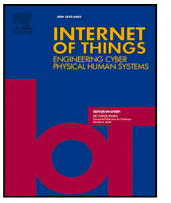
Internet of Things 30 (2025) 101527

Contents lists available at ScienceDirect

Internet of Things

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



Research article

DDoSViT: IoT DDoS attack detection for fortifying firmware Over-The-Air (OTA) updates using vision transformer

Muhammad Ali [a], Yasir Saleem [a], Sadaf Hina [b,*], Ghalib A. Shah [c]

[a] Computer Engineering, University of Engineering and Technology, Lahore, Pakistan
[b] School of Science, Engineering and Environment, The University of Salford, UK
[c] Department of Cybersecurity, Air University, Islamabad, Pakistan

ARTICLE INFO

Dataset link: DDoSViT (Original data)

Keywords:
Internet of Things (ioT)
Over-The-air (OTA) firmware update
Denial of service (doS)
Distributed doS (DDoS)
Vision Transformer (viT)

ABSTRACT

The widespread adoption of Internet of Things (IoT) devices has introduced numerous vulnerabilities, particularly in firmware over-the-air (OTA) updates. These updates are essential for improving device functionality and addressing security vulnerabilities. However, they have increasingly become the focus of distributed denial of service (DDoS) attacks designed to disrupt the update process. Historically, the infamous Mirai botnet and its variants have exploited IoT vulnerabilities to carry out successful DDoS attacks. In recent years, deep learning models, especially Vision Transformers, have gained significant attention due to their exceptional performance in image classification tasks. To optimize detection and alert mechanisms, this novel study proposes a DDoSViT framework. This Vision Transformer (ViT)-based multi-vector DDoS and DoS attack detection framework converts attack flows into images and trains Vision Transformers on an attack image dataset. To validate the proposed framework, this study extensively reviewed diverse datasets and selected CICIoT2023 and CICIoMT2024 datasets ensuring these contain real-world attack scenarios and multi-vector real attacks. The proposed methodology and rigorous experimentation demonstrated 99.50% accuracy in multi-class classification across 23 different variants of DDoS and DoS attacks, outperforming contemporary models. The model's performance was assessed using metrics such as accuracy, precision, recall, and F1-score. This research provides significant benefits to security practitioners and administrators, offering reduced false positives and reliable alerts during firmware over-the-air updates in IoT-edge devices.

1. Introduction

The Internet of Things (IoT), defined as an interconnection of the objects and devices to the internet, [1,2] transmit and receive real-time data. This technology encompasses a wide range of applications [3], from smart home devices such as thermostats, washing machines, and security cameras to industrial automation and healthcare devices for various critical operations. IoT devices are embedded with sensors and actuators that enable them to communicate with other devices, perform operations they are designed for, aggregate data, and be controlled remotely.

As IoT-based systems continue to evolve, their improved efficiency, ease of use, and data-driven decision-making in various applications have made it a significant technology of interest for both businesses and consumers. However, the rapid proliferation

* Corresponding author.
E-mail addresses: ali.mughal@kics.edu.pk (M. Ali), yasir@uet.edu.pk (Y. Saleem), s.hina@salford.ac.uk (S. Hina), ghalib.asadullah@au.edu.pk (G.A. Shah).

https://doi.org/10.1016/j.iot.2025.101527
Received 8 November 2024; Received in revised form 7 January 2025; Accepted 28 January 2025
Available online 5 February 2025

of IoT devices without appropriate security and privacy considerations also raises concerns regarding data management and communication [4].

IoT vulnerabilities [5,6] are inherent weaknesses in IoT devices which malicious attackers can exploit. These vulnerabilities often arise from inadequate security measures, such as inadequate authentication protocols, unencrypted communication, insufficient access controls and delayed firmware updates. A large number of IoT devices are deployed and operated with default passwords or lack scheduled firmware updates, making them a playground for cybercriminals. Furthermore [7], the multiplicity of IoT devices and the deficiency of the benchmarked security frameworks complicate the landscape. As a result, compromised IoT devices are exploited for a spectrum of malicious purposes, i.e. unauthorized access, data leaks, and even participation in larger-scale flooding attacks, in particular, distributed denial-of-service (DDoS) attacks.

Adversaries exploit vulnerabilities to access devices without authorization, interrupt services, and/ or exfiltrate critical data. These attacks can take various forms, including DDoS attacks, in which compromised IoT devices flood a victim with illegitimate traffic, disrupting the availability of critical services and resources. The interconnected feature of IoT devices envisages that a single compromised IoT device can provide an entry point into a larger network, potentially impacting numerous systems. Massive incidents, such as the Mirai botnet attack [8,9], have outlined the scale and severity of cyber-attacks in IoT, leading to increased scrutiny from regulators and the need for robust security measures in the design and implementation of IoT solutions.

From 2020 to 2024, several botnet campaigns were run by RaptorTrain [10,11]. The botnet targeted small office/ home office (SOHO) devices like D-Link IP Cameras, Netis, and ZTE routers on a large scale, spread via weak and default credentials and zero-day or one-day vulnerabilities, in legacy services such as Telnet and SSH. It compromised 60,000 SOHO and IoT devices at its peak. The attackers used a sophisticated command and control (C2) server, Sparrow, which incorporates a centralized Node.js backend and a cross-platform Electron front end to manage more than 60 C2 severs, and cracked Cisco Umbrella & Cloudflare Radar popularity list and infected nodes. Some of the malicious activities performed include exploitation, remote command execution, and IoT DDoS attacks.

The over-the-air (OTA) firmware updates [12] for IoT devices is a critical process for maintaining and enhancing functionality and patching security. OTA updates allow manufacturers to remotely push firmware updates to IoT devices without the requirement of physical access, which is essential for devices deployed in inaccessible locations. These updates can enhance features, correct glitches and patch security vulnerabilities, ensuring devices remain compliant with evolving standards and regulations. However, the implementation of OTA firmware updates must be carefully managed to avoid possible disruptions or vulnerabilities [13] that could arise during the update process. Secure authentication and encryption are vital during these updates to prevent unauthorized access or tampering, which can jeopardize the device and the broader network.

Cyber threats to firmware over-the-air (OTA) updates in edge devices, particularly from DoS, DDoS, [14] and MITM attacks, can significantly compromise device security and functionality. Distributed denial-of-service attacks can swamp IoT edge devices directly, overloading them with malicious and benign traffic that prevents them from receiving essential firmware updates. This hindrance/ delay in updates leaves unpatched devices vulnerable to adversarial exploits. Furthermore, with MITM attacks [15], if an adversary intercepts communications during the update process, they can manipulate the firmware delivered to the edge devices and compromise the integrity of the updates. This manipulation/ injection of malicious packets can potentially result in the installation of malicious software, enabling attackers to take control of the devices, exfiltrate sensitive data, do lateral movement, or disrupt operations. The combination of DDoS and MITM attacks can severely degrade the reliability and security of edge devices, underscoring the need for robust encryption techniques, secure authentication, and vigilant monitoring to protect the integrity of firmware updates and the devices themselves [16].

The Vision Transformer (ViT) [17] represents a paradigm shift in computer vision by leveraging the Transformer architecture, traditionally employed in natural language processing, to process visual data through a patch-based input mechanism. In contrast to convolutional neural networks (CNNs) [18], which depend extensively on local receptive fields and hierarchical feature extraction, ViT partitions images into fixed-size patches, treating each patch as an individual token. This allows the model to utilize self-attention mechanisms to capture global contextual relationships among patches, facilitating the extraction of intricate, long-range dependencies that are often critical in visual understanding tasks [19].

DDoS attack detection highlights the capabilities of the different machine learning models, including CNN, LSTM, KNN, and RF. CNNs [20] have been employed to analyze network traffic patterns by treating data as images, enabling efficient real-time detection through their ability to identify complex spatial hierarchies. LSTMs [21], on the other hand, excel in capturing temporal dependencies in time-series data, making them congruent for monitoring traffic patterns over time. Random Forest offers robustness and accuracy by utilizing ensemble learning to build multiple decision trees, providing insights into feature importance. In contrast, K-Nearest Neighbors [22] remains a straightforward yet effective method for anomaly detection, though it can be less efficient with larger datasets.

Hybrid approaches like LSTM-CNN [23] and CNN-LSTM [24] have gained traction in recent studies, combining the strengths of both CNNs and LSTMs. The LSTM component addresses temporal dynamics while the CNN extracts spatial features, allowing for comprehensive analysis of DDoS attacks that exhibit complex patterns across time, and data characteristics. These models demonstrate improved accuracy and detection rates in various scenarios, highlighting a trend towards more sophisticated architectures in the fight against DDoS attacks. Overall, the literature emphasizes a growing reliance on deep learning techniques to enhance detection capabilities and improve response times in network security.

ViT exhibits superior scalability concerning both dataset size and model parameters. It has shown significantly improved performance when trained on large datasets; a feature that contrasts with CNNs, which can suffer from diminishing returns or even performance degradation without careful architectural tuning. Additionally, the flexible nature of the ViT architecture accommodates

varying input resolutions and sizes more seamlessly than CNNs, which are typically constrained by fixed kernel sizes and pooling layers.

Moreover, ViT can effectively utilize advanced techniques from the NLP domain, such as transfer learning and self-supervised learning, enhancing its ability to generalize across diverse tasks and datasets. The integration of these methodologies allows ViT to learn rich, high-dimensional feature representations that are robust and contextually informed. As such, the Vision Transformer emerges as a compelling alternative to traditional CNNs, particularly in scenarios that demand a nuanced understanding of complex visual contexts and relationships.

