	0.00000	0.99987	0.99937
DDoS-SYN_Flood	0.94175	0.01801	0.62787	0.75343
DDoS-SynonymousIP_Flood	0.73066	0.00088	0.96606	0.83203
DDoS-TCP_Flood	0.97974	0.01099	0.73244	0.83823
DDoS-UDP_Flood	0.88375	0.00788	0.80162	0.84069
DDoS-UDP_Fragmentation	0.98835	0.00026	0.99272	0.99053
nonDDoS	0.80117	0.03896	0.91295	0.85342

DDoS attacks while increasing the accuracy of the model with the proposed two-stage deep learning approach. For the first stage we determined whether the incoming traffic is attack or not. In the second stage, we identified the DDoS and nonDDoS attacks. IoT network security experts must identify attack classes to respond to present threats effectively, forecast future risks, and improve their cybersecurity capabilities. In the second two stage model, we also determined the subclasses of the DDoS attacks.

When the studies in Table 16 are examined, the 2-stage model was applied in our research on the CICIoT2023 dataset. It is seen that the datasets in the current articles do not contain current IoT traffic [13,21] in Table 1. We used the CICIoT2023 dataset that contains up-to-date IoT traffic. According to Tables 1 and 16, it can be seen that either only binary [14,20,23] or only multiclass [16–18] classification is preferred in the literature. In contrast to several studies in the literature, our research combines both binary and multiclass systems. Our study investigated prediction times by considering a range of packages for both binary and multiclass classification. The comparison results show no significant difference in prediction times between 1–100 packages. It has been observed that when the number of packages increases logarithmically, there is a substantial increase in calculation times.

Recognizing certain limitations related to detecting DDoS for IoT networks is critical. IoT devices often need higher bandwidth and patchy network access. This can make the data collection process and communication difficult. Data from IoT devices is usually

Table 14
Class-based performance results for 12-class two-stage LSTM model.

Traffic type	TPR	FPR	Precision	F-score
BenignTraffic	0.95417	0.04930	0.91167	0.93244
DDoS-ACK_Fragmentation	0.98800	0.00010	0.99721	0.99258
DDoS-ICMP_Flood	0.99853	0.00000	0.99994	0.99923
DDoS-ICMP_Fragmentation	0.98707	0.00021	0.99400	0.99052
DDoS-PSHACK_Flood	0.99886	0.00001	0.99964	0.99925
DDoS-RSTFINFlood	0.99914	0.00001	0.99980	0.99947
DDoS-SYN_Flood	0.95260	0.01747	0.63760	0.76390
DDoS-SynonymousIP_Flood	0.74302	0.00092	0.96509	0.83962
DDoS-TCP_Flood	0.97074	0.01030	0.74317	0.84185
DDoS-UDP_Flood	0.89825	0.00823	0.79738	0.84482
DDoS-UDP_Fragmentation	0.99100	0.00019	0.99463	0.99281
nonDDoS	0.81961	0.03641	0.91987	0.86685

Table 15
Prediction times for number of packages.

Model		Prediction times (ms)				
		Single packages	10-packages	100-packages	1000-packages	10 000-packages
Binary classification	DNN	58.98070	58.04759	**67.15446**	**123.39751**	**685.40920**
	CNN	**55.44567**	**55.78399**	69.67783	130.09667	820.88017
	LSTM	63.13028	62.28703	82.27034	233.13702	1751.03737
Multiclass classification	DNN 11-class	56.51215	57.88882	**64.81108**	**127.70649**	**654.41554**
	CNN 11-class	**54.31986**	**53.85590**	69.77391	130.69773	820.99557
	LSTM 11-class	59.83589	60.19802	79.00174	229.49213	1735.31520

Table 16
Comparison of the study with the recent research.

Author	Year	Dataset	Method	Accuracy	Classification type
Neto et al. [3]	2023	CICIoT2023	LR, Perceptron, Adaboost, DNN, RF	98.90, 98.17, 99.58, 99.68, 99.44 83.16, 86.66, 35.13, 99.43, 99.11	Binary, Multiclass
Abbas et al. [47]	2023	CICIoT2023	CNN, RNN, LSTM, BiLSTM,DL-BiLSTM	92.21,92.73, 92.75,93.05,93.13	Multiclass
Wang et al. [48]	2024	CICIoT2023	DNN, CNN,RNN	84.73,94.30, 95.89	Multiclass
Nkoro et al. [49]	2024	CICIoT2023	CNN-LSTM,DNN,RNN,CNN-BiLSTM	87,88,93, 94.878, **99.0**	Multiclass
Our Model	2024	CICIoT2023	DNN, CNN, LSTM, RF, RF(unbalanced)	94.598, 94.878, 94.966,94.03, **99.76**	Binary
Our Model	2024	CICIoT2023	Two Stage-(DNN, CNN, LSTM)	89.888, 90.644, **91.273**	Multiclass

large and can change rapidly. This volume and dynamism of data can make them difficult to process and analyze effectively. The IoT ecosystem comprises various devices, and manufacturers have different standards and communication protocols. This heterogeneity makes it difficult to develop a standardized DDoS detection solution. As attackers constantly develop new and advanced DDoS attack methods, it can be difficult for DDoS detection systems to keep up with this rapid change. Machine learning-based DDoS detection systems generally require large amounts of training data. However, limited and heterogeneous data from IoT devices may pose a challenge in providing training data effectively.

