Received: 6 March 2026 | Revised: 3 June 2026 | Accepted: 23 June 2026 | Published online: 27 July 2026

RESEARCH ARTICLE

Journal of Data Science and Intelligent Systems
2026, Vol. 00(00) 1–21
DOI: 10.47852/bonviewJDSIS62029543

BON VIEW PUBLISHING

A Deep Reinforcement Learning–Based Adaptive Framework for Early DDoS Attack Detection and Prevention in Heterogeneous Networks

Sujit Sutradhar[1,*], Joy Lal Sarkar[2] and Abhijit Biswas[3]

[1] *IT Department, ICFAI University, India*

[2] *Department of Computer Science and Engineering, ICFAI University, India*

[3] *Faculty of Science & Technology, ICFAI University, India*

Abstract: Distributed Denial-of-Service (DDoS) attacks are constantly growing in scale, intensity, and attack plan strategies, which are causing significant threats to the heterogeneous network environments, including Internet of Things, edge, cloud, Software-Defined Networks, and 5G networks. This paper presents an adaptive framework of deep reinforcement learning (DRL) in early-stage detection and proactive mitigation of DDoS attacks. The proposed DRL agent operates in active network environments to develop policies through online learning. This helps it address new and unexpected attack patterns. Traditional machine learning and deep learning models depend on unchanging training methods, which run their training process from a fixed point in time. The framework combines the flow-level feature analysis and the packet-level feature analysis with the Deep Q-Network–based learning mechanism to facilitate real-time decision-making. Numerous experiments were carried out on benchmark datasets and heterogeneous traffic simulations. This indicates that the suggested methodology can be considered stable and offers an approximation of 98% accuracy, 96% precision, 96% recall, and 96% macro F1-score on the datasets with up to 300,000 network flows. Further flow analysis shows that flow duration, packet-level variability, and distribution of destination ports are critical in differentiating between benign and malicious traffic. The findings prove that the developed framework can offer detection and mitigation of DDoS threats, which are scalable, robust, and adaptive, to facilitate the creation of intelligent and self-educating cybersecurity systems in complex network settings.

Keywords: deep reinforcement learning, DDoS detection, network security, adaptive intrusion detection, traffic classification

1. Introduction

1.1. Distributed Denial-of-Service (DDoS) attacks background

Distributed Denial-of-Service (DDoS) attacks are a significant cybersecurity problem today [1]. DDoS attacks send a large amount of traffic to the server, service, or network of a victim over a few infected sources, typically a botnet. Malicious requests consume network bandwidth, server resources, and application processing power, thereby degrading overall system performance. DDoS attacks have become more intense, massive, and sophisticated over the past ten years. As per the recent cyber data, there was a high increase in high-priority DDoS attacks against corporations worldwide [2]. The attacks currently are more than 2.5 terabits per second (Tbps). Dark web tools and botnets are available to more users, and new DDoS forms are being developed. They simplify sophisticated assaults and make them cost-effective to an amateur. Improperly set up Internet of Things (IoT) gadgets, routers, and endpoints make botnets of thousands or millions of infected PCs. Mirai, Mozi, and Bashlite botnets have been used by many nontechnical individuals to initiate DDoS attacks. DDoS attacks have many different effects on different Open Systems Interconnection layers [3].

Volumetric attacks are a common type of cyberattack that aims at slowing down the bandwidth of the target network using techniques such as UDP floods, ICMP floods, or Domain Name System (DNS) amplification. The next mass of attacks is protocol assaults. They are directed to services that are publicly available and exploit vulnerabilities in either end of layer 3 (routing) or layer 4 (Transmission Control Protocol) protocols. An example of such attacks is SYN floods, which consume the resources and connection table of a server, and packet fragmentation attacks, which consume memory. The number of attacks occurring at the application layer is increasing. Application-layer attacks are

***Corresponding author:** Sujit Sutradhar, IT Department, ICFAI University, India. Email: sujitsutradhar@iutripura.edu.in

targeted at services on layer 7 (including HTTP and HTTPS) and are designed to drain or overload some applications or Application Programming Interfaces. A common form of DDoS attack is an application-layer attack. The attacks typically need fewer resources than volumetric attacks. The smaller area covered by these methods makes them more dangerous for active systems. DDoS attacks are better handled by firewalls and intrusion detection systems (IDS) and rate-limiting methods than they were in the past [4]. The system uses specific rules and traffic patterns and established limits to identify malicious network traffic.

The attackers of DDoS attacks have also mastered the techniques of masquerading as normal traffic. This has rendered it practically impossible to distinguish between normal and bad users based on rule-based techniques. Detection is even more challenging when low-and-slow DDoS attacks hold the bandwidth below the limits of detection. The greater the number of networks unveiled, the more likely to have a DDoS attack. The European NIS Directive, the standards of NIST (US), and others ensure that organizations are more resistant to such attacks [5, 6]. Security analysts believe that compliance is not protective and that the threat environment is constantly evolving. This research must be provided with solutions that are smart, highly adaptable, and self-directed. DDoS attacks have taken on new forms. Attackers are now smarter, and networks are now more complex. Consequently, traditional defense mechanisms are no longer sufficient. Therefore, state-of-the-art solutions are required to detect, predict, and respond to emerging DDoS attack strategies in real time.

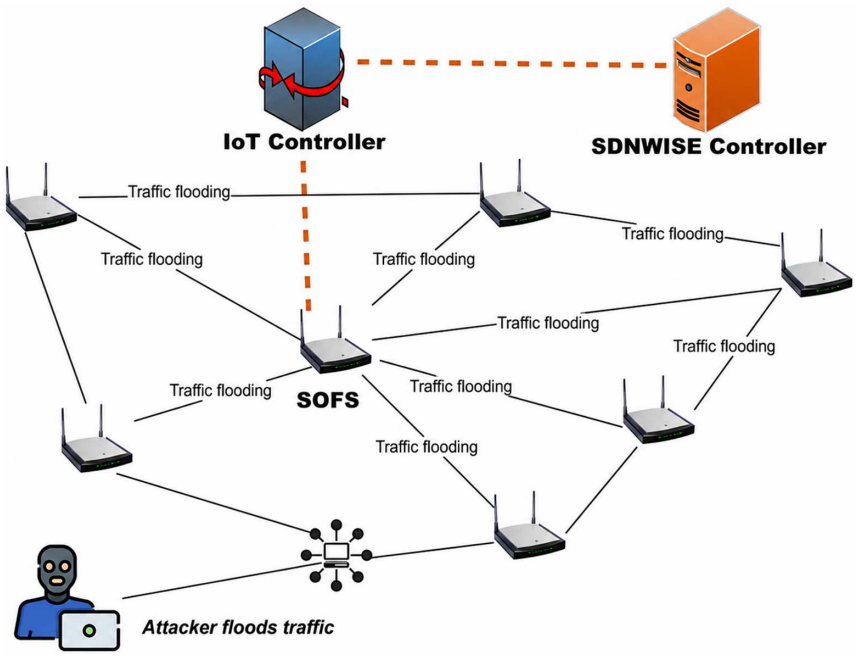
The diagram in Figure 1 [7] shows an integrated DDoS attack. The attacker uses network saturation to create an IoT node overload. This enables him to send false information through the Software-Defined OpenFlow Switch. The traffic flood attacks cause all IoT devices to stop working because these attacks break their connections with the IoT Controller and the SDNWISE Controller.

The network becomes so busy that controllers lose their ability to track ongoing activities. This results in reduced service quality and potential system failures. This research work introduces a new approach that enables DDoS detection across various network environments through the development of stable learning mechanisms and practical deployment capabilities.

1.2. Heterogeneous network complexity

In 5G design, Multi-access Edge Computing (MEC) transfers processing even further to the user, putting computing much nearer to the user. Localized monitoring can fail to detect edge interactions. The attacks should be detected locally and across the network [7]. Network slicing eases service delivery. However, making visibility and security requirements less visible and thus easy to exploit slice differences by attackers. Various networks depend on cloud services. With the increasing popularity of hybrid and multi-cloud, workloads in AWS, Azure, and Google Cloud are distributed. All cloud providers possess their own tools, settings, and logging options. Therefore, you can separate security monitoring. A compromise may cross the cloud settings or from on premises to the cloud platforms, thus becoming hard to locate [8]. Cloud infrastructure is not fixed; a flood of traffic can be mistakenly interpreted as scaling, concealing DDoS attacks. Along with technology, organizations and administration are complex. Multi-vector DDoS attacks are becoming more popular with attackers as they can be used to attack network levels [9]. With these problems, it is hard to identify the existence of a DDoS through static and isolated detection. There is a need for intelligent and self-learning algorithms to monitor various networks [10]. Deep reinforcement learning (DRL) would be useful. Since DRL is learnable, detection models can evolve with the changing network behaviors, traffic patterns, and attacks. It sets in place dynamically

Figure 1
DDoS attack via traffic flooding in an IoT-SDN environment



responsive defense procedures to the scope and complexity of the existing infrastructure.

1.3. Shortcomings of conventional detection and prevention mechanisms

DDoS attacks have proven the weaknesses of the previous detection and mitigation systems due to the intensity and complexity of the attacks [11]. Most of the available solutions, including signature-based IDS, fixed firewalls, threshold-based anomaly detection, and rate limiting, operate on predefined rules or self-written signatures. These strategies are ineffective in cloud infrastructure, IoT ecosystems, and 5G networks because they change frequently. Signatures do not prevent distributed, fraudulent, or rapidly evolving attacks. Bogus traffic or IP origin changes can be generated by botnets and fast-flux DNS [12]. Software upgrades or flash mobs increase the traffic, which is wrongly identified as a DDoS pattern, leading to a large number of false positives when there are threshold-based methods. Monolithic and centralized security policies cannot work on elastic networks. This has varying domains as service endpoints, traffic volume, and bandwidth availability vary. The updates are required manually to address new attack vectors [13]. The conventional methods of detection are erratic and unresponsive. They are incapable of dynamic adaptation to novel threats, especially in systems with multiple devices.

1.4. Deep reinforcement learning (DRL)

DRL could adjust detection rules and mitigation strategies in a real-time and network-specific situation without a preset signature or human intervention. It is an active, network-scale, proactive DDoS protection. Because they operate on labeled data, pre-coded signatures, and fixed rules of decision-making, traditional machine learning (ML)-based and rule-based security systems have been unable to respond to more dynamic and multi-vector DDoS attacks [14]. These approaches are best in the case of stable traffic patterns and require retraining. They have a hard time when network behavior is taken over. In practice, the use of static systems is not applicable to real-time situations that cannot cope with the actual changes and new methods of attack that alter network traffic. Conventional ML models are not dynamic in their operation because they are founded on historical data. Due to this, they usually generate numerous false positives and do not work with new attack behaviors and require manual retraining of their rules [15]. They are not applicable in heterogeneous and real-time scenarios where cloud, IoT, edge, and 5G infrastructure traffic properties are not constant because they cannot be modified. The DRL agents continuously search their surroundings, extract lessons from their actions, and use feedback to improve their rules. This simplifies the detection of attacks of the zero day and those that are evolving [16].

DDoS mitigation can be regarded as a choice of options an agent makes based on traffic patterns. The agent then blocks, permits, or rates traffic off. This algorithm operates in accordance with the Markov decision process (MDP) framework of reinforcement learning (RL) [17]. DRL applies deep neural networks to learn high-dimensional and complex attributes of flow statistics, packet data, and behavioral characteristics. It is highly scalable, universal, and responsive in real time. DRL also reduces false positives and performance decay as compared to fixed rule-based systems through the long-term stability of the network. Since DRL can easily learn and adjust to traffic behaviors that are typical of diverse networks, which include cloud platforms, IoT devices, edge nodes, and 5G environments [18]. Even though the static detection methods are not applicable everywhere, the DRL can set up context-sensitive regulations on portions of the network. Some of the RL models are Deep Q-Network (DQN), Proximal Policy Optimization (PPO), and Actor–Critic, which can detect stealthy and polymorphic DDoS attacks in a better way than their counterparts without affecting quality of service (QoS) by modifying mitigation strategies and prioritizing resource distribution. The novelty of the proposed work does not solely arise from the use of a DQN, since DQN-based cybersecurity frameworks have already been reported in the literature. Instead, the contribution lies in the integration of multilayer traffic analysis and adaptive RL within a unified heterogeneous network security framework. The proposed system simultaneously exploits packet-level behavioral characteristics and flow-level temporal dynamics to construct a richer network state representation for policy learning. In contrast to existing DRL-based approaches that are typically validated in a single network domain, the proposed framework is designed to operate across IoT, edge, cloud, Software-Defined Network (SDN), and 5G environments. Moreover, stability-aware learning mechanisms, including controlled policy updates, experience replay, and validation-driven adaptation, are incorporated to improve robustness under non-stationary traffic conditions. These design elements collectively establish the methodological novelty of the proposed framework.