The Vision Transformer (ViT) is an innovative approach to malware classification that treats [25] binary files as visual data by converting them into images [26] or spectrograms. Leveraging its self-attention mechanism, ViT effectively captures complex, global patterns within the input data, enhancing its potential to recognize multi-layered features indicative of specific malware families. Its scalability allows for the effective handling of large datasets [27], improving generalization and robustness against obfuscation techniques. While a promising and successful implementation requires careful consideration of input representation and computational resources, ViT represents a significant advancement in malware detection methodologies.

DoS and DDoS attacks targeting edge devices can have severe consequences including operational disruptions, security vulnerabilities, and resource exhaustion. These attacks can prevent edge devices from receiving critical OTA firmware updates leaving them exposed to exploitation and compromising their performance. With limited computational resources, edge devices can be overwhelmed, leading to data loss and degraded functionality, affecting overall network integrity. Additionally, such attacks may result in increased maintenance and recovery costs, as well as the potential for broader network compromise if attackers gain access without authorization. To mitigate these risks, security researchers in the organization must implement efficient security measures that account for the unique constraints of edge IoT devices.

In this study, the authors highlight significant advancements in deep learning for real-time detection and mitigation of evolving DDoS threats in IoT-Edge devices:

(1) Fortifying over-the-air (OTA) firmware updates to defend against evolving DoS/DDoS multi-vector attacks targeting IoT edge devices is critical. The proposed approach focuses on mitigating vulnerabilities that attackers could exploit during such incidents. Ensuring the secure reception of updates is vital to maintaining the device's firmware integrity and operational stability.
(2) The DDoSViT framework is specifically tailored for IoT-edge environments, offering a versatile and enhanced solution for real-time attack detection. This architecture is capable of identifying 23 different variants of DoS and DDoS attacks, making it a versatile tool for enhancing cybersecurity in resource-constrained settings.
(3) An efficient data normalization technique, Quantile-Transformation, robust against outliers to enhance detection accuracy, is proposed. This method allows the system to distinguish genuine attack patterns from noise effectively. By improving data quality, it supports reliable detection and response mechanisms.
(4) The system is trained, validated and tested on the latest CICIoT2023 [28] and CICIoMT2024 [29] datasets. These datasets reflect real-time IoT attack scenarios, ensuring the model adapts to the evolving threat landscape. The validation process confirms the DDoSViT architecture's practical effectiveness for real-world applications.

The remaining part of the paper is structured as stated: Section 2 examines the related work, while Section 3 briefly explains the proposed DDoSViT framework, data preprocessing techniques, underclass sample handling and overall architecture. Section 4 details experiments conducted on the CICIoT2023 and CICIoMT2024 datasets, comparing results and discussion with shallow machine learning methods. Section 5 bring to a conclusion, and Section 6 discusses the limitations in the framework and future directions.

2. Literature review

In [30], a Recurrent Neural Network algorithm is proposed for intrusion detection in IoT environments. It was trained and tested on the NSL-KDD dataset and achieved an accuracy of 87%. It suggested that future improvements could focus on incorporating optimization algorithms to enhance detection accuracy. Another researcher in [31], proposed a lightweight intrusion detection system (IDS) that combined hybrid feature selection using XGBoost and MaxPoolingID to enhance security for MQTT-enabled IoT systems. The algorithm captured complex patterns and extracted relevant features, tested in two scenarios: unidirectional flows and bidirectional flows of the MQTT attacks dataset. Results demonstrated outstanding performance, with accuracy, precision, recall, and F1-Score excelling to 90% in both scenarios. The bidirectional scenario outperformed unidirectional across all metrics, making it the preferred option for network traffic classification.

IoTProtect, a machine learning-based IDS, was proposed to enhance security for IoT devices. It used the TON_IoT dataset for training and testing. It achieved an impressive 99.999% attack detection accuracy, with only 0.001% FPR and 0% FNR, while also demonstrating excellent timing performance [32]. Another study, [33], proposed a resilient network intrusion detection system designed to improve security in industrial IoT (IIoT) networks. It assessed the potential of using Long Short-Term Memory and other ML models in detecting network traffic irregularities. The system was trained and tested using the EdgeIIoT-2021 dataset. Experimental results showed that the ERT-based IIoT-NIDS achieved a 99.93% detection accuracy, while the LSTM-based IIoT-NIDS achieved an accuracy of 99.85%, outperforming existing state-of-the-art NIDS.

Later a study [34], proposed an artificial neural network-based intrusion detection system for IoT devices, aimed at addressing authentication challenges. Utilizing a supervised learning algorithm, the system detected attacks and discarded classified threats. The ANN architecture includes input layers, hidden layers, and output layers, where data are processed with assigned weights and

activation functions. The incorporated approach effectively detected attacks, achieved 84% average precision and FPR less than 8% in 10-fold based cross-validation. It demonstrated the effectiveness and detection accuracy of the method, indicating its ability to enhance intrusion detection systems in large, heterogeneous datasets.

The authors [35] proposed a solution integrating deep learning-based IDS methods for IoT environments, employing a convolutional neural network (CNN). The approach involved extracting log information from the IoT system, including location and service data, to create an original feature set. This feature set was then enhanced and encoded into a digital matrix, which was used for training and detection in the CNN. The method has been evaluated using cross-validation, achieving an average accuracy of 98.9%.

An AI-based IPS/IDS, proposed by [36]. It was designed for real-time threat protection of IoT networks. It employed an ensemble features complexity reduction approach to drop irrelevant and redundant features from the dataset, enhancing model performance. The system utilizes the N-BaIoT dataset for training, enabling it to process network data at the IoT gateway and predict real-time attacks. A responsive background script based on machine learning analysis was employed for intrusion prevention. The Light Gradient Boosting Machine (LightGBM) classifier was used for decision-making, achieving an impressive accuracy of 99.9%. This proactive security framework represents a significant advancement in safeguarding the evolving IoT landscape.

In another interesting study, the proposed multitude of solutions employed Decision Tree and K-Nearest Neighbors classifiers to protect IoT devices from attacks, achieving a 100% detection rate, when combined. This approach demonstrated high effectiveness in enhancing IoT security [37]. The authors [38] critically reviewed the effectiveness of advanced deep learning models, comprised of an Autoencoder-Fully Connected Network (Autoencoder-FCN) and Fully Convolutional Network (FCN), in a network intrusion detection system (NIDS), using the CICIDS2017 dataset. Both models achieved over 97% accuracy, with FCN slightly outperforming Autoencoder-FCN while also demonstrating faster training times in local environments.

In [39], an ML-based IDS framework was proposed for IoT environments, evaluating the ten learning methods utilizing the TON_IoT attack dataset. The stacking-ensemble model emerged as the top performer, achieving 0.9971 Matthews correlation coefficient (MCC) scores for binary classification and multi-class classification, it achieved 0.9909. [40] addressed the critical issue of enhancing IoT network security by implementing anomaly-based IDS using various ML algorithms on the IoT network intrusion dataset. The results demonstrated high efficiency, achieving accuracy rates between 99% and 100% in detecting anomalies. [41] presented a robust system designed to detect network attacks targeting IoT devices by employing a CatBoost regression model on the IDS2017 dataset. The proposed approach achieved 92.5% accuracy and incorporated various attributes as key parameters for identifying intrusion attacks, demonstrating its effectiveness against the other contemporary methods.

The authors [42] addressed security threats in Software Defined Networking (SDN) for IoT devices by implementing a hybrid feature selection method (RF-RFE) combined with a fine-tuned ML model to detect anomalies. The results reveal that the RF-based RF-RFE-SDNIoT-NIDS achieved an accuracy of 99% in binary classification as well as multi-class classification using minimal features, outperforming other machine learning approaches. [43] presented a solution for detecting DDoS attacks in cloud computing, focusing on reducing misclassification errors. By employing effective feature selection techniques, RF Features importance and Mutual Information, and testing various algorithms, the study found that Gradient Boosting, KNN, Weighted Voting Ensemble, and RF achieved 99% accuracy, with RF excelling the best performance, misclassifying only one attack as normal.

The authors in another study [44] proposed a DDoS attack detection architecture integrated feature selection technique and the Random Forest Classifier, deployed on edge computing devices within the SDN environments. The experimental results on the CIC-DDoS2019 dataset demonstrate that the solution achieved an impressive accuracy of 99.99%, with a prediction time of just 0.4 s, outperforming other DDoS detection solutions. [45] addressed application layer DDoS attacks by inspecting packet characteristics such as HTTP frame size, IP address counts, and port mappings. Using a multilayer perceptron (MLP) deep learning algorithm, the proposed method achieved 98.99% efficiency in DDoS detection with a reduced FPR of 2.11%, outperforming ML classifiers like Decision Stump and Naïve Bayes. Another study in [46] proposed a real-time detection system for application layer DDoS attacks by employing RF Classifier and Multi-Layer Perceptron models, achieving 99.5% accuracy and significantly reduced prediction times through big data frameworks like Apache Spark. Another study addressed the vulnerabilities in Industrial IoT protocols and nodes to cyber-attacks due to their easy exploitation and transformation into attack vectors. It proposed a novel protocol combining federated learning (FL) with fog/edge computing to enhance security. This approach trained a global model using distributed datasets from various collaborators, overcoming data and communication limitations. By leveraging FL, the protocol significantly improved detection accuracy, reduced mitigation response time by 72%, and increased the cost of attacks by 2.7 times. The evaluation showed that the detection accuracy of the FL-based method was approximately 98%, comparable to centralized training. Overall, this method offered a robust solution for combating malicious codes in IIoT environments [47].

With the rise of IoT botnet DDoS attacks, IoT security has become a critical concern. Existing security measures often struggle against new variants of IoT malware, particularly zero-day attacks. [48] introduced a honeypot-based method utilizing machine learning for malware detection. By generating and analyzing data from IoT honeypots, the approach dynamically trained a machine learning model to accurately identify and mitigate zero-day DDoS attacks. This method represented a promising advancement in dealing with the challenges caused by emerging threats in IoT security.