7. Conclusion

Artificial intelligence-based detection systems are widely used to ensure security in IoT networks. In this paper, we analyzed the CICIoT2023 database used in the study, and DDoS attack types are explained in detail. Three sub-datasets were created from the high-volume CICIoT2023 dataset to train and test the proposed models. Preprocessing operations such as unimportant feature elimination, duplication removal, and log normalization were applied to these datasets. Five different models were developed for the CICIoT2023 dataset. Unlike the studies in the literature, experimental results were obtained on the two-stage model. A 2-stage deep learning model is proposed to detect DDoS attacks faster for IoT devices. In the two-stage model, firstly, it is determined whether the traffic is an attack or not, and then the subtypes of the DDoS attack are determined. Thus, IDS can be done more efficiently and quickly. According to the test results, the proposed CNN-based two-stage model has an accuracy of 91.27%. The packet processing time is 55.44 msec for binary and 54.31 msec for multiclass.

Future research with realistic limitations should focus on developing more efficient deep learning models optimized for edge devices to improve the suggested IDS. Edge devices frequently have limited processing power and memory capacity. Developing deep learning models that function effectively under these limits while maintaining performance remains a serious issue. The IoT ecosystem includes a wide range of devices from multiple manufacturers, each with unique hardware specs and communication protocols. Designing deep learning models that can tolerate this variability while remaining compatible and efficient offers a

significant challenge. Deep learning models, such as those employed in IDS, are vulnerable to adversarial attacks, in which hostile actors intentionally modify input data to fool the system. Developing strong models to survive such attacks while retaining high detection accuracy is an ongoing research topic.

CRediT authorship contribution statement

Selman Hizal: Writing – review & editing, Writing – original draft, Software, Methodology, Investigation. **Unal Cavusoglu:** Writing – review & editing, Writing – original draft, Validation, Software, Methodology, Investigation. **Devrim Akgun:** Writing – review & editing, Writing – original draft, Methodology, Investigation.

Declaration of competing interest

We declare that there is no conflict of interest. We have no financial or personal relationships with any individuals or organizations that could inappropriately influence our work.

Data availability

The CIC-IoT 2023 dataset, which was utilized to back up the study's conclusions, is available at: https://www.unb.ca/cic/datasets/iotdataset-2023.html.