The rest of this paper will be structured in the following way. Section 2 will provide a review of the related work and the recent state-of-the-art studies concerning DDoS detection and mitigation. Section 3 provides an overview of the proposed DRL-based system structure and learning model. The research methodology, dataset description, and feature engineering process. Section 4 presents the experimental results, performance comparison with baseline models, statistical validation, and discussion of the findings. Section 5 concludes the paper and outlines future research directions.

The main contributions of this work are summarized as follows:

1) A DQN-based DRL system that enables DDoS detection across multiple network environments that include IoT, edge, cloud, and SDN and 5G systems.
2) An early-stage DDoS detection mechanism that enables organizations to stop attacks before they develop into major system outages.
3) A stability-aware learning method that uses controlled policy updates, experience replay, and validation mechanisms to maintain dependable performance during changing conditions of non-stationary traffic.
4) The system achieves cross-domain adaptive detection because it can learn to function in new network environments without needing to go through complete offline retraining processes.
5) The researchers conduct a complete assessment of dynamic traffic conditions, which includes testing real-world scenarios that extend beyond laboratory control tests.
6) A unified multilayer traffic representation mechanism that jointly models packet-level variability and flow-level temporal behavior to improve adaptive policy learning for DDoS detection in heterogeneous networks.

2. Related Work

The recent DDoS attacks are now more advanced and have a great impact on large-scale network infrastructures. This has a negative influence on the service reliability and availability [19]. Conventional signature-based and rule-based detection mechanisms are useful in detecting known attack patterns, but not zero-day and changing threats. To address these drawbacks, a few scholars have implemented ML and deep learning (DL) methods to detect anomalies and classify traffic [20, 21]. Most ML and DL methods depend on their fundamental design, which requires static model training. The researchers Bakro et al. [22] used both the Random Forest algorithm and the support vector machine (SVM) classifier to identify DDoS attacks against cloud services, and they achieved impressive results. The method cannot adapt to new attack techniques that emerge in fast-changing network environments. The researchers had low capacity to adapt to changing and hidden attack methods [23]. The hybrid CNN-LSTM model of IoT networks offered by Manokaran and Vairavel [24] achieved superior classification accuracy while delivering substantial price costs through elevated model complexity and extensive training database requirements. The model requires excessive computational resources because it cannot adapt to new environmental conditions. Nomikos et al. [25] proposed a DQN adaptive mitigation mechanism for SDNs. The system only operates within single-domain SDN environments because it lacks the ability to function across different network types. Hady et al. [26] proposed a multi-agent DRL-based system for preventing DDoS attacks in 5G networks using PPO and A3C algorithms. Although the method successfully improved mitigation efficiency, it required excessive operational resources and exhibited slow response under heavy traffic conditions. The system operates at high resource demand, while it takes an extended time to respond when handling large traffic loads. Jiao et al. [27] applied Actor–Critic and GRU models to develop an early-warning system for heterogeneous network systems. The framework does not support cross-layer integration, and its capabilities for generalizing to new situations remain limited. The suggested framework failed to support federated learning and cross-layer contextual integration, which prevented it from achieving generalization capabilities. They proposed a Deep Q-Learning IDS for adaptive cybersecurity while Hossain [21] used deep feature extraction with RL to detect IoT intrusion. The techniques available limit their application to single-domain systems because they fail to handle cross-domain variability. Most research studies demonstrate that DRL functions effectively for adaptive security, but they restrict their experiments to one domain while ignoring cross-domain differences. Security solutions for DDoS detection and mitigation have made significant progress, but their current implementations still contain multiple unresolved issues. The majority of models are statically trained and tested and, therefore, cannot be adjusted to changing traffic patterns [28].

Most systems only function in single network environments because they lack multi-domain capabilities. The implementation of real-time systems demonstrates challenges because both computational expenses and labeled data requirements exceed acceptable limits. The absence of cross-layer visibility, together with missing built-in solutions, results in reduced practical effectiveness. Table 1 presents a summary of current research from 2022 to 2025 about DDoS detection and mitigation research, which details their methods, implementation specifications, and main limitations. The analysis in Table 1 shows that existing methods do not provide complete cross-domain adaptability and stability-based learning, which makes the proposed framework necessary.

Current ML, DL, and DRL methods for DDoS attack detection still have multiple unresolved problems. Existing methods demonstrate limitations because they only function in one specific domain and do not support cross-domain operation that network environments require. The majority of existing methods depend on static training processes or need ongoing retraining, which makes them unsuitable to handle traffic that changes throughout time. Stability-aware learning and real-time adaptability remain two areas that researchers have not yet achieved together.

The present study introduces a DQN-based DRL framework, which enables cross-domain adaptability, early-stage detection, and stability-aware learning capabilities for heterogeneous network environments.

Table 2 compares the proposed framework with existing DRL-based DDoS detection approaches. The results show that

Table 1
Summary of recent related works on DDoS detection and mitigation (2022–2025)

Author(s)	Approach	Technique	Environment	Main limitations
Doğan et al. [23]	ML-based detection	Random Forest, SVM	Cloud networks	Limited adaptability to unseen attacks
Manokaran and Vairavel [24]	DL-based classification	CNN + LSTM	IoT networks	High training cost; limited real-time adaptability
Manokaran and Vairavel [24]	DRL-based mitigation	DQN	SDN	Poor scalability in heterogeneous settings
Hady et al. [26]	Multi-agent DRL	PPO, A3C	5G networks	High resource overhead; slow response
Jiao et al. [27]	Early detection	Actor–Critic + GRU	Heterogeneous networks	Limited cross-layer integration
Jiao et al. [27]	RL-based IDS	Deep Q-Learning	Enterprise networks	Single-domain focus
Sangoleye et al. [29]	DRL-based IDS	Deep features + RL	IoT networks	Limited generalization

Table 2
Comparative analysis of existing DRL-based DDoS detection frameworks and the proposed adaptive framework

Feature	Existing DRL studies	Proposed framework
DQN-based learning	✓	✓
Flow-level analysis	Partial	✓
Packet-level analysis	Partial	✓
Multilayer traffic representation	Limited	✓
Early-stage DDoS detection	Limited	✓
Stability-aware learning	Rarely addressed	✓
Cross-domain deployment (IoT + edge + cloud + SDN + 5G)	✗	✓
Adaptive online policy learning	Partial	✓
Validation-driven policy updates	✗	✓
Heterogeneous traffic evaluation	Limited	✓

the proposed model provides broader traffic analysis, adaptive learning, and support for heterogeneous environments. It also addresses key gaps in prior studies, particularly in early-stage detection, stability-aware learning, and validation-driven policy updates.

Research gap and novelty positioning

Despite the growing adoption of DRL for DDoS detection and mitigation, several limitations remain in existing studies. Most reported frameworks focus on a single deployment environment such as SDN, IoT, cloud, or 5G networks. Although DQN, PPO, Actor–Critic, and other DRL variants have demonstrated promising performance, their evaluation is typically restricted to isolated network domains and controlled experimental settings.

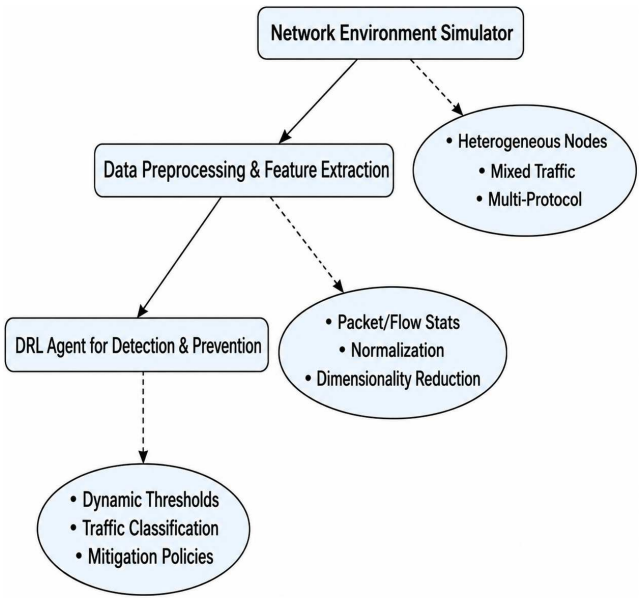
Furthermore, existing DRL-based approaches primarily emphasize attack classification accuracy or mitigation effectiveness, while limited attention has been given to cross-domain adaptability, early-stage attack identification, and learning stability under continuously evolving traffic conditions. Most studies rely on static traffic representations and do not explicitly integrate packet-level and flow-level behavioral characteristics into a unified adaptive decision-making framework.

The proposed framework addresses these limitations by introducing a DQN-based adaptive architecture that combines packet-level variability, flow-level temporal behavior, and RL-driven policy optimization within a single heterogeneous network security framework. Unlike prior approaches that focus on a specific network domain, the proposed system is designed to operate across IoT, edge, cloud, SDN, and 5G environments. In addition, stability-aware policy updating, experience replay, and validation-driven adaptation mechanisms are incorporated to improve robustness under non-stationary traffic conditions. These characteristics collectively distinguish the proposed framework from existing DRL-based DDoS detection systems.

3. Research Methodology

In this section, the workflow of the experimental part of the proposed adaptive framework employing the DQN algorithm to detect and mitigate early DDoS is described. The methodology will involve the preparation of datasets, simulation of traffic, feature engineering, formulation of RL, training of models, statistical validation, and evaluation of performance. The proposed framework is described in Figures 2 and 3.

Figure 2
Architecture of the proposed DQN-based adaptive DDoS detection and mitigation framework



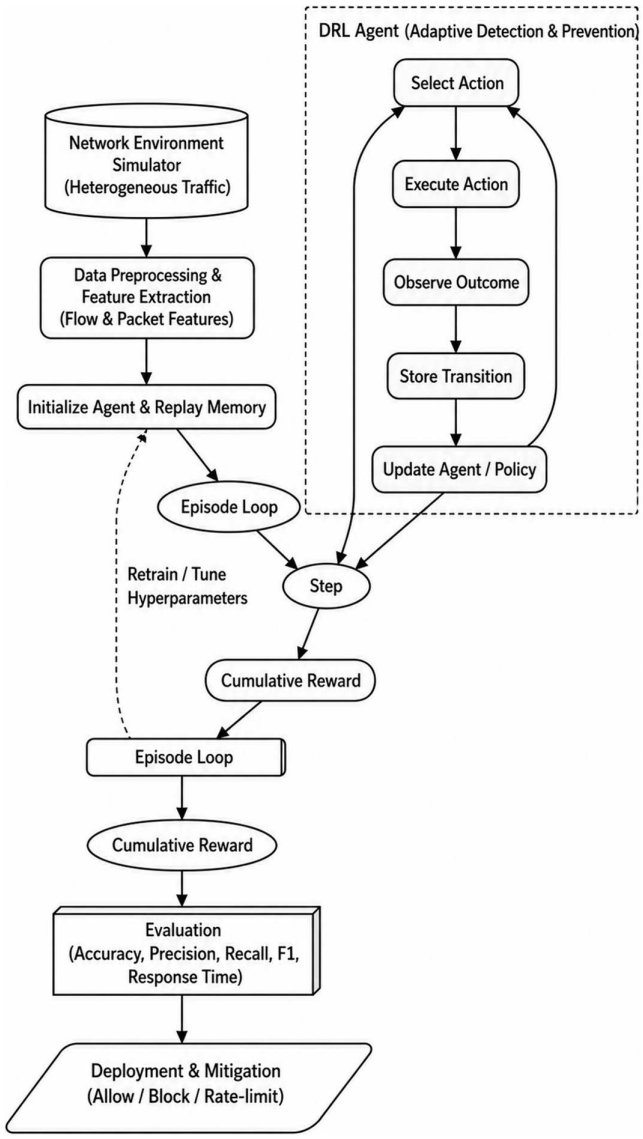
The general system architecture is provided in Figure 2, and the specifics of the operational and learning workflow are presented in Figure 3. These numbers reflect the methods of collecting, processing, learning, and transforming network traffic into adaptive mitigation responses.