[49] presents a Protocol Based Deep Intrusion Detection (PB-DID) solution, which creates a dataset from merging IoT traffic of the both UNSWNB15 and Bot-IoT datasets. PB-DID classified traffic into normal, DoS, and DDoS categories, dealing with the issues of class imbalance and overfitting. Using deep learning techniques, it achieved an attack-recognizing accuracy of 96.3%, enhancing the detection of IoT-based attacks.

In a recent research study, [50], the author presented an advanced IDS designed to classify DDoS attacks in IoT environments by integrating DL with multi-objective optimization. Traditional IDS methods struggle with the diverse data from IoT devices, so the proposed system employed a technique for dimensional reduction, Jumping Gene adapted NSGA-II and a CNN model enhanced by

LSTM techniques for classification. Tested on the CISIDS2017 dataset using a High-Performance Computer (HPC), the IDS achieved an accuracy of 99.03% and a reduction in training time using a five-fold scheme, outperforming existing machine learning-based IDS methods.

The authors addressed security challenges in edge computing, highlighting the vulnerability of edge servers to DDoS attacks. Traditional mitigation methods are inadequate, leading to the modeling of the edge DDoS mitigation (EDM) problem as NP-hard, with two proposed solutions: EDMOpti and EDMGame. Additionally, they leveraged deep learning, converting network traffic into images for analysis with a CNN, achieving 99.99% accuracy in attack classification and outperforming existing methods [51].

In a study, the author proposed that their approach integrated artificial intelligence, specifically deep learning models like CNNs. By transforming network traffic data into images, the methodology trained the ResNet model, achieving an impressive accuracy of 99.99% in the binary classification of DoS and DDoS attacks. Additionally, it recognized eleven different attack patterns with an average precision of 87%, outperforming existing cutting-edge solutions by 9% [52].

In [53], authors presented a customizable architecture exploiting SDN that employed various ML and DL models to classify transport layer and application layer DDoS attacks. By utilizing the real-time CICDoS2017 and CICDDoS2019 datasets, the models achieved over 99% accuracy in classifying unseen traffic. In a Mininnet-based simulated environment and the ONOS SDN controller, detection rates excelled to 98% for transport layer DDoS attacks and reached up to 95% for application layer attacks.

This study introduced a novel secure IoT framework, utilizing SDN, designed to detect vulnerabilities and recognize malicious traffic in IoT devices through session Internet Protocol payload analysis and counter. The DDoS attack detection component of the framework employed advanced models capable of identifying DDoS attacks in SD-IoT networks by inspecting various parameters under high traffic volumes. Implemented on an SDN controller, the framework was tested by generating substantial traffic from a compromised node, leading to successful detection and notification of attacks. Results demonstrate that the proposed framework [54] achieved high accuracy in early attack detection ranging from 98% to 100% and a low FPR.

A study navigated the challenges of mitigating DDoS attacks initiated by rogue wireless IoT devices flooding IoT servers [55]. The proposed security technique incorporated cloud computing and the SDN approach, introducing a novel solution called LEDEM, Learning Driven Detection Mitigation. LEDEM employed a semi-supervised ML model to detect and mitigate DDoS attacks. Evaluated in both a physical testbed and an emulated testbed. LEDEM demonstrated an improved accuracy rate of 96.28% in DDoS detection against cutting-edge solutions.

FlowGuard [56], a novel DDoS technique based on traffic variations, employed two ML models for attack identification and classification. To substantiate the robustness of these models, a large dataset was generated using the two SlowHTTPTest and BoNeSi DDoS attacks simulators and combined with the real-time CICDDoS2019 dataset. Results showed that the identification accuracy of the proposed LSTM model reached 98.9%, significantly surpassing four other established learning models. Additionally, the convolutional neural network achieved a classification accuracy of 99.9%. Furthermore, the models effectively met IoT delay requirements when installed on edge servers with increased computational capacity than personal computers.

[57] explored the use of ML models to classify DDoS flooding attacks in SDNs, which are increasingly vulnerable on account of their programmability and global network view. The study investigated various ML techniques, including classification and regression tree (CART), k-nearest neighbor (k-NN), quadratic discriminant analysis (QDA), and Gaussian Naïve Bayes (GNB), utilizing real-time experimental data including throughput, jitter and response time performance metrics from a mid-sized enterprise SDN emulated network in Mininet. DDoS attacks were simulated using the Low Orbit Ion Cannon (LOIC) for different protocols (HTTP, TCP, UDP). Results indicated that while all algorithms performed well, CART outperformed others with an average prediction accuracy of 98%, a prediction speed on observations per second was 5.3×10, a training time of 12.4 ms, and overall robustness in attack detection.

Recently, the authors [58] researched the Botnet detection activity in Home Automation devices, which have become increasingly targeted by DDoS attacks. A novel detection model leveraging DL, specifically a Bidirectional Long Short Term Memory Recurrent Neural Network (BLSTM-RNN), was employed as a solution. The model employed Word Embedding techniques to convert attack packets into tokens format for text recognition. The performance of the BLSTM-RNN was contrasted to a standard LSTM-RNN in detecting four multi-vector attacks associated with the Mirai Botnet, with evaluations focusing on accuracy and loss. Results indicated that while the bidirectional model incurred additional processing time per epoch, it demonstrated superior long-term performance.

The critical evaluation of the current literature on IoT DoS/DDoS attack detection for IoT Edge devices during firmware over-the-air updates faces many challenges. In extant studies, the researchers devised blockchain solutions as blockchain technology can provide tamper-proof transmission of the firmware. However, this cannot detect and mitigate zero-day and evolving multi-vector DoS/DDoS attacks. In addition, machine learning and deep learning-based employed solutions to detect DoS/DDoS attacks were also researched. Many of the presented models were trained and validated on outdated datasets or imbalanced class instances with limited attack detection capabilities. They also deficit the ability to treat data distribution and imbalance effectively. The researchers of this study acknowledged this research gap and proposed the cutting-edge DDoSViT framework to navigate the existing challenges. DDoSViT is a Vision Transformer (ViT) based deep learning model for fortifying firmware over-the-air updates for IoT Edge. The framework is robust against 23 different variants of DoS/DDoS attacks. It converts flow packets to images after performing a data standardization technique called Quantile Transformer (QT). QT is robust against un-normalized features which introduces biases and poor model performance. It treats imbalanced minority classes using Synthetic Minority Over-sampling TEchnique (SMOTE) which significantly improves multi-class classification. The training, validating and testing were performed on the CICIoT2023 and CICIoMT2024 datasets. It can detect zero-day, high and low volumetric DoS/DDoS attacks with high accuracy, reduced computational and memory requirements, excels in contemporary multi-class classification solutions and fortifies the firmware over-the-air updates.

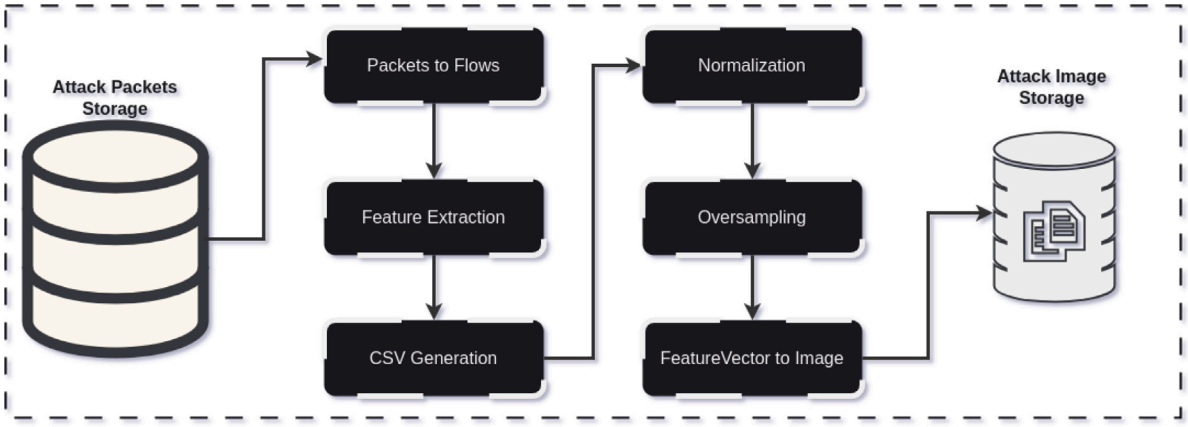


Fig. 1. DDoSViT preprocessing module.

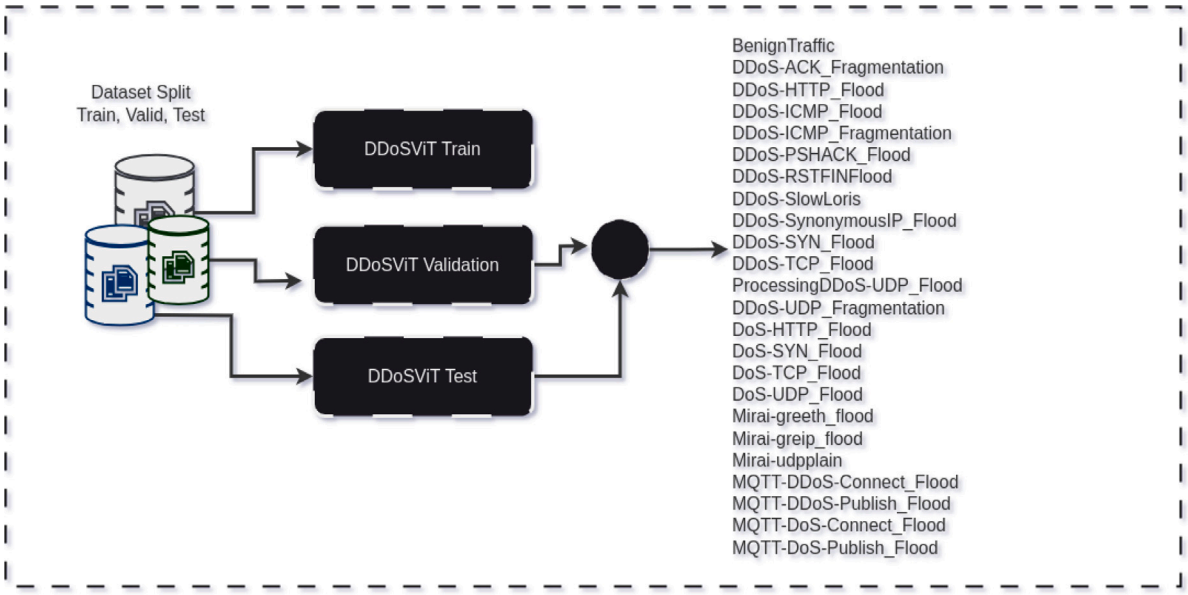


Fig. 2. DDoSViT deep learning module.