References

[1] S.A. Bakhsh, M.A. Khan, F. Ahmed, M.S. Alshehri, H. Ali, J. Ahmad, Enhancing IoT network security through deep learning-powered intrusion detection system, Internet Things 24 (2023) 100936.
[2] A. Lara, V. Mayor, R. Estepa, A. Estepa, J.E. Díaz-Verdejo, Smart home anomaly-based IDS: Architecture proposal and case study, Internet Things 22 (2023) 100773.
[3] E.C.P. Neto, S. Dadkhah, R. Ferreira, A. Zohourian, R. Lu, A.A. Ghorbani, CICIoT2023: A real-time dataset and benchmark for large-scale attacks in IoT environment, Sensors 23 (13) (2023).
[4] M.M. Inuwa, R. Das, A comparative analysis of various machine learning methods for anomaly detection in cyber attacks on IoT networks, Internet Things 26 (2024) 101162.
[5] I. Ahmad, Z. Wan, A. Ahmad, A big data analytics for DDOS attack detection using optimized ensemble framework in Internet of Things, Internet Things 23 (2023) 100825.
[6] A. Aldhaheri, F. Alwahedi, M.A. Ferrag, A. Battah, Deep learning for cyber threat detection in IoT networks: A review, Internet Things Cyber-Phys. Syst. (2023).
[7] B. Madhu, M.V.G. Chari, R. Vankdothu, A.K. Silivery, V. Aerranagula, Intrusion detection models for IOT networks via deep learning approaches, Meas.: Sens. 25 (2023) 100641.
[8] C. Alex, G. Creado, W. Almobaideen, O.A. Alghanam, M. Saadeh, A comprehensive survey for IoT security datasets taxonomy, classification and machine learning mechanisms, Comput. Secur. 132 (2023) 103283.
[9] F.D. Keersmaeker, Y. Cao, G.K. Ndonda, R. Sadre, A survey of public IoT datasets for network security research, IEEE Commun. Surv. Tutor. (2023) 1.
[10] B. Kaur, S. Dadkhah, F. Shoeleh, E.C.P. Neto, P. Xiong, S. Iqbal, P. Lamontagne, S. Ray, A.A. Ghorbani, Internet of things (IoT) security dataset evolution: Challenges and future directions, Internet Things (2023) 100780.
[11] B. Sharma, L. Sharma, C. Lal, S. Roy, Anomaly based network intrusion detection for IoT attacks using deep learning technique, Comput. Electr. Eng. 107 (2023) 108626.
[12] G. de Carvalho Bertoli, L. Alves Pereira Junior, O. Saotome, A.L. dos Santos, Generalizing intrusion detection for heterogeneous networks: A stacked-unsupervised federated learning approach, Comput. Secur. 127 (2023) 103106.
[13] W. Yao, H. Shi, H. Zhao, Scalable anomaly-based intrusion detection for secure internet of things using generative adversarial networks in fog environment, J. Netw. Comput. Appl. 214 (2023) 103622.
[14] B. Thiyam, S. Dey, Efficient feature evaluation approach for a class-imbalanced dataset using Machine learning, Procedia Comput. Sci. 218 (2023) 2520–2532, International Conference on Machine Learning and Data Engineering.
[15] X.-H. Nguyen, K.-H. Le, Robust detection of unknown DoS/DDoS attacks in IoT networks using a hybrid learning model, Internet Things (2023) 100851.
[16] V. Hnamte, J. Hussain, DCNNBiLSTM: An efficient hybrid deep learning-based intrusion detection system, Telemat. Inform. Rep. 10 (2023) 100053.
[17] R. Zhao, Y. Huang, X. Deng, Y. Shi, J. Li, Z. Huang, Y. Wang, Z. Xue, A novel traffic classifier with attention mechanism for industrial Internet of Things, IEEE Trans. Ind. Inform. (2023) 1–12.
[18] M. Vishwakarma, N. Kesswani, A transfer learning based intrusion detection system for Internet of Things, 2023, preprint.
[19] S. Alzughaibi, S. El Khediri, A cloud intrusion detection systems based on DNN using backpropagation and PSO on the CSE-CIC-IDS2018 dataset, Appl. Sci. 13 (4) (2023).
[20] A. Thakkar, R. Lohiya, Attack classification of imbalanced intrusion data for IoT network using ensemble-learning-based deep neural network, IEEE Internet Things J. 10 (13) (2023) 11888–11895, http://dx.doi.org/10.1109/JIOT.2023.3244810.
[21] J. Jose, D.V. Jose, Deep learning algorithms for intrusion detection systems in internet of things using CIC-IDS 2017 dataset, Int. J. Electr. Comput. Eng. (IJECE) 13 (1) (2023) 1134–1141.
[22] S. Rizvi, M. Scanlon, J. McGibney, J. Sheppard, Deep Learning Based Network Intrusion Detection System for Resource-Constrained Environments, in: The 13th EAI International Conference on Digital Forensics and Cyber Crime, in: ICDF2C '22, Springer, New York, NY, USA, 2022, pp. 1–16.
[23] Kamaldeep, M. Malik, M. Dutta, Feature engineering and machine learning framework for DDoS attack detection in the standardized Internet of Things, IEEE Internet Things J. 10 (10) (2023) 8658–8669, http://dx.doi.org/10.1109/JIOT.2023.3245153.
[24] R. Chaganti, W. Suliman, V. Ravi, A. Dua, Deep learning approach for SDN-enabled intrusion detection system in IoT networks, Information 14 (1) (2023) http://dx.doi.org/10.3390/info14010041.
[25] H. Zhang, B. Zhang, L. Huang, Z. Zhang, H. Huang, An efficient two-stage network intrusion detection system in the Internet of Things, Information 14 (2) (2023).
[26] P. Aravamudhan, A novel adaptive network intrusion detection system for internet of things, PLoS One 18 (4) (2023) e0283725.