3.1. Experimental workflow

The experimental workflow can start with the network traffic creation in heterogeneous settings, the consideration of which comprises IoT, edge, cloud, SDN, and 5G infrastructure (Figures 2 and 3). This generated traffic is preprocessed and converted into structured feature vectors, and this is the state representation of the DQN agent.

The agent addresses the environment by choosing mitigation behavior, monitoring system reactions, and modifying its policy based on experience replay. Tuning is done by continuous feedback and validation to make the best use of detection and mitigation effectiveness.

Figure 3
DRL-based adaptive framework for early DDoS detection and mitigation



3.2. Dataset design and traffic simulation

To build a realistic heterogeneous network space, the current study uses publicly available benchmark datasets, that is, CICDDoS2019, NSL-KDD, and UNSW-NB15. All these datasets include about 2.5 million labeled network flow records, both benign and in the context of various types of attacks. Out of this corpus, stratified sampling was used to create subsets of experiments of 50,000–300,000 flows, maintaining the original class distribution. The representation of a network flow as a five-tuple presenting a source IP, destination IP, source port, destination port, and protocol is encapsulated within a fixed-time window. To simulate heterogeneous environments, synthetic IoT, edge, and cloud traffic was created with the help of a network simulator and combined with benchmark traffic. The dataset partitioning was executed through chronological methods, which prevented data leakage; the partitioned data were divided into three parts with 70% used for training, 15% for validation, and 15% for testing. Such a time-based division provides independence between training and testing data.

The network simulation system, which creates synthetic traffic patterns for testing purposes, uses a setup that replicates IoT, edge computing, and cloud computing environments to create its simulation. Traffic patterns were designed based on statistical distributions that researchers observed in benchmark datasets that contained different flow duration, packet size, and protocol usage patterns. Cross-layer attack traffic was created by combining actual attack patterns from CICDDoS2019 with simulated multi-domain interaction. The hybrid method produces traffic that accurately simulates actual heterogeneous network traffic patterns.

Although benchmark datasets provide a standardized basis for comparative evaluation, they may not fully capture the continuously evolving characteristics of real-world network environments. To partially address this limitation, the proposed framework was additionally evaluated using heterogeneous traffic simulations that emulate dynamic interactions among IoT devices, edge nodes, cloud services, SDN controllers, and 5G infrastructures. Temporal traffic variations, protocol diversity, and changing attack distributions were incorporated into the simulation process to improve environmental realism. Nevertheless, the authors acknowledge that simulated environments cannot completely replicate operational network conditions, and therefore, the reported performance should be interpreted as a controlled experimental assessment rather than a direct indicator of deployment-level performance.

3.3. Preprocessing and engineering of data

Systematic preprocessing and feature extraction of raw traffic records were used to convert them into structured feature vectors. In preprocessing, records that had multiple copies and those with missing values were deleted, the media was used to fill in missing values, the continuous features were normalized with Z-score normalization, and the categorical variables were encoded with one-hot encoding. Flow-level and packet-level features were both conveyed, such as flow duration, the number of packets, the rate of packets, the inter-arrival time, the number of bytes, the protocol type, the destination port, and the length of packets. The state at time t is given by the following equation:

$$s_t = [f_1, f_2, \ldots, f_m] \tag{1}$$

The state vector combines temporal, statistical, and protocol-related traffic characteristics derived from both flow-level and packet-level observations. Feature selection was performed using correlation analysis and Random Forest importance ranking to reduce redundancy while preserving discriminative information relevant to DDoS detection. The resulting state representation provides the DRL agent with a compact yet informative description of current network behavior for adaptive decision-making.

The process of feature selection used two methods, which were statistical correlation analysis and model-based feature importance assessment. The process of selecting important features used correlation thresholds to eliminate highly correlated redundant features, whereas Random Forest feature importance scores were used to identify essential features. The process of state representation in the DRL agent used only the most relevant temporal features together with statistical features and flow-based features.

3.4. Formulation of deep reinforcement learning

The DDoS detection and mitigation issue is represented as: $(\boldsymbol{S}, \boldsymbol{A}, \boldsymbol{P}, \boldsymbol{R}, \boldsymbol{\gamma})$. The MDP is characterized by the following:

1) State Space (S), Action Space (A), and Reward Function (R)

Each state is an instantaneous network state that is comprised of normalized traffic features as obtained. The discrete action space is given as:

$$A = \{0, 1, 2\} \tag{2}$$

where 0 means open traffic, 1 means closed traffic, and 2 means rate-limiting traffic. The rewarding function balances the QoS and detection accuracy as follows:

$$(s_t, a_t) = \begin{cases} +2, & \text{Correct attack detection and mitigation} \\ +1, & \text{Correct allowance of benign traffic} \\ -2, & \text{Missed attack (false negative)} \\ -1, & \text{Incorrect blocking of benign traffic} \end{cases} \tag{3}$$

The reward values were selected to balance attack detection effectiveness and service availability. Higher positive rewards were assigned to correct attack detection and mitigation actions, while larger penalties were imposed on missed attacks because false negatives may lead to significant network disruption. Smaller penalties were assigned to incorrect blocking decisions to discourage unnecessary restriction of legitimate traffic. This reward structure was designed to guide the agent toward maximizing long-term network security while maintaining acceptable QoS levels.

2) Episode definition

In every training episode, 5000 successive flows are processed. When this limit is achieved, or when the cumulative reward attains a value, an episode is ended.

3.5. DQN-based learning algorithm

Each experiment uses a DQN framework to achieve reproducibility and clarity of the algorithm. The overall learning process is concluded in Algorithm 1.

Algorithm 1: DQN-Based Adaptive DDoS Detection Framework

Input:
Environment E, state space S, action space A, replay buffer M, learning rate α, discount factor γ, exploration rate ϵ, number of episodes N
Output:
Optimized network parameters θ^*
1. Initialize Q-network $Q(s, a; \theta)$ with random weights
2. Initialize target network $Q'(s, a; \theta^-)$
3. Initialize replay buffer $M \leftarrow \emptyset$
4. For episode = 1 to N do
  4.1 Reset environment and observe initial state s_0
  4.2 For each time step t do
    a. With probability ϵ, select a random action a_t
    b. Otherwise, select
$a_t = \arg\max_a Q(s_t, a; \theta)$
    c. Execute action a_t in environment E
    d. Observe reward r_t and next state s_{t+1}
    e. Store transition (s_t, a_t, r_t, s_{t+1}) in M
    f. Sample a mini batch from M
    g. Compute target value:
$y_t = r_t + \gamma \max_{a'} Q'(s_{t+1}, a'; \theta^-)$
    h. Update network parameters by minimizing loss:
$L = (y_t - Q(s_t, a_t; \theta))^2$
    i. Every C steps, update target network: $\theta^- \leftarrow \theta$
  4.3 End For
5. End For
6. Return optimized parameters θ^*

The Q-value update rule is given by:

$$Q(s_t, a_t) \leftarrow r_t + \gamma \max_{a'} Q(s_{t+1}, a'; \theta^-)$$

Experience replay and target network synchronization are used to stabilize training and reduce learning oscillations.

3.6. Hyperparameter optimization and training

The training was performed on 2000 episodes in a replay memory of 100,000 transitions. Network updates were performed in mini-batches of 64, and parameter maximization was done via the Adam optimizer with a learning rate of 0.0001. The grid search was used to optimize hyperparameters on the validation dataset. Termination was done in cases where no improvements were made in the performance of the validation over 50 consecutive episodes. Five independent random seeds were used to repeat all the experiments to make it robust and minimize performance variance. In order to test the adaptability of the online models, further experiments were performed, training the model on the previous traffic and testing it on the temporally shifted datasets with altered attack patterns.

The system showed two characteristic behaviors that matched the patterns of DDoS attacks that evolved in actual network environments. The research examined two aspects when assessing the framework's adaptive capabilities through its performance decline and subsequent restoration period. The last hyperparameter setup is the one used in all experiments and is summarized in Table 3.

Table 3
Hyperparameter settings for the proposed DQN model

Parameter	Value
Learning rate (α)	0.0001
Discount factor (γ)	0.95
Exploration rate (ε)	1.0 → 0.05
Replay memory size	100,000
Mini-batch size	64
Neural network layers	3
Activation function	ReLU
Optimizer	Adam
Training episodes	2000
Target network update frequency	500 steps

The researchers executed hyperparameter tuning through a grid-search approach, which they applied to the validation dataset. The researchers tested different combinations of key parameters, which included learning rate, discount factor, batch size, and exploration rate, until they found the best settings. The researchers selected final parameter settings by evaluating two factors, which included convergence stability and validation

performance, to achieve results that could be reproduced and maintain system reliability.

Hyperparameter selection was guided by validation performance, convergence stability, and computational efficiency. Multiple candidate configurations were evaluated through grid-search exploration of learning rate, discount factor, batch size, replay memory size, and exploration parameters. The final configuration was selected based on consistent validation performance across repeated experimental runs and stable learning behavior throughout training.

3.7. Statistical validation: Wilcoxon Signed-Rank Test

The signed-rank test compared the performance results of 10 experimental runs that used the same data split and random seeds. The null hypothesis states that performance remains unchanged, while the alternative hypothesis shows that the proposed model demonstrates superior performance. The researchers established 0.05 as their chosen level of significance.

The framework operates with high computational efficiency, which enables real-time inference to function with only minimal processing requirements, thus making it appropriate for use in resource-limited situations that exist in IoT and edge network environments.

3.8. Evaluation of metrics and experimental environment

The metrics used to measure performance were accuracy, precision, recall, macro F1-score, false positive rate, detection latency, and throughput under attack. These measures evaluate classification dependability, mitigation effectiveness, and real-time responsiveness. The manner in which the study is conducted is experimental. Experiments were carried out under the same conditions. PyTorch and Python 3.9 were the implementation tools on Ubuntu 20.04. Tests were conducted on an NVIDIA RTX graphics card, 16 GB of memory, Intel i7 processors, and 32 GB of RAM. The time taken in training varied between 6 and 8 h per run. Experiments were fixed on random seeds, preprocessing pipelines, and data splits to be reproducible.

To improve reproducibility, all experiments were conducted using fixed random seeds, identical preprocessing pipelines, consistent train-validation-test partitions, and standardized hardware and software configurations. These measures were adopted to minimize experimental variability and ensure fair comparison across evaluation scenarios.

4. Results and Discussion

4.1. Performance analysis compared to ML/DL models

The study used the same testing conditions to compare the performance of Logistic Regression, SVM, Random Forest, multilayer perceptron (MLP), and the DQN-based DRL framework, which the researchers developed in Figures 4, 5, and 6. In the context of inactive evaluation, the Random Forest classifier has the best classification rate (around 99%), and its ability to perform as an ensemble learner and its quality on structured tabular data are the reasons. Nonetheless, its performance is still tied to fixed training distributions and not intrinsically dependent on changing traffic patterns. The suggested DRL model is consistent with 96–98% accuracy with a test flow size of between 50,000 and 300,000.