3. Proposed methodology

This study presents the DDoSViT framework, Figs. 1,2, a Vision Transformer (ViT) based approach for DoS and DDoS attacks detection that is deployed in IoT-Edge. The proposed DDoSViT framework encapsulates the learning of malicious patterns from traffic to enable the recolonization of DoS and DDoS patterns regardless of their temporal positioning. An essential benefit of ViT is to produce the same output irrespective of where a pattern appears in the input. This modularity and learning of features during training the model eliminates the need for rigorous feature engineering, ranking and selection. To support a real-time attack detection system, the researchers used a state-of-the-art pre-processing method for the network traffic that generates a spatial data representation used as input to the DDoSViT framework.

A. ***Convert Attacks PCAPs to Attacks CSVs*** Algorithm 1, essence from [28,29], is designed to process multiple cyberattack PCAP (Packet Capture) files in parallel, extract flow-level features from the packets, and generate labeled CSV files for each attack. The goal is to handle multiple large PCAPs efficiently and split them into manageable 20 MB chunks, Fig. 3 depicts parallel processing to extract relevant features, and finally labeling the data with the corresponding attack type before outputting the final CSV. This approach ensures both speed and scalability.
The algorithm begins by iterating through each attack PCAP file in the set. Each PCAP is split into smaller chunks of 20 MB to make the data easier to process and manage. These chunks are processed in parallel, leveraging up to 20 threads

Algorithm 1 Convert DDoS PCAPs to Labeled CSVs

Function: DDoSViT_toCSV(P, L)
Input:
  Set of cyberattack PCAPs: $P = \{P_1, P_2, \ldots, P_n\}$
  Label for each attack: $L = \{l_1, l_2, \ldots, l_n\}$
  Chunk size: 20 MB
  Number of parallel threads: 20
Output:
  Labeled CSV dataset for each attack D_{labeled}
for each attack PCAP $P_i \in P$ **do**
  Split the PCAP into 20 MB chunks:
    $P_{i,j} \leftarrow f_{\text{split\_pcap}}(P_i, 20MB)$
  for each chunk $P_{i,j}$ **do**
    Process chunks in parallel (20 threads):
      $F_{i,j} \leftarrow f_{\text{extract\_features}}(P_{i,j})$
      $C_{i,j} \leftarrow f_{\text{convert\_to\_csv}}(F_{i,j})$
  end for
  Merge CSVs of all chunks for PCAP P_i:
    $C_i \leftarrow f_{\text{merge\_csvs}}(C_{i,1}, C_{i,2}, \ldots, C_{i,k})$
  Add attack label l_i to each row in the CSV:
    $C_{i,\text{labeled}} \leftarrow f_{\text{add\_label}}(C_i, l_i)$
end for
Return all labeled CSVs for each attack:
  $D_{\text{labeled}} \leftarrow \{C_{1,\text{labeled}}, C_{2,\text{labeled}}, \ldots, C_{n,\text{labeled}}\}$

simultaneously. This parallelization speeds up the process of extracting features, Table 1, which is crucial when dealing with multiple large PCAP files. For each chunk, flow-level features are extracted, representing key network traffic characteristics that can be used for analysis and classification.

After feature extraction, the features of each chunk are saved as a CSV file. Once all chunks of a particular PCAP have been processed, the individual CSV files are merged back into one comprehensive CSV file. This merged CSV contains all the features from the original attack PCAP but is now split into more manageable parts for further processing. After the merging process, the algorithm adds a label to each row of the CSV, indicating the specific type of attack associated with that PCAP file. This labeling is essential for supervised learning tasks where each data point must be associated with its corresponding attack type.

In the final step, the algorithm outputs the labeled CSVs for each attack and these labeled CSV files are further preprocessed. By utilizing parallel processing, the algorithm ensures efficient handling of multiple large PCAP files, providing a structured and labeled dataset suitable for further analysis, such as training ML models to recognize and classify different variants of cyberattacks.

B. ***Preprocess the Attacks CSVs***

Algorithm 2 outlines a process to merge multiple labeled CSV datasets of cyberattack data into a single comprehensive data frame, ensuring data cleanliness by removing rows with null values. It applied quantile transformation to the numeric features, enhancing their distribution for subsequent analysis. The cleaned and transformed data was then grouped by attack labels, allowing for easy segmentation based on the type of attack. The Quantile Transformer is a preprocessing technique that transforms features to follow a uniform or normal distribution, enhancing model performance by stabilizing variance and reducing skewness.

$$X_{\text{uniform}} = \text{Quantile}(X) \cdot (b - a) + a$$

where a and b are the bounds for the uniform distribution. Finally, each group was saved as a separate CSV file in a specified output directory, facilitating further analysis or training of models on specific attack types. This method ensured an organized and efficient workflow for preprocessing cyberattack data.

C. ***Treatment for Imbalanced Attacks Categories***

During the analysis, Fig. 4, researchers identified the presence of minority attack classes within our dataset, specifically on 452 489 samples for DDoS-ICMP_Fragmentation, 286 925 samples for DDoS-UDP_Fragmentation, 285 104 samples for DDoS-ACK_Fragmentation, 214 952 samples for MQTT-DDoS-Connect_Flood,71 864 samples for DoS-HTTP_Flood, 52 881 samples for MQTT-DoS Publish_Flood,36 039 samples for MQTT-DDoS-Publish_Flood,28 790 samples for DDoS-HTTP_Flood,23 426 samples for DDoS-SlowLoris, and 15 904 samples for MQTT-DoS-Connect_Flood. These minority classes contribute to a significant class imbalance, leading to potential bias in the ViT model. When certain classes are underrepresented, the model may struggle to accurately learn their characteristics, resulting in poor predictive performance for these categories. The imbalance classes not only negatively affect the model's as a whole accuracy but also its ability to generalize well across

Algorithm 2 Merge, Clean, Transform, and Group Labeled CSVs

Function: DDoSViT_ApplyPreprocessing($D_{labeled}$)
Input: Set of labeled CSV datasets for each attack:
  $D_{\text{labeled}} = \{C_1^{\text{labeled}}, C_2^{\text{labeled}}, \ldots, C_n^{\text{labeled}}\}$
Output: Separate CSV files for each attack label:
  $\{C_{\text{label1}}, C_{\text{label2}}, \ldots, C_{\text{label}m}\}$
Steps:
Merge all labeled CSVs into a single DataFrame:
  $df_{\text{merged}} \leftarrow f_{\text{merge\_csvs}}(D_{\text{labeled}})$
Remove rows with null values:
  $df_{\text{clean}} \leftarrow f_{\text{drop\_na}}(df_{\text{merged}})$
Apply quantile transformation on numeric features:
  numeric_features $\leftarrow f_{\text{select\_numeric}}(df_{\text{clean}})$
  $tmp \leftarrow f_{\text{quantile\_transform}}(df_{\text{clean}}[\text{numeric\_features}])$
  $df_{\text{clean}}[\text{numeric\_features}] \leftarrow tmp$
Group by attack label:
  groups $\leftarrow f_{\text{group\_by}}(df_{\text{clean}}, \text{label\_column})$
Store each group into separate CSV files:
for each group G_{label} in groups **do**
    $f_{\text{save\_csv}}(G_{\text{label}}, \text{label} + \text{".csv"})$
end for

all classes. Addressing this imbalance is essential to affirm that the model fairly represents and achieves high performance in all classes, ultimately improving the reliability and effectiveness of the ViT model.

To address the identified class imbalance, particularly among the minority classes, researchers in previous studies have harnessed the Synthetic Minority Over-sampling Technique. By employing SMOTE, they generated synthetic samples for the minority classes, which helped to balance the dataset and reduce the bias that could negatively impact our model's performance.

Synthetic instance Generation formula:

$$\text{new\_sample} = \text{orig\_inst} + \lambda \times (\text{neighbor} - \text{orig\_inst})$$

Here, λ is a randomly generated number between 0 and 1. The selection of the number of neighbors (k) is crucial for the effectiveness of SMOTE. A small k may lead to overfitting, as the synthetic samples closely mimic existing instances, while a large k might introduce noise from majority class instances, diluting the quality of the generated samples. Therefore, the researchers of this study chose the default configuration k = 7. By utilizing SMOTE, they created synthetic attack samples for these undersample classes, effectively increasing their representation within the dataset Fig. 5. This process helped to mitigate the bias that could adversely affect the proposed model's performance, enabling it to learn the characteristics of all classes more effectively. As a result, a balanced dataset that improved the model's predictive accuracy and robustness was achieved, ensuring it could generalize well across all categories, particularly for the MQTT flooding attacks categories.