[27] T.H.H. Aldhyani, H. Alkahtani, Cyber security for detecting distributed denial;of service attacks in agriculture 4.0: Deep;learning model, Mathematics 11 (1) (2023).
[28] B.I. Farhan, A.D. Jasim, Performance analysis of intrusion detection for deep learning model based on CSE-CIC-IDS2018 dataset, Indones. J. Electr. Eng. Comput. Sci. 26 (2) (2022) 1165–1172.
[29] B. Bowen, A. Chennamaneni, A. Goulart, D. Lin, BLoCNet: a hybrid, dataset-independent intrusion detection system using deep learning, Int. J. Inf. Secur. (2023) 1–25.
[30] L. Vigoya, A. Pardal, D. Fernandez, V. Carneiro, Application of machine learning algorithms for the validation of a New CoAp-IoT anomaly detection dataset, Appl. Sci. 13 (7) (2023).
[31] S. Kalpesh Patel, S. Sadhwani, R. Muthalagu, P. Mothabhau Pawar, Deep learning based intrusion detection systems techniques in IoT - survey, in: 2023 International Conference on Computational Intelligence and Knowledge Economy, ICCIKE, 2023, pp. 53–58, http://dx.doi.org/10.1109/ICCIKE58312.2023.10131739.
[32] W.M. Abed, Deep learning-based Internet of Things intrusion detection, Eurasian Res. Bull. 19 (2023) 47–57.
[33] R. Selvam, S. Velliangiri, Iotsdl: Internet of things security for deep learning techniques-A research perspectives, in: 2023 International Conference on Computer Communication and Informatics, ICCCI, 2023, pp. 1–7, http://dx.doi.org/10.1109/ICCCI56745.2023.10128558.
[34] O.D. Okey, D.C. Melgarejo, M. Saadi, R.L. Rosa, J.H. Kleinschmidt, D.Z. Rodríguez, Transfer learning approach to IDS on cloud IoT devices using optimized CNN, IEEE Access 11 (2023) 1023–1038, http://dx.doi.org/10.1109/ACCESS.2022.3233775.
[35] M. Termos, Z. Ghalmane, A. Fadlallah, A. Jaber, M. Zghal, et al., GDLC: A new graph deep learning framework based on centrality measures for intrusion detection in IoT networks, Internet Things (2024) 101214.
[36] F. Nie, W. Liu, G. Liu, B. Gao, M2VT-IDS: A multi-task multi-view learning architecture for designing IoT intrusion detection system, Internet Things 25 (2024) 101102.
[37] A.A. Toony, F. Alqahtani, Y. Alginahi, W. Said, MULTI-BLOCK: A novel ML-based intrusion detection framework for SDN-enabled IoT networks using new pyramidal structure, Internet Things (2024) 101231.
[38] S. Li, Y. Cao, S. Liu, Y. Lai, Y. Zhu, N. Ahmad, HDA-IDS: A hybrid DoS attacks intrusion detection system for IoT by using semi-supervised CL-GAN, Expert Syst. Appl. 238 (2024) 122198.
[39] H. Nandanwar, R. Katarya, Deep learning enabled intrusion detection system for industrial IOT environment, Expert Syst. Appl. 249 (2024) 123808.
[40] O. Yoachimik, J. Pacheco, DDoS threat report for 2024 Q2, 2024, URL https://blog.cloudflare.com/ddos-threat-report-for-2024-q2.
[41] M.A. Ferrag, O. Friha, D. Hamouda, L. Maglaras, H. Janicke, Edge-IIoTset: A new comprehensive realistic cyber security dataset of IoT and IIoT applications for centralized and federated learning, IEEE Access 10 (2022) 40281–40306.
[42] N. Moustafa, M. Keshky, E. Debiez, H. Janicke, Federated TON_IoT windows datasets for evaluating AI-based security applications, in: 2020 IEEE 19th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom), IEEE, 2020, pp. 848–855.
[43] S. Garcia, A. Parmisano, M.J. Erquiaga, IoT-23: A labeled dataset with malicious and benign IoT network traffic, 2021, http://dx.doi.org/10.5281/zenodo.4743746.
[44] N. Koroniotis, N. Moustafa, E. Sitnikova, B. Turnbull, Towards the development of realistic botnet dataset in the internet of things for network forensic analytics: Bot-iot dataset, Future Gener. Comput. Syst. 100 (2019) 779–796.
[45] D.P. Kingma, J. Ba, Adam: A method for stochastic optimization, 2014, arXiv preprint arXiv:1412.6980.
[46] K. Contributors, Keras v2.5, 2023, URL https://keras.io/.
[47] Z. Wang, H. Chen, S. Yang, X. Luo, D. Li, J. Wang, A lightweight intrusion detection method for IoT based on deep learning and dynamic quantization, PeerJ Comput. Sci. 9 (2023) e1569.
[48] S. Abbas, I. Bouazzi, S. Ojo, A. Al Hejaili, G.A. Sampedro, A. Almadhor, M. Gregus, Evaluating deep learning variants for cyber-attacks detection and multi-class classification in IoT networks, PeerJ Comput. Sci. 10 (2024) e1793.
[49] E.C. Nkoro, J.N. Njoku, C.I. Nwakanma, J.-M. Lee, D.-S. Kim, Zero-trust marine cyberdefense for IoT-based communications: An explainable approach, 13 (2) (2024) http://dx.doi.org/10.3390/electronics13020276.