In contrast to the static classifiers, the DRL agent develops a policy that is constantly updated depending on the environmental feedback. This allows scalability to be stable and more robust when dealing with heterogeneous traffic and changing traffic over time. Accuracy as high as between 96.5 and 97.5 says our MLP model is competitive in performance, but, then, there is a need for any new attack behavior to be retrained offline. SVM has an average performance (95.5–96.4%) but has higher latency with high-traffic load. Logistic Regression indicates slightly less accuracy (94–94.5%) as it has low capability of modeling nonlinear and intricate relationships that exist in multi-vector DDoS attacks. Such findings suggest that whereas the Random Forest is a strong performer in the case of a stationary setting, the suggested DRL

Figure 4
Comparative accuracy performance of machine learning and deep reinforcement learning models for DDoS detection

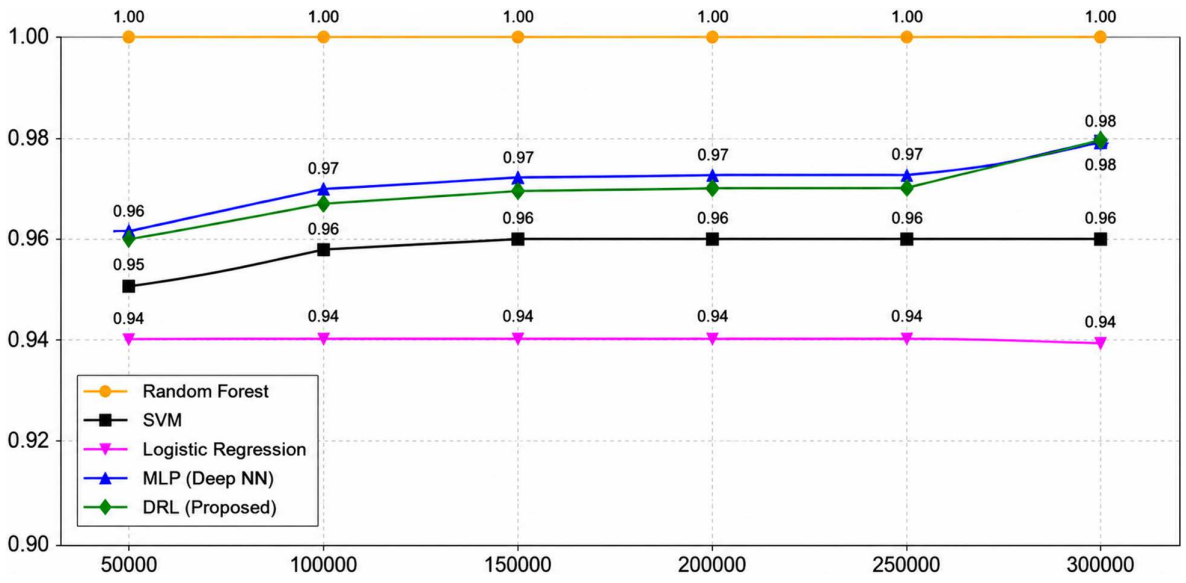


Figure 5
Comparative accuracy and macro F1-score of DDoS detection models

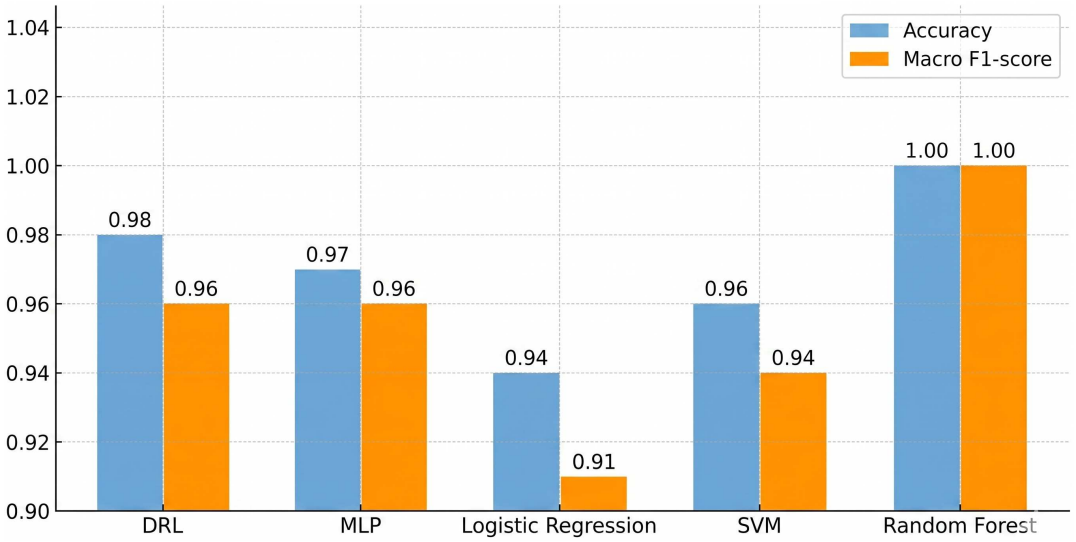
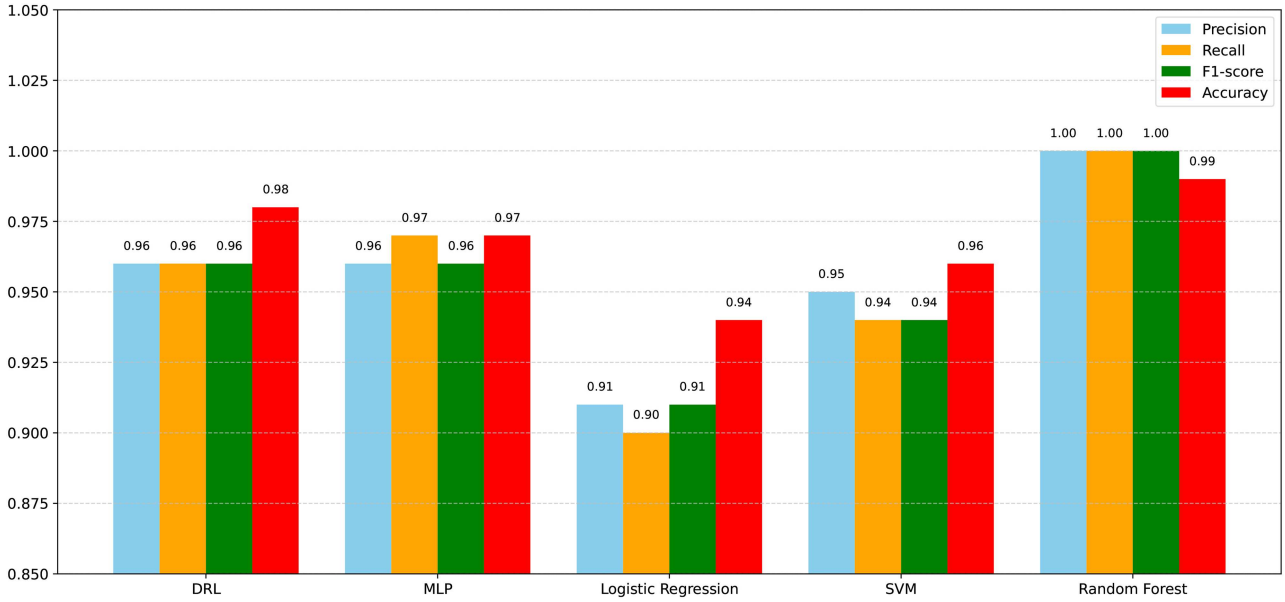


Figure 6
Detailed comparative evaluation of precision, recall, F1-score, and accuracy across DDoS detection models



framework offers greater adaptability and stability in the case of a dynamic heterogeneous environment.

The framework that we propose for evaluation purposes needs to include current DL and RL methods, which exist as the most advanced technologies according to Table 4. Although the primary experimental comparison was conducted against representative ML and DL baselines, recent adaptive cybersecurity studies have increasingly adopted RL and advanced DL architectures, including DQN, PPO, Actor–Critic, Soft Actor–Critic (SAC), and Transformer-based intrusion detection frameworks. Direct experimental comparison with all such approaches was not feasible because many reported studies utilize different datasets, traffic distributions, feature spaces, and evaluation protocols, making strict reproduction difficult.

To provide broader context, a literature-based comparison with recent state-of-the-art adaptive DDoS detection approaches has been included. This comparison enables evaluation of the proposed framework relative to contemporary DRL-based solutions while acknowledging differences in experimental settings. The results indicate that the proposed framework achieves competitive detection performance while emphasizing cross-domain

Table 4
Statistical significance analysis (Wilcoxon signed-rank test)

Comparison	*p*-value
DRL vs. Random Forest	0.041
DRL vs. MLP	0.033
DRL vs. SVM	0.017
DRL vs. Logistic Regression	0.008

adaptability, stability-aware learning, and heterogeneous network deployment capabilities.

The system includes transformer-based models together with advanced DRL techniques, which show exceptional performance results during controlled testing. The methods need especially high computing power together with extensive training datasets, which restrict their use in environments where multiple types of networks operate simultaneously. The proposed framework achieves detection accuracy together with scalable performance and real-time adaptability, which makes it suitable for practical use across IoT networks, edge systems, and 5G networks.

Figure 4 shows the classification accuracy of five DDoS detectors tested by using test flows of 50,000–300,000 samples. Random Forest model is the most successful in terms of the highest static accuracy (around 99%) in all tested flow sizes. This proves the efficiency of the ensemble-based learning method in the case of structured traffic data. The suggested DQN-based DRL model has been found to have a stable and scalable performance, with the accuracy improving as the flow rises from approximately 96% at 50,000 flows to approximately 97.5% at 300,000 flows. This convergence stability and strength imply a gradual increase with the size of the dataset. On a comparable note, the MLP model is also shown to be competitive with an accuracy of 96.5–97.5 with similar behavior in larger flow sizes. SVMs achieve average performance because their accuracy increases with higher flow size, while Logistic Regression achieves lower accuracy between 94% and 94.5%, which demonstrates its inability to accurately model complex traffic patterns. In general, the findings show that the performance of Random Forest is high when the evaluation is conducted under a static situation, but the suggested DRL framework can demonstrate competitive accuracy with greater scalability and flexibility in heterogeneous and dynamic network scenarios.

Figure 5 provides a comparison of different DDoS detection models based on their accuracy and macro F1-score. Ensemble and DL models are proven to be more effective than traditional machine learning methods. Random Forest has a good classification performance according to the static evaluation conditions; the accuracy is around 99%, and the value of the F1-score is high, which indicates its successful application in the organized study of traffic. The suggested DRA architecture provides competitive accuracy, approximately 98, and a macro F1-score of approximately 96, which means that accuracy is balanced between the majority and the minority attack classes. Equally, the MLP model also attains similar outcomes, showing an accuracy of approximately 97% and consistent values of F1-score, but its flexibility depends on offline retraining. SVM shows mediocre results, where the accuracy is about 96% and the F1-score is about 94%, and the lowest classification results are obtained by Logistic Regression, which has an accuracy of about 94% and an F1-score of about 91%. This is an indication of its low ability to simulate complex and nonlinear traffic patterns related to multi-vector DDoS attacks. In general, although Random Forest is competitive in terms of baseline performance in the case of a static setting, the suggested DRL framework is competitive in terms of accuracy and better adaptability and generalization in the context of heterogeneous networks that dynamically change.

Figure 6 will offer a comparative analysis of the metrics of classification—accuracy, precision, recall, and F1-score—of all the considered models. Random Forest shows the best performance in the case of fixed evaluation conditions, where the accuracy is around 99% and high values of precision and recall rates. This is indicative of the ability of ensemble-based learning in structured tabular traffic data. The suggested DQN-based DRL model has a competitive score, with an accuracy of approximately 98% and an equal level of precision and recall (around 96%). Balanced metric distribution implies that there is a fixed detection capability between the benign and attack classes. The error probability of false positives and false negatives is minimal. It is notable that, in contrast to fixed models, the DRA model allows flexibility in changing traffic states. These are also similar to the MLP model, which has an accuracy of about 97% with a comparable accuracy and recall pair. Nevertheless, its capacity for adaptation is still limited to offline retraining. SVM has moderate performance (accuracy ≈96%), whereas Logistic Regression has the lowest performance (accuracy ≈94%), which is caused by the inability of this method to model most nonlinear traffic trends. Random Forest proves to be better in the context of static testing. The suggested DRL framework offers competition in terms of accuracy and comes with increased flexibility; thus, it would be especially useful in the context of dynamic and heterogeneous networks.