D. ***Flow-to-Image***

Algorithm 3 outlines the process of converting a set of attack CSV files into RGB images and storing them in corresponding folders. It begins by loading each CSV file into a DataFrame and flattening the numeric features into a single array. The algorithm then calculates the number of images required based on the total number of values, ensuring that each image has the appropriate size of $46 \times 46 \times 3$ pixels. The samples of 23 distinct variants of DoS/DDoS attacks are depicted in Fig. 4. If deemed necessary, padding was added to the data to match the required size. The flattened data was reshaped into images, and each image was saved as an RGB file in a folder named after the attack label. The process was repeated for all provided CSV files, ensuring a structured and organized output (see Fig. 6).

E. ***DDoSViT Architecture***

Algorithms 4 & 5 for training a DDoSViT using an attack image dataset begin with the necessary inputs. This includes folders containing attack images for different classes, represented as $F = \{F_1, F_2, \ldots, F_n\}$. The dataset was split into three parts based on specified ratios: for training 70%, for validating 10%, and for testing 20%. The algorithm also requires the Vision Transformer model, denoted as ViT, and a set of hyperparameters h that define aspects such as the number of layers, heads, embedding dimensions, patch_size, learning_rate, and the number of epochs. The DDoSViT framework is illustrated in Fig. 1. The first step involved loading the image files from the specified folders, where each image was associated with a specific attack class label. Next, the dataset was split into pre-defined rations for training, validating, and testing. After preparing the data, the Vision Transformer was initialized with the specified hyperparameters, ensuring that the model was configured correctly for training. Following initialization, pixel values of, [0, 255], the images were normalized by converting them to the scale of [0, 1], which is a standard pre-processing step in image processing tasks.

Algorithm 3 Flow To Image and Store in Folders

Input: Attack CSVs: $\{C_{label1}, C_{label2}, \ldots, C_{labelm}\}$
Output: Each Attack Folder, containing $46 \times 46 \times 3$ RGB images.
Function: DDoSViT_toImage($D_{labeled}$)
for each attack CSV C_{label} **do**
  $df \leftarrow$ load_csv(C_{label})
  $data_{flattened} \leftarrow$ flatten_data(df)
  $num\_values \leftarrow$ len($data_{flattened}$)
  $num\_images \leftarrow \lceil \frac{num\_values}{6348} \rceil$
  if necessary **then**
    $padded\_data \leftarrow$ pad_data($data_{flattened}, num\_images \times 6348$)
  end if
  create folder
  for $i = 0$ **to** $num\_images - 1$ **do**
    $image_i \leftarrow$ reshape($padded\_data[i \times 6348 : (i+1) \times 6348], (46, 46, 3)$)
    Save image in folder
  end for
end for

Algorithm 4 DDoSViT Model

Function: DDoSViT_TrainTestModel($D_{labeled}$)
Input:
  Folders contain Attack Images: $F = \{F_1, F_2, \ldots, F_n\}$
  Dataset split: train = 70%, valid = 10%, test = 20%
  DDoSViT model: $DDoSViT$
  Hparams: $h =$
  {n_layers, n_heads, embed_dim, patch_size, lr, epochs}
Output:
  DDoSViT model: $DDoSViT_{trained}$
Steps:
Load Image Files from Folders:
  $I \leftarrow fload\_images(F)$
Split the Dataset:
  $(I_{train}, I_{val}, I_{test}) \leftarrow fsplit(I, (70\%, 10\%, 20\%))$
Initialize DDoSViT:
  $DDoSViT \leftarrow finit\_ddosvit(h)$
Normalize Pixel Values:
  $(I'_{train}, I'_{val}, I'_{test}) \leftarrow \left(\frac{I_{train}}{255}, \frac{I_{val}}{255}, \frac{I_{test}}{255} \right)$
Train the Vision Transformer:
  $DDoSViT_{trained} \leftarrow ftrain(ViT, I'_{train}, I'_{val})$
Evaluate the Model:
  $accuracy \leftarrow fevaluate(DDoSViT_{trained}, I'_{test})$
Return the Trained Model:
  return $DDoSViT_{trained}$

For dataset splitting, the dataset of 12 Million samples was converted into 86 952 images. Subsequently, Table 2, 62 606 (70%) images were allocated for training, 8695 (10%) for validation to fine-tune hyperparameters, and 15 651 (20%) for testing to assess the model's performance. The split was carried out using random and shuffle techniques. This technique ensured that the model had a robust training set, an effective validation set for optimizing its performance, and a separate test set to evaluate its generalization capabilities on unseen data.

The training phase then began, where the Vision Transformer model was trained on the normalized training dataset and validated using the validation set. Once training was complete, the model was evaluated on the test set to calculate its accuracy. Finally, the trained Vision Transformer model was returned for further use. This comprehensive approach ensured that the model was effectively trained and validated, prepared for deployment in IoT Edge devices.

F. ***DDoSViT Hyperparameters Optimization***

Hyperparameters play a crucial role in determining a Vision Transformer (ViT) model's architecture, learning behavior, and ultimately its performance, particularly when deploying in IoT-Edge devices. The hyperparameters were tailored for a dataset with input images sized at $46 \times 46 \times 3$ pixels and requiring classification into 23 different variants of DoS and DDoS attacks.

Algorithm 5 DDoSViT Architecture

Require: Image dataset $I = \{I_1, I_2, \ldots, I_n\}$, labels $L = \{l_1, l_2, \ldots, l_n\}$, ViT model ViT, hyperparameters $h = \{n_{\text{layers}}, n_{\text{heads}}, \text{embed\_dim}, \text{patch\_size}, \ldots\}$
Ensure: Trained Vision Transformer model ViT_{trained}
for each image $I_i \in I$ **do**
  Image-to-Patches:
  $P_i = \{p_1, p_2, \ldots, p_k\}$, where $p_j \in \mathbb{R}^{\text{patch\_size} \times \text{patch\_size}}$
  Patch Embedding:
  $E(P_i) = \{e_1, e_2, \ldots, e_k\}$, where $e_j = f_{\text{embed}}(p_j) + \text{pos}_j$
  Transformer Encoder:
  $Z = \text{TransformerEncoder}(E(P_i)) = \{z_1, z_2, \ldots, z_k\}$, where $z_j \in \mathbb{R}^{\text{embed\_dim}}$
  Classification Head:
  $y_i = \text{softmax}(f_{\text{cls}}(z_{[CLS]}))$, where $y_i \in \mathbb{R}^C$ and C is the number of classes
end for
Training: Minimize the loss function:
$L = -\sum_{i=1}^{n} l_i \log(y_i)$
using optimization algorithm (e.g., Adam) with learning rate lr for e epochs:
$ViT_{\text{trained}} = \text{minimize}(L, \text{optimizer} = \text{Adam}, lr)$
Evaluation: Measure accuracy:
$Acc = \frac{1}{n} \sum_{i=1}^{n} 1(y_i = l_i)$ on validation set

Table 1
Dataset features.

Features	Features	Features
Header length	Protocol type	IGMP
Duration	Rate	Weight
Srate	Drate	Variance
fin flag number	syn flag number	Covariance
rst flag number	psh flag number	Radius
ack flag number	ece flag number	Magnitue
cwr flag number	ack count	Number
syn count	fin count	IAT
urg count	rst count	Tot size
HTTP	HTTPS	Std
DNS	Telnet	AVG
SMTP	SSH	Max
IRC	TCP	Min
UDP	DHCP	Tot sum
ARP	ICMP	LLC
IPv		

The suggested hyperparameters were carefully chosen to make a balance between model complexity, training efficiency, and effective feature extraction while considering the resource constraints typical of edge devices.

Fine-tuned hyperparameters are summarized in Table 3 depicting a learning_rate to 0.0005, weight_decay to 0.0001, and a 32 batch size, which are standard practices in deep learning. The epochs value was set to 15, with a patience value of 30, allowing for early stopping if there were no improvements in the validation metrics and 23 patch size. However, further analysis suggested refining these patch parameters to optimize the model's performance and suitability for edge deployment. The MLP head units were reduced from higher values of [2048,1024] to [512,256]. This reduction not only optimized the model's performance and prevented overfitting but also significantly decreased the number of trainable parameters from 2.1 Million to approximately 0.1 Million with a slight performance difference of 0.001. By reducing the units, the model became more lightweight and suitable for IoT-Edge. This streamlined architecture is particularly advantageous for edge devices, where memory and computational resources are limited, allowing the network to learn meaningful representations without introducing excessive complexity. The suggested configuration emphasized the importance of balancing complexity and generalization while maintaining a lightweight model suitable for deployment in constrained environments. By carefully adjusting these hyperparameters, the DDoSViT framework achieved 100% accuracy in top-5 validation accuracy, maintained almost similar performance in testing, and demonstrated the effectiveness of these configurations in dealing with the unique challenges raised by DoS and DDoS attack recognition and classification.

Table 2
Cyberattacks images split.

Fold	Total samples
Train	62 606
Validation	8695
Test	15 651

Table 3
DDoSViT optimized hyperparameters.

Hyperparameter	Optimized values
num_classes	24
input_shape	(46, 46, 3)
learning_rate	0.0005
weight_decay	0.0001
batch_size	32
num_epochs	15
patience	30
image_size	46
patch_size	23
num_patches	4
projection_dim	4
num_heads	6
transformer_units	[8, 4]
transformer_layers	8
mlp_head_units	[512, 256]

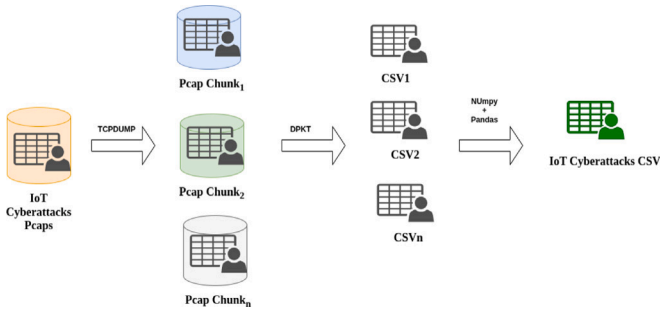


Fig. 3. Converting PCAPs into CSVs.

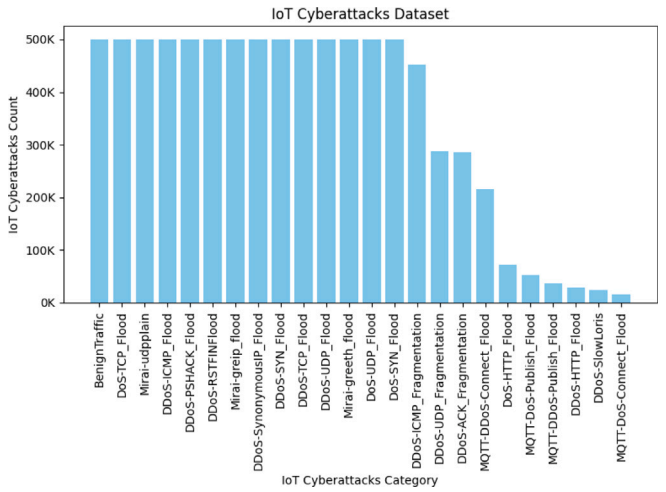


Fig. 4. DoS/DDoS attacks with majority and minority classes.

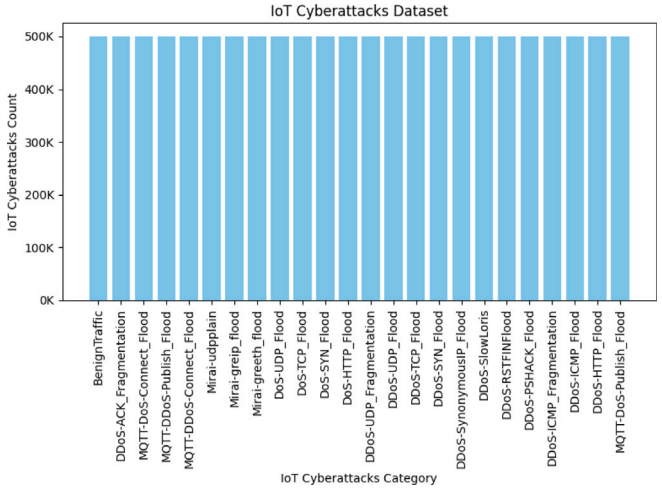


Fig. 5. DoS/DDoS attacks with oversampling minority classes.

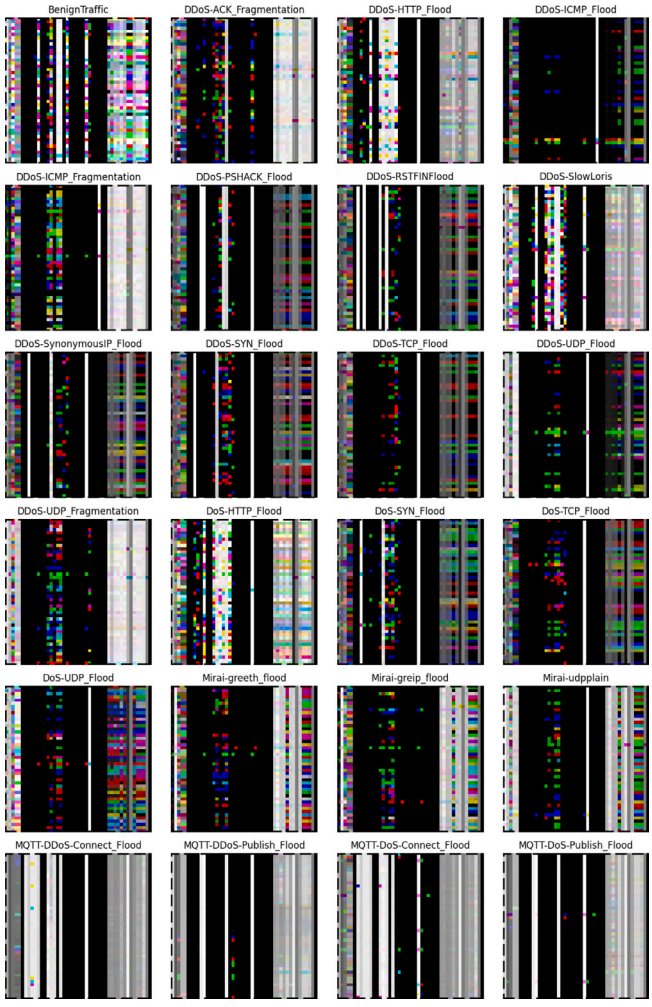


Fig. 6. IoT DDoS attacks images dataset.

Table 4
Performance metrics description.

Metrics	Description
True Positive (TP)	The total of attacks which are accurately classified
False Positive (FP)	The total of attacks which are inaccurately classified
True Negative (TN)	The total of benign which are accurately classified
False Negative (FN)	The total of benign which are inaccurately classified

4. Results and discussion

The implementation of the DDoSViT, a Vision Transformer (ViT) based multi-vector DoS and DDoS attack detection framework for fortifying firmware over-the-air update yielded significant results. The overall multi-class DoS and DDoS attack detection accuracy achieved by the framework is 99.50%, 99.53% precision, recall and f1-score 99.50%. The framework demonstrated its effectiveness and robustness in the identification/detection of a wide range of 23 different variants of DoS/DDoS attack patterns in IoT-Edge devices.

The feature normalization/standardization technique using Quantile Transformer (QT) contributed to improving the multi-class classification significantly and increasing the model accuracy, ensuring that the benign and attack traffic were accurately identified. The representation of flows as images was incorporated, which helped the model to learn spatial relationships between features and improved complex DoS/DDoS attack patterns.

The ViT training, cross-validation and testing on the latest CICIoT2023 and CICIoT2024 real-time IoT DoS/DDoS datasets confirmed the DDoSViT framework's practical effectiveness for real-world IoT-Edge applications. The SMOTE oversampling technique was also incorporated which substantially further improved the model performance.

Confusion matrix is a highly effective tool. It is used to summarize and visualize the efficiency of the deep learning binary classification and multiclass classification algorithms. It has four fundamental components, True Positive, False Positive, True Negative and False Negative, summarized in Table 4. They are crucial for driving different performance metrics.

These are the outlined formulas:

- **Accuracy**:

$$\text{Acc} = \frac{\text{TN} + \text{TP}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}}$$

- **Precision (PPV)** :

$$\text{Prec} = \frac{\text{TP}}{\text{FP} + \text{TP}}$$

- **Recall (TPR)**:

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

- **F1-Score**:

$$\text{F1-Score} = 2 \times \frac{\text{PPV} \times \text{TPR}}{\text{PPV} + \text{TPR}}$$

The critical examination of the proposed DDoSViT framework and comparison with contemporary solutions demonstrates strong effectiveness and robustness in detecting multi-vector DoS and DDoS attacks in Internet of Things Edge devices enhanced with reduced memory and preprocessing requirements, which makes it more compatible with lightweight or constrained IoT applications. It achieved excellent accuracy and recall measures. The cross-validation and optimization techniques helped make the model more generalized. It outperformed the contemporary Restnet18 [52] based solution in terms of multi-factors. DDoSViT selected reduced features 46 and $46 \times 46 \times 3$ input image as compared to Resnet18, 60 features and $224 \times 224 \times 3$ input image, without compromising detection accuracy. They transformed a $60 \times 60 \times 3$ image to a $224 \times 224 \times 3$ image to make the input image compatible with model requirements which introduced more overhead to preprocessing, model training and memory. [52] utilized the Min-Max scaling technique which preserves relationships but is sensitive to outliers. The proposed Quantile transformation(QT) makes DDoSViT robust to outliers and skewness while preserving the relative relationships. Our proposed model excelled 12.53% in terms of precision against Restnet18 and other cutting-edge solutions in multi-class classification in complex features and multi-vector attacks detection, summarized in Table 5. In terms of multi-vector, our proposed model excelled in detecting 23 different variants of IoT DoS and DDoS attacks as compared to contemporary solutions.

5. Conclusion

In this research study, a cutting-edge novel framework, DDoSViT, is proposed to effectively detect 23 distinct variants of DDoS and DoS attacks during firmware over-the-air (OTA) updates for the Internet of Things (IoT) edge devices. The deep learning model incorporated the Vision Transformer (ViT) architecture, which enabled high detection accuracy while minimizing processing time

Table 5
DDoSViT framework comparison summary.

Solution	Acc	Pre	Recall	F1-score
DDoSViT	**0.9950**	**0.9953**	**0.9950**	**0.9950**
RestNet18 [52]	0.8706	0.8700	0.8600	0.8600
LEDEM [55]	0.9628	0.9700	0.9800	0.9700
DeepGFL [59]	0.9300	0.7567	0.3024	0.4321
MLP [60]	0.8634	0.8847	0.8625	0.8735
ID-CNN [60]	0.9514	0.9017	0.9017	0.9399
LSTM [60]	0.9624	0.9814	0.8989	0.8959
ID-CNN-LSTM [60]	0.9716	0.9741	0.9910	0.9825

and computational resource requirements. To ensure compatibility with resource-constrained edge computing environments, the DDoSViT framework was optimized using TensorFlowLite, a technique that reduces the model's complexity without compromising its accuracy. The DDoSViT framework underwent thorough training, validation and testing using the latest real-time datasets, CICIOT2023 and CICIOMT2024, demonstrating its adaptability across various scenarios and attack intensities. In contrast to contemporary cutting-edge models, the DDoSViT framework achieved a 99.50% detection accuracy. Moreover, the framework performance was assessed under full activation conditions to test its robustness in detecting DoS and DDoS attacks during firmware over-the-air updates.