Figure 7 shows the distribution of the flow duration of benign and malicious network traffic classes. Longer and more consistent flow times are the traits of benign traffic, which depict the sustained and uninterrupted communication behavior as was characteristic of legitimate user activity. Alternatively, denial-of-service (DoS) attacks such as DoS Hulk, DoS GoldenEye, DoS Slowloris, and DoS Slowhttptest have high density at short flows. These temporary streams are associated with bursty and rapid transmission patterns of packets aimed at flooding target resources. There are some similarities between the temporal distributions of PortScan and bot traffic, including short and intermittent flows, which are also in line with automated probing and scanning behavior. Web-based attacks, including SQL Injection, Cross-site Scripting, Brute Force, and infiltration attempts, tend to have relatively shorter flow durations, with a few cases overlapping with harmless traffic.

Overall, these time-specific aspects prove the discriminatory nature of flow duration in distinguishing benign and malicious activities. The notable difference between long stable flows and short attack bursts is a supporting factor of the suitability of flow-based temporal characteristics in identifying high-rate DDoS and scanning attacks in heterogeneous networks.

Figure 8 gives the logarithmic density of flow durations in benign and malicious traffic classes. Normal user traffic is marked by minimized and steady communication sessions characteristic of benign traffic, which has a high density during medium and long flow periods. Conversely, volumetric DoS attacks (such as DoS Hulk, DoS Slowloris, and DoS GoldenEye) have strong density peaks at very short flow times (below one second). These trends show sporadic and violent transmission patterns with the aim of depleting target resources quickly. On the same note, PortScan traffic indicates a high concentration at brief flow periods, which is typical of automated probing and scanning processes. The local density peaks of infiltration traffic and web-based attacks (SQL Injection, XSS, and Brute Force) are short-lived and transient communication patterns of covert attempts at intrusion. The density distributions of Bot and Secure Shell (SSH)-Patator traffic have some overlap with benign traffic and yet have relatively short flow times. Overall, the unique density distributions in the traffic classes indicate the high discriminatory power of flow duration to distinguish between benign and malicious behavior. The results of these findings support the use of temporal flow analysis to identify DDoS and other cyberattacks early in the heterogeneous network setting.

Figure 7
Flow duration distribution across benign and malicious network classes

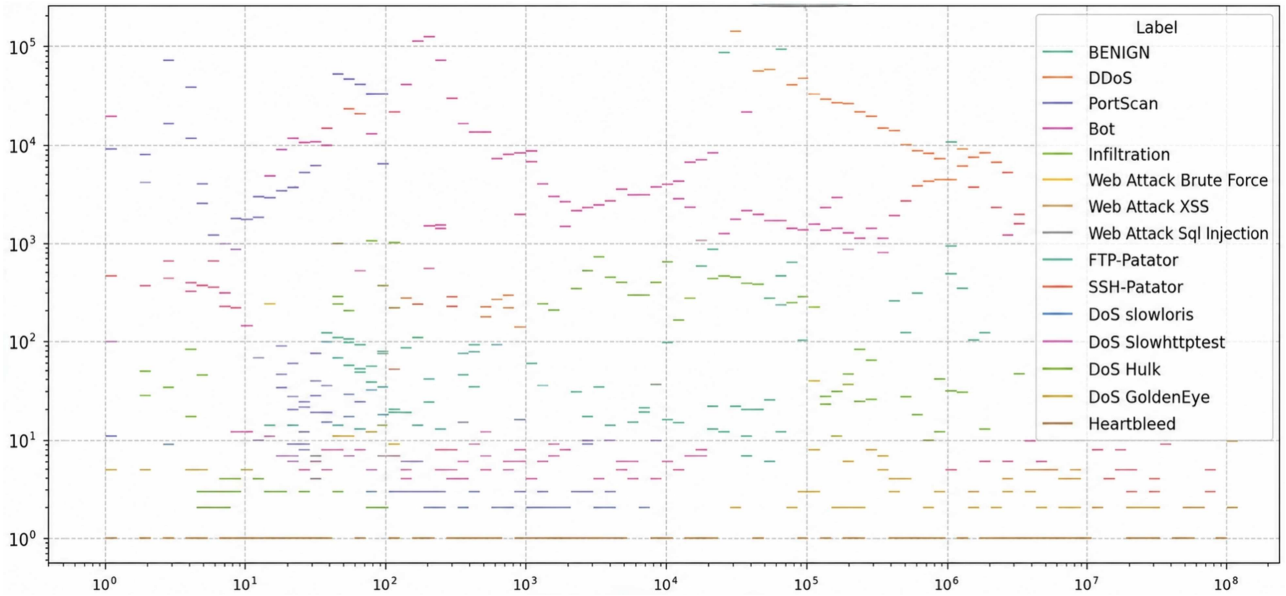
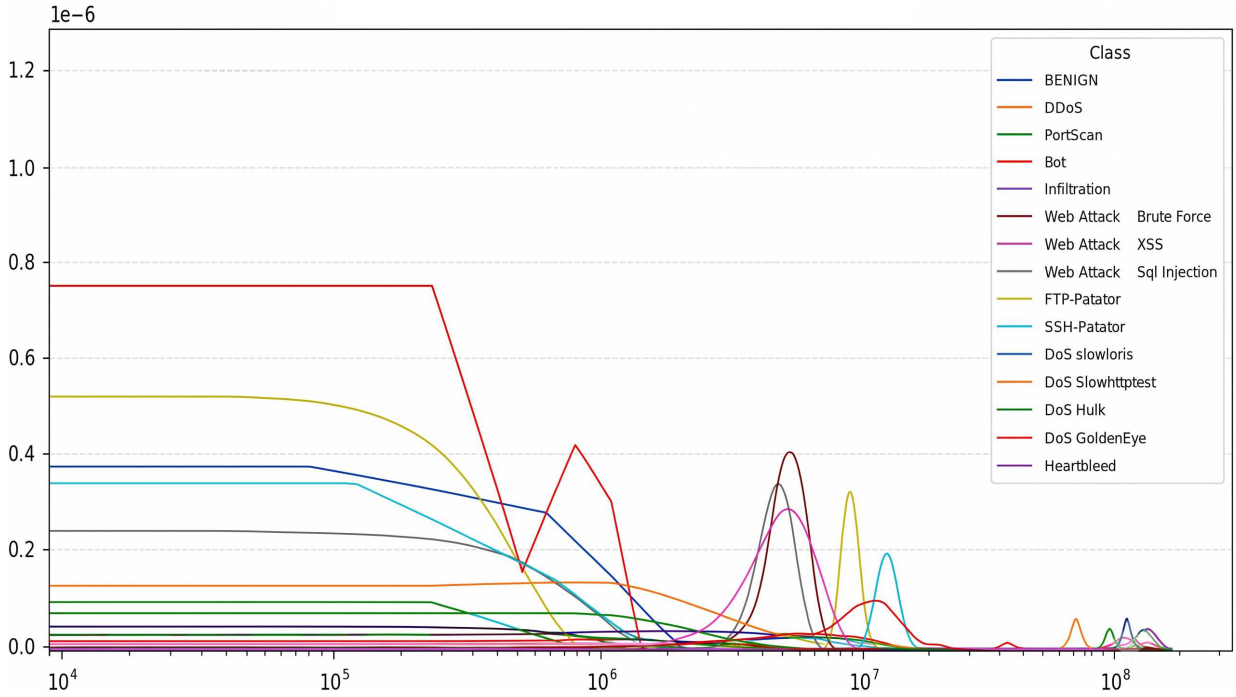
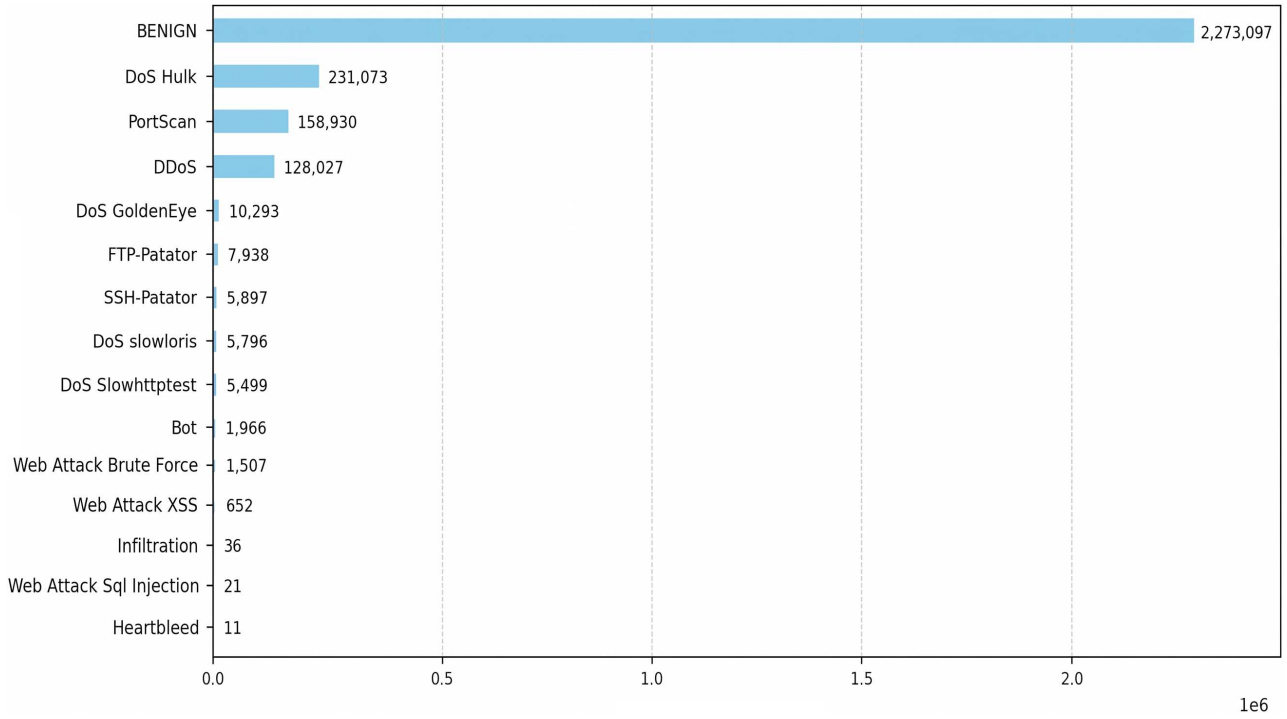


Figure 8
Flow duration density distribution across network traffic classes



This is shown in Figure 9, where the distribution of network traffic samples in the dataset is shown by class. Benign traffic type is the most frequent, with 2,273,097 samples, implying a skewed data distribution. DoS Hulk is the biggest category in terms of the quantity of samples (231,073), followed by PortScan (158,930) and generic DDoS (128,027). In the middle range are DoS GoldenEye (10,293), FTP-Patator (7,938), and SSH-Patator (5,897). Others like DoS Slowloris (5,796), DoS Slowhttptest (5,499), and Bot (1,966) are not so common. There are also Web-based attacks with a relatively low number, such as Brute force (1,507), XSS (652), and SQL Injection (21), and Infiltration (36) and Heartbleed (11) are extreme and minority classes. This strong class imbalance captures the disproportionate incidence of attacks in real-life network settings, with the most common attacks being the high-volume DoS attacks as compared to the stealthy intrusion attempts. This reduces the accuracy with which minority classes

Figure 9
Distribution of samples across benign and malicious attack classes



can be sensitized, and this underscores the vitality of adaptive learning routines and weighted appraisal indicators when formulating efficient DDoS detection models.