6. Limitations and future work

The limitations of using DDoSViT for detecting stealthy Generated Adversarial Network (GAN) based DoS/DDoS attacks include its potential difficulty in distinguishing between benign traffic and malicious traffic generated by GANs. Additionally, GANs are designed to mimic benign behavior, which can lead to challenges in detecting subtle anomalies, potentially resulting in higher false negatives. Future work can focus on enhancing the DDoSViT framework by incorporating additional CoAP DDoS/DoS attack traffic, training and validating on stealthy GAN-based DoS/DDoS attacks and feature complexity reduction for more lightweight operation. Further, efforts can be directed towards refining the packet processing module to improve the system's efficiency and robustness.

CRediT authorship contribution statement

Muhammad Ali: Writing – original draft, Visualization, Validation, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Yasir Saleem:** Supervision, Resources. **Sadaf Hina:** Writing – review & editing, Visualization, Conceptualization. **Ghalib A. Shah:** Supervision, Resources, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Link to FigShare dataset is attached.

DDoSViT (Original data) (Figshare)

References

[1] M. Mansour, A. Gamal, A.I. Ahmed, L.A. Said, A. Elbaz, N. Herencsar, A. Soltan, Internet of things: A comprehensive overview on protocols, architectures, technologies, simulation tools, and future directions, Energies 16 (2023) 3465, http://dx.doi.org/10.3390/en16083465.
[2] M. Kavre, A. Gadekar, Y. Gadhade, Internet of things (IoT): A survey, in: 2019 IEEE Pune Section International Conference, PuneCon, Pune, India, 2019, pp. 1–6, http://dx.doi.org/10.1109/PuneCon46936.2019.9105831.
[3] R. Chataut, A. Phoummalayvane, R. Akl, Unleashing the power of IoT: A comprehensive review of IoT applications and future prospects in healthcare, agriculture, smart homes, smart cities, and industry 4.0, Sensors 23 (2023) 7194, http://dx.doi.org/10.3390/s23167194.
[4] T. Mazhar, D.B. Talpur, T.A. Shloul, Y.Y. Ghadi, I. Haq, I. Ullah, K. Ouahada, H. Hamam, Analysis of IoT security challenges and its solutions using artificial intelligence, Brain Sci. 13 (2023) 683, http://dx.doi.org/10.3390/brainsci13040683.
[5] B.I. Mukhtar, M.S. Elsayed, A.D. Jurcut, M.A. Azer, IoT vulnerabilities and attacks: SILEX malware case study, Symmetry 15 (2023) 1978, http://dx.doi.org/10.3390/sym15111978.
[6] S.A. Baho, J. Abawajy, Analysis of consumer IoT device vulnerability quantification frameworks, Electronics 12 (2023) 1176, http://dx.doi.org/10.3390/electronics12051176.
[7] M. Hossain, G. Kayas, R. Hasan, A. Skjellum, S. Noor, S.M.R. Islam, A holistic analysis of internet of things (IoT) security: Principles, practices, and new perspectives, Futur. Internet 16 (2024) 40, http://dx.doi.org/10.3390/fi16020040.

[8] J. Kim, M. Shim, S. Hong, Y. Shin, E. Choi, Intelligent detection of IoT botnets using machine learning and deep learning, Appl. Sci. 10 (2020) 7009, http://dx.doi.org/10.3390/app10197009.

[9] T.G. Palla, S. Tayeb, Intelligent Mirai malware detection for IoT nodes, Electronics 10 (2021) 1241, http://dx.doi.org/10.3390/electronics10111241.

[10] Ionut Ilascu, Chinese Botnet Infects 260 000SOHO Routers, IP Cameras with Malware, BleepingComputer, 2024, https://www.bleepingcomputer.com/news/security/flax-typhoon-hackers-infect-260-000-routers-ip-cameras-with-botnet-malware/.

[11] B.L. Labs, Derailing the Raptor Train - Lumen, Lumen - Lumen's Blog, 2024, https://blog.lumen.com/derailing-the-raptor-train/ (accessed 24 October 2024).

[12] X. Feng, X. Zhu, Q.-L. Han, W. Zhou, S. Wen, Y. Xiang, Detecting vulnerability on IoT device firmware: A survey, IEEE/CAA J. Autom. Sin. 10 (1) (2023) 25–41, http://dx.doi.org/10.1109/JAS.2022.105860.

[13] X. Feng, X. Zhu, Q.-L. Han, W. Zhou, S. Wen, Y. Xiang, Detecting vulnerability on IoT device firmware: A survey, IEEE/CAA J. Autom. Sin. 10 (1) (2023) 25–41, http://dx.doi.org/10.1109/JAS.2022.105860.

[14] A. Palmieri, P. Prem, S. Ranise, U. Morelli, T. Ahmad, MQTTSA: A tool for automatically assisting the secure deployments of MQTT brokers, in: 2019 IEEE World Congress on Services, SERVICES, Milan, Italy, 2019, pp. 47–53, http://dx.doi.org/10.1109/SERVICES.2019.00023.

[15] K. Sahlmann, V. Clemens, M. Nowak, B. Schnor, MUP: Simplifying secure over-the-air update with MQTT for constrained IoT devices, Sensors 21 (2021) 10, http://dx.doi.org/10.3390/s21010010.

[16] A.J. Hintaw, S. Manickam, M.F. Aboalmaaly, S. Karuppayah, MQTT vulnerabilities, attack vectors and solutions in the internet of things (IoT), IETE J. Res. 69 (6) (2021) 3368–3397, http://dx.doi.org/10.1080/03772063.2021.1912651.

[17] K. Han, et al., A survey on vision transformer, IEEE Trans. Pattern Anal. Mach. Intell. 45 (1) (2023) 87–110, http://dx.doi.org/10.1109/TPAMI.2022.3152247.

[18] [1]A. Dosovitskiy, et al., An image is worth 16x16 words: Transformers for image recognition at scale, 2020, arXiv:2010.11929 [cs].

[19] A. Berroukham, K. Housni, M. Lahraichi, Vision transformers: A review of architecture, applications, and future directions, in: 2023 7th IEEE Congress on Information Science and Technology, CiSt, Agadir - Essaouira, Morocco, 2023, pp. 205–210, http://dx.doi.org/10.1109/CiSt56084.2023.10410015.

[20] A.R. Shaaban, E. Abd-Elwanis, M. Hussein, DDoS attack detection and classification via Convolutional Neural Network (CNN), in: 2019 Ninth International Conference on Intelligent Computing and Information Systems, ICICIS, Cairo, Egypt, 2019, pp. 233–238, http://dx.doi.org/10.1109/ICICIS46948.2019.9014826.

[21] M. Sinthuja, K. Suthendran, DDoS attack detection using enhanced long-short term memory with hybrid machine learning algorithms, in: 2022 3rd International Conference on Smart Electronics and Communication, ICOSEC, Trichy, India, 2022, pp. 1213–1218, http://dx.doi.org/10.1109/ICOSEC54921.2022.9951976.

[22] S. Dong, M. Sarem, DDoS attack detection method based on improved KNN with the degree of DDoS attack in software-defined networks, IEEE Access 8 (2020) 5039–5048, http://dx.doi.org/10.1109/ACCESS.2019.2963077.

[23] T. Alasmari, A. Eshmawi, A. Alshomrani, L. Hsairi, CNN-LSTM based approach for DDoS detection, in: 2023 Eighth International Conference on Mobile and Secure Services, MobiSecServ, Miami Beach, FL, USA, 2023, pp. 1–6, http://dx.doi.org/10.1109/MobiSecServ58080.2023.10329028.

[24] M. Salehi, A. Yari, Detecting DOS attacks using a hybrid CNN-LSTM model, in: 2024 10th International Conference on Web Research, ICWR, Tehran, Iran, Islamic Republic of, 2024, pp. 397–401, http://dx.doi.org/10.1109/ICWR61162.2024.10533358.

[25] A.D. Smith, S. Du, A. Kurien, Vision transformers for anomaly detection and localisation in leather surface defect classification based on low-resolution images and a small dataset, Appl. Sci. 13 (2023) 8716, http://dx.doi.org/10.3390/app13158716.

[26] Mohammad Rahman, Anushua Ahmed, Fahmid Kibria, Mohammad Mahin, Mutasim Khan, Dewan Karim, Mohammad Kaykobad, CNN vs transformer variants: Malware classification using binary malware images, 2024, http://dx.doi.org/10.1109/COMNETSAT59769.2023.10420585.

[27] M.M. Belal, D.M. Sundaram, Global-local attention-based butterfly vision transformer for visualization-based malware classification, IEEE Access 11 (2023) 69337–69355, http://dx.doi.org/10.1109/ACCESS.2023.3293530.

[28] 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 (2023) 5941, http://dx.doi.org/10.3390/s23135941.

[29] Sajjad Dadkhah, Euclides Carlos Pinto Neto, Raphael Ferreira, Reginald Chukwuka Molokwu, Somayeh Sadeghi, Ali A. Ghorbani, CICIoMT2024: A benchmark dataset for multi-protocol security assessment in IoMT, Internet Things 28 (2024) 101351, http://dx.doi.org/10.1016/j.iot.2024.101351.

[30] A.F. Almutairi, A. Abdulghani Alshargabi, Using deep learning technique to protect internet network from intrusion in IoT environment, in: 2022 2nd International Conference on Emerging Smart Technologies and Applications, eSmarTA, Ibb, Yemen, 2022, pp. 1–6, http://dx.doi.org/10.1109/eSmarTA56775.2022.9935467.

[31] S.H.S. Ariffin, N.H. Mustaffa, F. Dewanta, I.W. Hamzah, M.A. Baharudin, N.H. Abdul Wahab, Hybrid feature selection based lightweight network intrusion detection system for MQTT protocol, in: 2023 15th International Conference on Software, Knowledge, Information Management and Applications, SKIMA, Kuala Lumpur, Malaysia, 2023, pp. 226–230, http://dx.doi.org/10.1109/SKIMA59232.2023.10387337.