The correlation of the most important network traffic characteristics employed in the detection of DDoS is shown in Figure 10. There is a high correlation (1.00) between Total Forward Packets and Total Backward Packets, which means that there is great redundancy between the two attributes of bidirectional flow. This implies that the two attributes are similar in the information they are conveying and can be partly replaced when making a model. The positive correlation between packet length mean and flow duration is moderate (0.43), which implies that the packet size of longer network sessions is more likely to be stable. Linear correlation, on the other hand, is weak (0.23) between flow bytes/s and flow packets/s, implying that the rate at which both bytes and packets are being transmitted gives different information about the traffic. Moreover, the low correlation rates between flow duration and flow packets/s (−0.12) and between flow duration and flow bytes/s (−0.03) show that longer flows do not automatically imply heavier traffic.

This shows that both temporal and volumetric behavior should be considered to model network behavior. Although not all the features are essential, the time-based, statistical, and throughput-based features, such as flow duration, packet length statistics, and rate-based features, provide complementary information in the classification. These relations aid in effective feature choice and improve the learning capacity of ML-based and RL-based DDoS.

Figure 11 groups flow live traffic classes using packet behavior. Normal traffic is diffused, and that is a characteristic of human communication, as it possesses greater packet lengths as well as flow times. The attack classes, including DDoS attacks, DoS Hulk, DoS GoldenEye, and DoS Slowloris, are characterized by short flow durations and average packet lengths due to their bursty and aggressive traffic behavior. PortScan and bot traffic are also characterized by higher variability of packet size and shorter time intervals of flows as compared to probe or automated traffic scanning. The web attacks (SQL Injection, XSS, Brute Force) or the attempt to penetrate the web reveals smaller clusters with shorter flow times and shorter average length of the packets, which means that there is clandestine activity. The disparity in the feature space shows that packet statistics and flow duration can be used as stand-ins to determine whether malicious or opportunistic traffic or benign and normal network traffic is in operation.

Figure 12 shows the statistics of the network traffic in the top 20 destination ports. The findings have shown that traffic is highly concentrated, with very few service ports that are being used intensively. Port 53 (DN) is the most prevalent in the dataset (957,971 flows), which constitutes over half of the traffic. The next ones are Port 80 (HTTP) and Port 443 (HTTPS), with 618,934 and 505,710 flows, respectively, representing the popularity of web-based and domain name resolution operations. Other ports utilized, such as Port 123 (NTP), Port 22 (SSH), and Port 21 (FTP), have much lower frequencies, and this means that they are of little operational value but are present in the network. Ports that are less frequently used, such as Port 8080 and Port 5353, add slightly to the total traffic. The high emphasis on DNS, HTTP, and HTTPS services indicates that such important elements of communication are the key targets of DDoS attacks. Therefore, it is critical to monitor and secure such high-traffic ports to increase the resilience of the network and to provide service in heterogeneous networks.

Random Forest classifier considers the packet statistical characteristics to be the most important characteristic of DDoS detection (Figure 13). The first 20 attributes are headed by packet length variance (0.055), then standard deviation (0.052), and maximum length (0.051). Moreover, such parameters of network

Figure 10
Correlation heatmap of key network traffic features

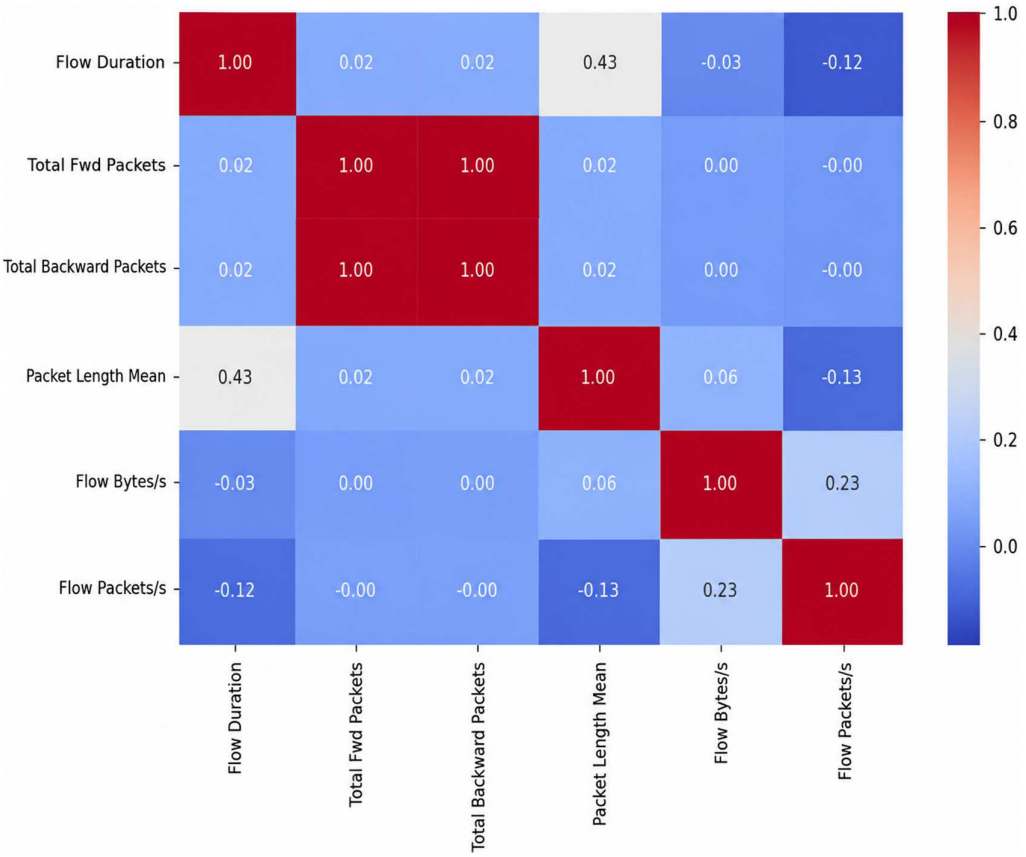


Figure 11
Scatter plot of flow duration vs. packet length mean across traffic classes detection models

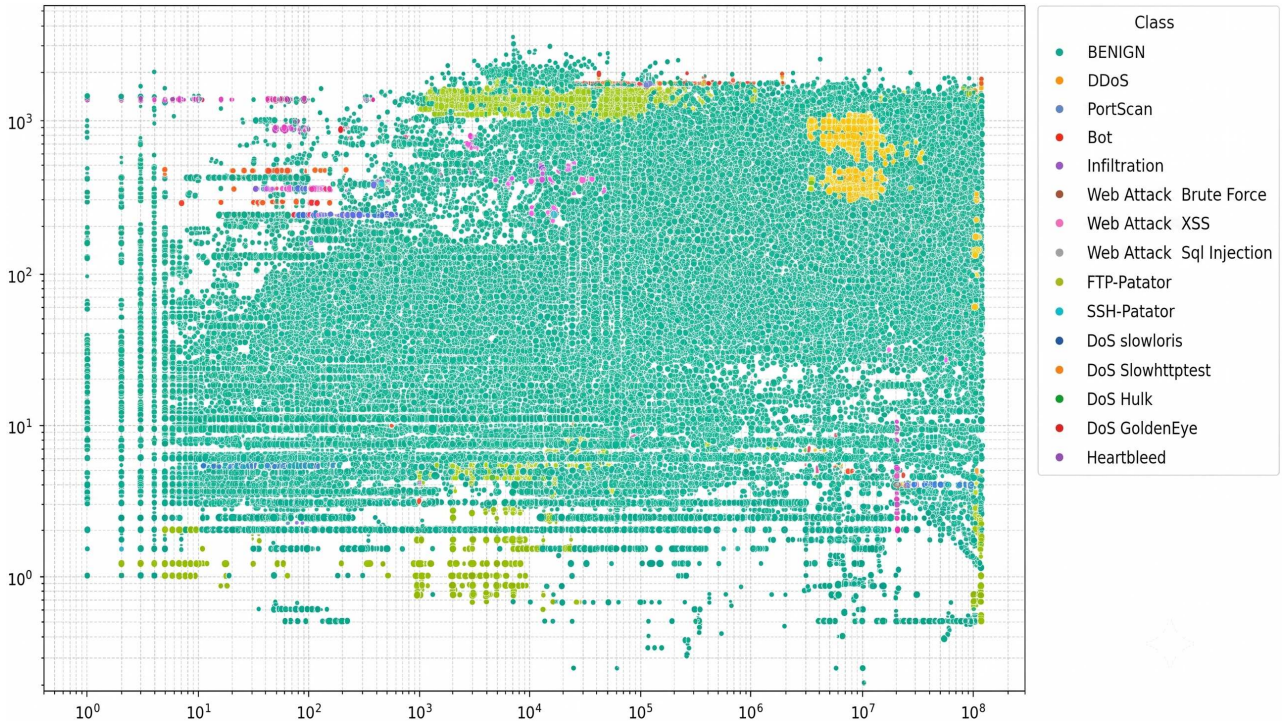


Figure 12
Distribution of Top 20 destination ports in network traffic flow

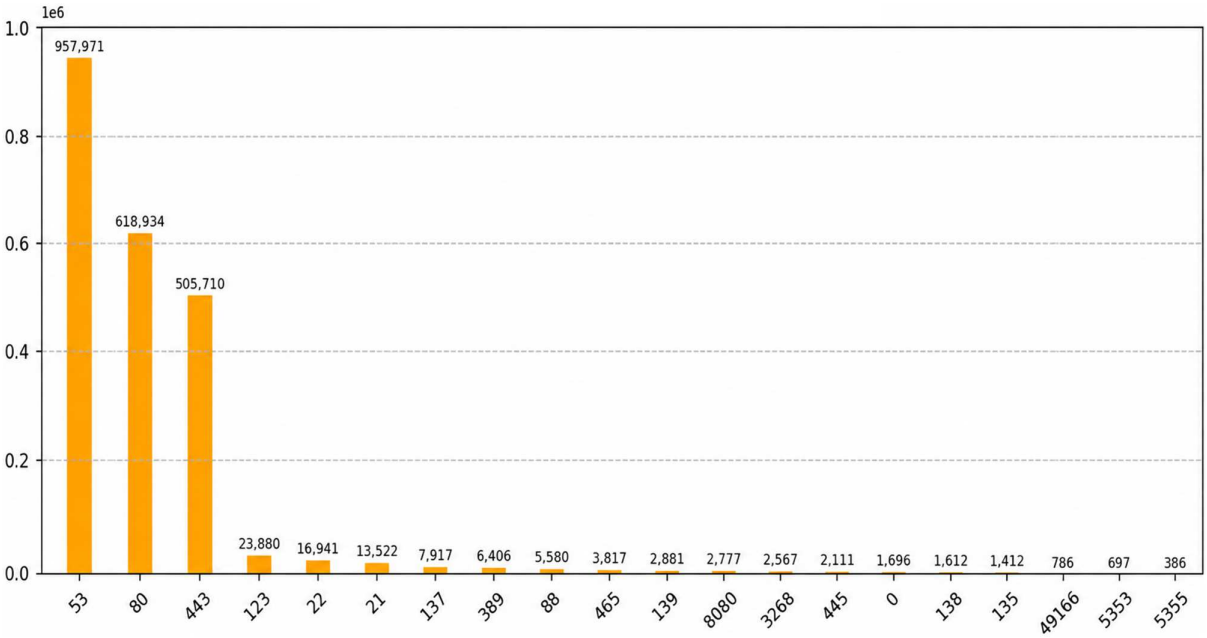
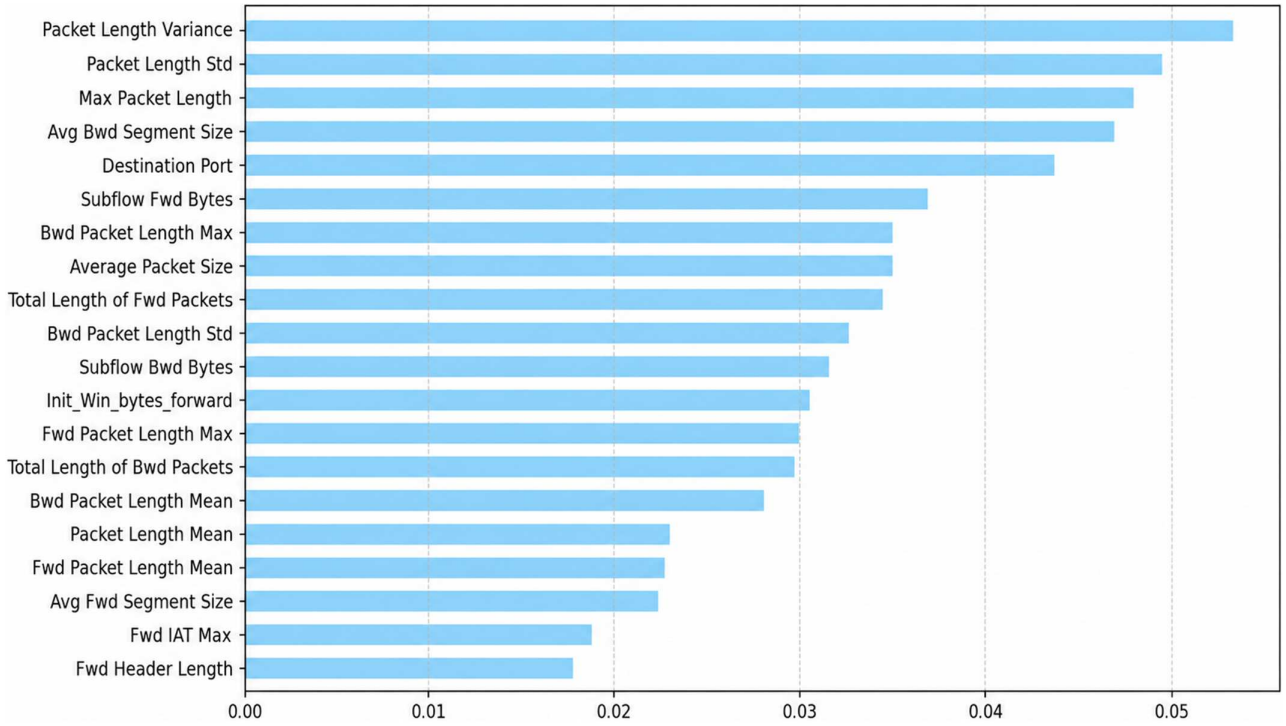