[32] M.M. Alani, IoTProtect: A machine-learning based IoT intrusion detection system, in: 2022 6th International Conference on Cryptography, Security and Privacy, CSP, Tianjin, China, 2022, pp. 61–65, http://dx.doi.org/10.1109/CSP55486.2022.00020.

[33] M. Ramaiah, M.Y. Rahamathulla, Securing the industrial IoT: A novel network intrusion detection models, in: 2024 3rd International Conference on Artificial Intelligence for Internet of Things, AIIoT, Vellore, India, 2024, pp. 1–6, http://dx.doi.org/10.1109/AIIoT58432.2024.10574728.

[34] S. Hanif, T. Ilyas, M. Zeeshan, Intrusion detection in IoT using artificial neural networks on UNSW-15 dataset, in: 2019 IEEE 16th International Conference on Smart Cities: Improving Quality of Life using ICT & IoT and AI, HONET-ICT, Charlotte, NC, USA, 2019, pp. 152–156, http://dx.doi.org/10.1109/HONET.2019.8908122.

[35] P.V. Huong, L.D. Thuan, L.T. Hong Van, D.V. Hung, Intrusion detection in IoT systems based on deep learning using convolutional neural network, in: 2019 6th NAFOSTED Conference on Information and Computer Science, NICS, Hanoi, Vietnam, 2019, pp. 448–453, http://dx.doi.org/10.1109/NICS48868.2019.9023871.

[36] P.O. Adebayo, M. Jubrin Abdulahi, O.M. Lawrence, Y.A. Ibrahim, S.A. Faki, B.A. Hassan, An artificial intelligence-based ensemble technique for intrusion detection and prevention in IoT systems, in: 2024 International Conference on Science, Engineering and Business for Driving Sustainable Development Goals, SEB4SDG, Omu-Aran, Nigeria, 2024, pp. 1–6, http://dx.doi.org/10.1109/SEB4SDG60871.2024.10629681.

[37] Z.H. Abdaljabar, O.N. Ucan, K.M. Ali Alheeti, An intrusion detection system for IoT using KNN and decision-tree based classification, in: 2021 International Conference of Modern Trends in Information and Communication Technology Industry, MTICTI, Sana'a, Yemen, 2021, pp. 1–5, http://dx.doi.org/10.1109/MTICTI53925.2021.9664772.

[38] S.K. Kodali, C.H. Muntean, An investigation into deep learning based network intrusion detection system for IoT systems, in: 2021 IEEE International Conference on Data Science and Computer Application, ICDSCA, Dalian, China, 2021, pp. 374–377, http://dx.doi.org/10.1109/ICDSCA53499.2021.9650111.

[39] G. Guo, X. Pan, H. Liu, F. Li, L. Pei, K. Hu, An IoT intrusion detection system based on TON IoT network dataset, in: 2023 IEEE 13th Annual Computing and Communication Workshop and Conference, CCWC, Las Vegas, NV, USA, 2023, pp. 0333–0338, http://dx.doi.org/10.1109/CCWC57344.2023.10099144.

[40] Z. Liu, N. Thapa, A. Shaver, K. Roy, X. Yuan, S. Khorsandroo, Anomaly detection on IoT network intrusion using machine learning, in: 2020 International Conference on Artificial Intelligence, Big Data, Computing and Data Communication Systems, icABCD, Durban, South Africa, 2020, pp. 1–5, http://dx.doi.org/10.1109/icABCD49160.2020.9183842.

[41] R. Latha, R.M. Bommi, Hybrid CatBoost regression model based intrusion detection system in IoT-enabled networks, in: 2023 9th International Conference on Electrical Energy Systems, ICEES, Chennai, India, 2023, pp. 264–269, http://dx.doi.org/10.1109/ICEES57979.2023.10110148.
[42] M. Ramaiah, A. Padma, R. Vishnukumar, M.Y. Rahamathulla, V. Chithanuru, A hybrid wrapper technique enabled Network Intrusion Detection System for Software defined networking based IoT networks, in: 2024 3rd International Conference on Artificial Intelligence for Internet of Things, AIIoT, Vellore, India, 2024, pp. 1–6, http://dx.doi.org/10.1109/AIIoT58432.2024.10574755.
[43] M. Alduailij, Q.W. Khan, M. Tahir, M. Sardaraz, M. Alduailij, F. Malik, Machine-learning-based DDoS attack detection using mutual information and random forest feature importance method, Symmetry 14 (2022) 1095, http://dx.doi.org/10.3390/sym14061095.
[44] R. Ma, Q. Wang, X. Bu, X. Chen, Real-time detection of DDoS attacks based on random forest in SDN, Appl. Sci. 13 (2023) 7872, http://dx.doi.org/10.3390/app13137872.
[45] S. Ahmed, Z.A. Khan, S.M. Mohsin, S. Latif, S. Aslam, H. Mujlid, M. Adil, Z. Najam, Effective and efficient DDoS attack detection using deep learning algorithm, multi-layer perceptron, Futur. Internet 15 (2023) 76, http://dx.doi.org/10.3390/fi15020076.
[46] M.J. Awan, U. Farooq, H.M.A. Babar, A. Yasin, H. Nobanee, M. Hussain, O. Hakeem, A.M. Zain, Real-time DDoS attack detection system using big data approach, Sustainability 13 (2021) 10743, http://dx.doi.org/10.3390/su131910743.
[47] J. Li, L. Lyu, X. Liu, X. Zhang, X. Lyu, FLEAM: A federated learning empowered architecture to mitigate DDoS in industrial IoT, IEEE Trans. Ind. Inform. 18 (6) (2022) 4059–4068, http://dx.doi.org/10.1109/TII.2021.3088938.
[48] R. Vishwakarma, A.K. Jain, A honeypot with machine learning based detection framework for defending IoT based botnet DDoS attacks, in: 2019 3rd International Conference on Trends in Electronics and Informatics, ICOEI, Tirunelveli, India, 2019, pp. 1019–1024, http://dx.doi.org/10.1109/ICOEI.2019.8862720.
[49] M. Zeeshan, et al., Protocol-based deep intrusion detection for DoS and DDoS attacks using UNSW-NB15 and bot-IoT data-sets, IEEE Access 10 (2022) 2269–2283, http://dx.doi.org/10.1109/ACCESS.2021.3137201.
[50] M. Roopak, G.Y. Tian, J. Chambers, An intrusion detection system against DDoS attacks in IoT networks, in: 2020 10th Annual Computing and Communication Workshop and Conference, CCWC, Las Vegas, NV, USA, 2020, pp. 0562–0567, http://dx.doi.org/10.1109/CCWC47524.2020.9031206.
[51] Q. He, et al., A game-theoretical approach for mitigating edge DDoS attack, IEEE Trans. Dependable Secur. Comput. 19 (4) (2022) 2333–2348, http://dx.doi.org/10.1109/TDSC.2021.3055559.
[52] F. Hussain, S.G. Abbas, M. Husnain, U.U. Fayyaz, F. Shahzad, G.A. Shah, IoT DoS and DDoS attack detection using ResNet, in: 2020 IEEE 23rd International Multitopic Conference, INMIC, Bahawalpur, Pakistan, 2020, pp. 1–6, http://dx.doi.org/10.1109/INMIC50486.2020.9318216.
[53] N.M. Yungaicela-Naula, C. Vargas-Rosales, J.A. Perez-Diaz, SDN-based architecture for transport and application layer DDoS attack detection by using machine and deep learning, IEEE Access 9 (2021) 108495–108512, http://dx.doi.org/10.1109/ACCESS.2021.3101650.
[54] J. Bhayo, R. Jafaq, A. Ahmed, S. Hameed, S.A. Shah, A time-efficient approach toward DDoS attack detection in IoT network using SDN, IEEE Internet Things J. 9 (5) (2022) 3612–3630, http://dx.doi.org/10.1109/JIOT.2021.3098029, 1 March1.
[55] N. Ravi, S.M. Shalinie, Learning-driven detection and mitigation of DDoS attack in IoT via SDN-cloud architecture, IEEE Internet Things J. 7 (4) (2020) 3559–3570, http://dx.doi.org/10.1109/JIOT.2020.2973176.
[56] Y. Jia, F. Zhong, A. Alrawais, B. Gong, X. Cheng, FlowGuard: An intelligent edge defense mechanism against IoT DDoS attacks, IEEE Internet Things J. 7 (10) (2020) 9552–9562, http://dx.doi.org/10.1109/JIOT.2020.2993782.
[57] A.O. Sangodoyin, M.O. Akinsolu, P. Pillai, V. Grout, Detection and classification of DDoS flooding attacks on software-defined networks: A case study for the application of machine learning, IEEE Access 9 (2021) 122495–122508, http://dx.doi.org/10.1109/ACCESS.2021.3109490.
[58] C.D. McDermott, F. Majdani, A.V. Petrovski, Botnet detection in the internet of things using deep learning approaches, in: 2018 International Joint Conference on Neural Networks, IJCNN, Rio de Janeiro, Brazil, 2018, pp. 1–8, http://dx.doi.org/10.1109/IJCNN.2018.8489489.
[59] Yepeng Yao, Liya Su, Zhigang Lu, DeepGFL: Deep feature learning via graph for attack detection on flow-based network traffic, 2018, pp. 579–584, http://dx.doi.org/10.1109/MILCOM.2018.8599821.
[60] M. Roopak, G. Yun Tian, J. Chambers, Deep learning models for cyber security in IoT networks, in: 2019 IEEE 9th Annual Computing and Communication Workshop and Conference, CCWC, Las Vegas, NV, USA, 2019, pp. 0452–0457, http://dx.doi.org/10.1109/CCWC.2019.8666588.