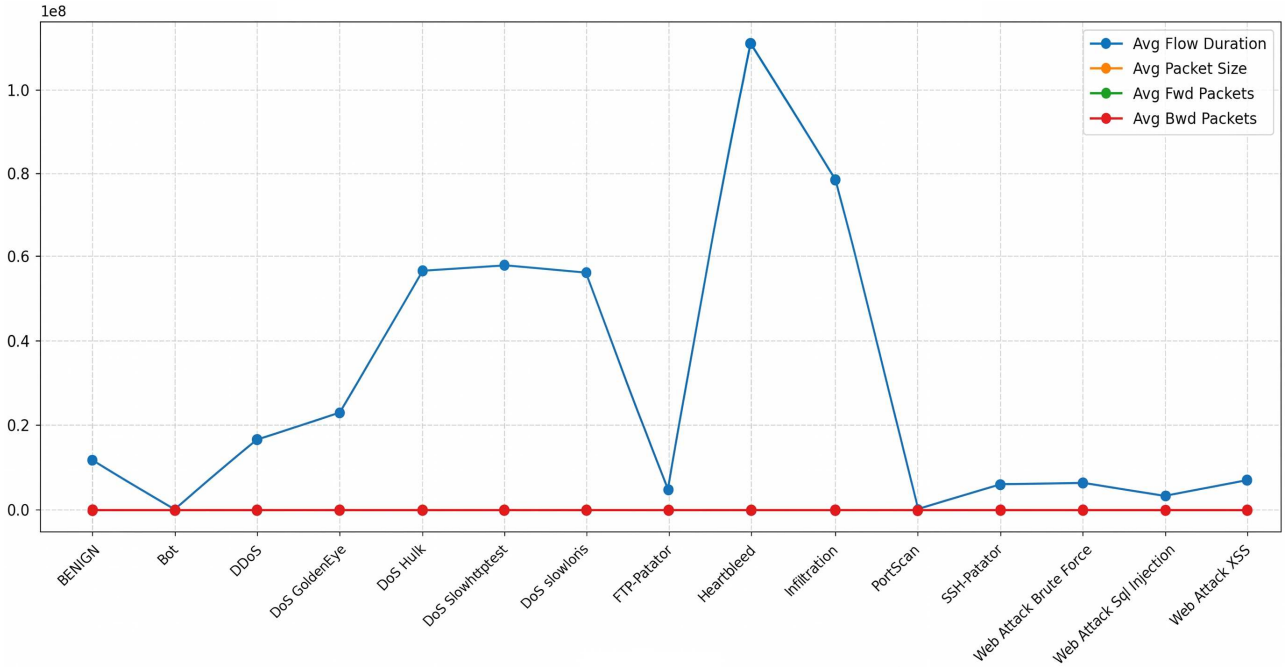
Figure 13
Top 20 feature importances for attack detection



behavior as average backward segment size (0.045) and destination port (0.041) were necessary. This demonstrates that the properties of packets and the distribution of traffic have to be changed to detect the malignant and benign flows. Other measurements, such as Subflow Forward Bytes (0.039) and Backward Packet Length Max (0.037), show asymmetric bidirectional flow of DDoS. Forward Inter-Arrival Time Max (0.018) and Forward Header Length (0.017) proved that timing anomalies and protocol-level characteristics can also indicate an attack, although not at a statistically significant level. Collectivism of variances of packets, flow, and destination-specific characteristics is most accurate in detecting DDoS attacks on heterogeneous networks regardless of being harmful.

Figure 14 shows DDoS detection behavior for different traffic classes through network traffic metric analysis. The results show that Heartbleed traffic has an average flow duration of

Figure 14
Comparison of average network traffic metrics across benign and attack classes



1.0×10^8, which exceeds the average flow duration of 1.2×10^7 for benign flows, 5.2×10^7 for DoS attacks, and 2.0×10^7 for DoS GoldenEye attacks, which demonstrates its ability to maintain hidden network communication for extended periods. Infiltration traffic shows increased flow durations of approximately 8.0×10^7 which demonstrate that attackers use persistent and covert methods to execute their attacks.

Web Attack–XSS and SQL Injection lightweight attacks show low flow durations of approximately 1.0×10^6, which creates partial traffic overlap with benign traffic and makes their detection difficult. The detection system needs adaptive mechanisms to identify the traffic behavior that shows only minor changes.

The packet size remains stable across different classes, but the flow duration and traffic intensity variations serve as essential features for detecting attacks. The detection of both high-rate and stealthy attacks depends on identifying the distinct temporal patterns that separate short-lived benign interactions from extended attack traffic. The system design of the proposed detection framework receives support from these fundamental observations, which show their value to research. The identified variations in flow duration and traffic patterns enable the DRL agent to learn context-aware policies that adapt to both rapid and persistent attack behaviors. The framework improves early detection and decision-making capabilities across various network environments by using dynamic traffic characteristics instead of static threshold measurements.

The research indicates that flow duration along with traffic metrics is a descriptive component and a critical component that allows the system to identify DDoS attacks through adaptive real-time detection mechanisms in the proposed system.

The ability to detect threats in actual time operates as an essential factor that exists alongside the system's capacity to classify different types of threats. The proposed framework demonstrates stable response behavior under varying traffic volumes, with policy decisions generated in near real time due to the use of pre-trained Q-networks. The system enables protection against high-rate DDoS attacks, which use multiple attack vectors to launch their attacks.

4.2. Statistical significance validation (Wilcoxon signed-rank test)

To test the hypothesis of statistical significance in the observed differences between the proposed DRL framework performance and performance of the performance of the baseline ML/DL models, a Wilcoxon signed-rank test was performed on experimental results of paired experiments, which took place in the same evaluation settings.

The hypotheses are formulated in the following way:

1) **Null Hypothesis (H):** There is no statistically significant difference between the performance of the DRL model and the baseline models.
2) **Alternative Hypothesis (H1):** The suggested DRL model proves to have statistically significant improvement.

The p-values are all less than the significance level of 0.05, meaning that there is statistical significance in the performance differences that are experienced in adaptive and temporally shifted evaluation situations. These findings confirm that the enhanced robustness of the proposed DRL framework is not a random occurrence but is related to its online learning and ability to adapt its policy.

4.3. Superiority of the DRL framework over ML/DL models

The DRL-based framework shows its ability to perform well in dynamic network environments because it uses its system to change its detection methods based on real-time traffic updates without needing to perform complete system updates. The main

value of this study established an adaptive cross-domain detection system that operates in different network environments while assessing its multiple performance metrics.

The DRA agent builds its decision-making capacity by continuously interacting with the network system as traditional ML/DL models set fixed limits on decisions that require frequent system updates to accommodate new network traffic. The system uses this capability to handle traffic changes, which include new DDoS attack methods and multiple simultaneous attack patterns. The system needs dynamic adaptation to handle changing conditions because dynamic adaptation does not guarantee better results in situations with unpredictable traffic patterns. The framework establishes controlled policy development processes that create dependable detection methods for operational use. The proposed framework provides network domain support for multiple domains, which include IoT networks, edge computing, cloud infrastructure, and 5G systems, whereas existing studies focus on single-domain research, which examines either SDN networks or IoT environments.

The system enables cross-domain operation, which permits learning of policies based on environmental conditions that better match actual network operations because network traffic patterns and security threats differ across multiple domains. The framework systems focus on two critical components that enable them to protect against DDoS attacks before they develop into major security incidents. The system first tests its effectiveness through controlled experiments before moving to actual environments, which include different operational conditions. The work creates a practical impact through its unique performance evaluation method, which compares DRL systems with ML and DL systems instead of standard output measurements. The study introduces new research methods that combine adaptive RL with cross-domain capabilities, stability-focused learning, and real-time detection functions. DQN-based DRL along with its traditional model testing showed previous research results.

The actual deployment of systems needs to assess three main requirements, which include computational overhead, system scalability, and their ability to provide immediate system response. The DRL framework uses lightweight state representation together with controlled policy updates to achieve efficient performance across different traffic conditions. The system needs computational power during training, but it operates effectively during inference, which allows its implementation in both edge and cloud distributed systems.

Unlike conventional ML baselines that primarily evaluate classification capability, adaptive DRL-based frameworks are designed to continuously interact with evolving network environments. Therefore, the superiority of the proposed framework should be interpreted not only in terms of classification accuracy but also in terms of adaptability, policy evolution, robustness to traffic variation, and cross-domain deployment capability. These characteristics provide additional practical value beyond traditional static evaluation metrics.

4.4. Limitations and risk analysis of DRL-based adaptive detection

The DQN-based DRL framework shows excellent adaptability and stability for working with unpredictable network environments, but its ability to learn dynamically presents particular challenges. The DRL agent establishes its detection method through ongoing updates, which involve incoming traffic patterns and reward feedback. The system achieves real-time adaptation through its current setup, which results in operational hazards when facing dynamic or hostile situations.

The DRL agent is mainly sensitive to two aspects that encompass concept drift and existence of noisy traffic data. The patterns of network traffic are subject to abrupt changes in the actual network operation of flash events, legitimate traffic surges, or attack strategies. The learning agent established a temporary suboptimal policy that causes a short-term decline in detection performance. The agent develops biased policies when reward signals come from misleading or imbalanced data, which leads to increased chances of false positives and false negatives.

Frequent policy changes create an additional challenge because they create unpredictable operational patterns. The detection model establishes new decision boundaries, which results in continuous online learning without established control mechanisms that handle system behavior. The detection system will become less consistent and less dependable because of this issue, which occurs in high-throughput network environments. Adversarial traffic patterns, which security attackers create for the purpose of disrupting the learning process, will make the DRL agent system less reliable because these patterns will allow attackers to exploit system weaknesses.

Adversarial attack scenarios create extra difficulties because attackers develop specific traffic patterns that they use to avoid detection through evasion attacks, and they create fake training data, which results in model performance degradation through poisoning attacks. The DRL agent becomes less dependable because it faces adversarial threats that need defenses based on adversarial learning mechanisms.

To address these challenges, the proposed framework incorporates several stabilization and control mechanisms. The policy update process is regulated using controlled learning rates and periodic update strategies to prevent abrupt behavioral shifts. A replay of experience mechanism is employed to make sure that the agent learns based on a richer set of past experiences instead of basing his/her learning on recent observations alone, which minimizes the effect of short-lived traffic abnormalities. Moreover, reward shaping methods are used to steer the learning process toward balanced performance in detection between benign and attack classes.

Moreover, a validation-driven update strategy is considered, where newly learned policies are evaluated against traffic validation before deployment, ensuring that performance does not degrade over time. Threshold-based decision constraints are also applied to maintain stability in classification outcomes. These measures collectively ensure that the adaptive nature of the DRL framework enhances detection capability while minimizing the risks associated with uncontrolled dynamic learning.

While the proposed framework incorporates stabilization mechanisms such as experience replay, controlled policy updates, validation-driven adaptation, and reward shaping, the present study primarily focuses on detection effectiveness and adaptive policy learning performance. A comprehensive quantitative investigation of convergence dynamics, training variance across multiple random initializations, and robustness under explicitly crafted adversarial traffic scenarios was beyond the scope of the current work. Although multiple independent experimental runs were conducted to reduce stochastic effects, detailed convergence statistics and adversarial robustness evaluations were not separately reported. Future studies will include learning-curve analysis, variance-based stability assessment, and adversarial stress-testing experiments to provide a more comprehensive understanding of long-term DRL behavior in operational network environments.

Overall, while dynamic RL introduces certain challenges, the integration of controlled learning mechanisms and validation strategies enables the proposed system to maintain stable, reliable, and high-performance DDoS detection in heterogeneous network environments.

Another important limitation concerns the use of benchmark datasets and simulated heterogeneous traffic environments. Although CICDDoS2019, NSL-KDD, and UNSW-NB15 are widely adopted for cybersecurity research, they may not fully represent the complexity, unpredictability, and temporal evolution of operational network infrastructures. Real-world deployments often exhibit concept drift, previously unseen attack behaviors, evolving protocol usage, and organization-specific traffic characteristics that cannot be completely reproduced through benchmark datasets. Consequently, while the proposed framework demonstrates strong performance under controlled experimental conditions, additional validation using real-time operational traffic traces and large-scale production environments is required to further assess its long-term generalization capability.

The research extends beyond previous studies that compared DQN-based DRL systems with standard ML methods through its new approach, which combines adaptive RL with multiple network systems that include IoT, edge, cloud, and 5G technologies. The proposed framework establishes a new approach that enables real-time policy changes and detects DDoS attacks in their initial phases, while it can function across multiple domains. The proposed method differentiates itself from standard DRL-based detection systems through its implementation of managed learning systems and its development of update methods that maintain system stability.

The DRL framework shows improved performance because it teaches sequential decision-making policies through environmental feedback learning. The DRL agent outperforms static models because it uses temporal dependencies to adapt to changing traffic patterns, which enables better detection of multi-vector and stealthy DDoS attacks. The model achieves better dynamic condition generalization through its adaptive behavior than it would through fixed decision-boundary methods.

4.5. Feature distribution and traffic behavior analysis

Figures 7–14 portray the discriminative ability of the chosen features by providing an analysis of exploratory traffic behavior. DDoS attacks with high rates are characterized by brief flow periods and bursty packet density, and benign traffic has longer and sustained patterns of sessions. The destination port analysis shows that legitimate traffic is based on DNS (53), HTTP (80), and HTTPS (443), which are the major targets of DDoS attacks. The analysis of feature importance also indicates that the variability of packets level like packet length variance and standard deviation is very critical in the classification of malicious and benign flows. Scatter plots and density distributions show that the malicious traffic is concentrated into tight clusters, whereas the benign traffic is more spread in the feature space. The behavior patterns can help the DRL agent utilize temporal, statistical, and volumetric characteristics to identify DDoS early and accurately in heterogeneous settings. To contextualize the suggested framework in the context of the current research, the performance outcomes are contrasted with the current state-of-the-art studies (2022–2025). Table 5 presents the literature reference values, according to journal reporting guidelines.

The analysis of feature distribution serves two purposes because it helps to explore data while it reveals the specific features that the DRL agent uses to enhance its detection and mitigation efforts.

Table 5 provides a literature-based comparison between the proposed framework and recent adaptive DDoS detection approaches, including DRL-based and transformer-based models. Although several studies report slightly higher accuracy values under specific experimental settings, most of these approaches were evaluated in single-domain environments such as SDN, IoT, or enterprise networks. In contrast, the proposed framework was designed and evaluated with the objective of supporting heterogeneous environments comprising IoT, edge, cloud, SDN, and 5G infrastructures. Therefore, the contribution of the proposed framework should be interpreted not solely in terms of classification accuracy, but also in terms of adaptive policy learning, cross-domain applicability, and stability-aware operation under evolving traffic conditions.

The evaluation uses traditional ML and DL models as base models for testing, but these models function as standard testing tools that assess adaptive learning systems. Recent transformer-based approaches, despite their powerful capabilities, require excessive computational resources and extensive training data, which restrict their use in various real-time operational settings. The selected baseline models enable an accurate and practical evaluation of the proposed framework.

The existing approaches demonstrate superior performance metrics when tested in controlled environments or with specific datasets. The methods presented in this study have restrictions because they operate only in fixed learning environments and single-domain network environments, which include SDN-based and IoT-specific frameworks that define their model retraining and deployment processes.

The proposed DRL-based framework functions across diverse network environments that experience continuous network

Table 5
Reported performance of recent DDoS detection approaches

Model	Dataset	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
CNN–BiLSTM	CIC-IDS2017 and UNSW-NB15	96.5	97.0	96.8	96.9
Soft Actor–Critic (SAC)	Custom SDN–IoT Dataset	97.5	96.4	97.0	96.7
SDN-DQN Mitigator	CICDDoS2019 (SDN)	98.4	97.7	96.3	97.5
Transformer-based IDS	CICDDoS2019 (reconstructed)	98.0	97.6	99.1	98.3
Proposed DRL framework	CICDDoS2019 + Heterogeneous Simulation	98.0	96.0	96.0	96.0

changes, while it integrates IoT systems, edge computing, cloud services, and 5G networks through its single adaptive detection framework. The key contribution of this work lies not merely in achieving competitive performance metrics but in enabling real-time policy adaptation, early-stage DDoS detection, and cross-domain generalization without requiring frequent offline retraining.

The framework implements stability-aware learning mechanisms that maintain performance at all times during shifting traffic patterns. The existing studies fail to address actual deployment difficulties that the framework needs to solve. The proposed approach provides a better solution for real-world heterogeneous network environments because it operates more effectively than some models, which show only slight accuracy improvements under controlled testing conditions.

The comparison shows that multiple top-performing models reach their best results under testing conditions, but the proposed DRL-based framework delivers results that match those achievements while showing a better ability to adapt and operate in different network conditions.

4.6. Generalization and real-world deployment considerations

The primary objective of this study was to evaluate the feasibility of adaptive DRL-based DDoS detection across heterogeneous network environments under controlled and reproducible experimental conditions. Benchmark datasets were selected because they provide standardized attack scenarios and facilitate fair comparison with existing literature. To improve environmental diversity, synthetic heterogeneous traffic representing IoT, edge, cloud, SDN, and 5G interactions was incorporated into the evaluation process.

However, real-world network environments are inherently more dynamic and may contain previously unseen attack strategies, traffic anomalies, and operational constraints. Therefore, the reported performance metrics should be interpreted as indicators of framework effectiveness under representative experimental conditions rather than absolute deployment performance. Future validation using live network traffic and operational cybersecurity infrastructures will be necessary to establish deployment-level generalizability and robustness.

5. Conclusion

The study introduces a DRL system that utilizes DQN technology to detect DDoS attacks at their initial stage and execute countermeasures across various network systems including IoT devices, edge computing, cloud computing, software-defined networking, and 5G networks. The system uses online policy learning together with traffic analysis at flow and packet levels to achieve real-time adaptive security monitoring, which can handle varying network conditions.

The experimental results show that the framework maintains its stable performance while achieving competitive results because it reaches an accuracy rate of 98% and maintains balanced precision, recall, and macro F1-score, which approaches 96%. The combination of Random Forest and MLP DL models with traditional ML models shows high performance when operating in controlled environments, but their fixed decision boundaries prevent them from handling changing traffic patterns. The proposed DRL framework supports ongoing policy changes, which makes it better suited for environments that experience both changing conditions and diverse operational patterns.

Dynamic learning does not result in better performance throughout every situation. DRL-based adaptation effectiveness can be affected by non-stationary traffic patterns, noisy data, and adversarial patterns, which have the potential to disrupt learning process stability. The framework employs controlled learning methods that maintain stability, which enables it to achieve consistent detection results that remain dependable across different time periods.

The major worth of this study is that it can operate in different fields, and its dynamism in real time makes the model capable of operating on different layers of the network and on different systems without necessarily having to undergo regular offline retraining. A replay of experience mechanism is employed to make sure that the agent learns based on a richer set of past experiences instead of basing his/her learning on recent observations alone, which minimizes the effect of short-lived traffic abnormalities. Moreover, reward shaping methods are used to steer the learning process toward balanced performance in detection between benign and attack classes. The study of traffic patterns demonstrates that temporal flow patterns, packet distribution details, and port usage patterns serve as crucial elements for identifying legitimate and harmful activities. The DRL agent uses these features to develop detection methods that recognize different types of DDoS attacks that range from high-rate attacks to stealth attacks and low-rate attacks.

The current research distinguishes itself from existing methods, which concentrate on performance enhancement within datasets or isolated environments, by demonstrating its capacity to function across diverse networks under real-world conditions.

Future research directions include incorporating additional minority attack classes, validating the framework using real-world operational traffic traces collected from heterogeneous network infrastructures, and exploring federated and multi-agent RL architectures for large-scale distributed environments. Future studies will further investigate continuous online adaptation under concept drift conditions, long-term deployment scenarios, convergence stability, training variance across multiple random seeds, and adversarial robustness under traffic manipulation attacks. Such investigations will provide deeper insight into the generalization capability, reliability, robustness, and long-term operational behavior of adaptive DRL-based cybersecurity frameworks deployed in dynamic and heterogeneous network environments.

Recommendations

Based on the experimental results, it is recommended that adaptive DRL-based security frameworks be deployed in heterogeneous network environments to enhance the early detection and proactive mitigation of DDoS attacks. The proposed DQN-based model demonstrates strong adaptability to evolving traffic patterns and multi-vector threats, making it more suitable than static ML/DL approaches for dynamic infrastructures such as IoT, cloud, edge, SDN, and 5G networks. For practical implementation, integration with real-time intrusion prevention systems and automated mitigation mechanisms is advised. Future work should focus on improving detection of minority attack classes and extending the framework toward scalable, multi-agent, or federated RL architectures for enhanced resilience and generalization.

Acknowledgment

The authors would like to acknowledge the creators and maintainers of the publicly available benchmark datasets, namely, CICDDoS2019, NSL-KDD, and UNSW-NB15, which facilitated comprehensive experimental evaluation and validation of the proposed DRL-based framework. The authors also acknowledge the computational resources utilized for conducting the experiments, including a system configured with Ubuntu 20.04, an NVIDIA RTX GPU, and an Intel i7 processor, which enabled efficient model training and performance evaluation. The implementation of the proposed framework was carried out using Python and relevant DL libraries, and the developed source code is available and can be provided upon reasonable request for research and verification purposes.

Ethical Statement

This study does not contain any studies with human or animal subjects performed by any of the authors.

Conflicts of Interest

The authors declare that they have no conflicts of interest to this work.

Data Availability Statement

The datasets used in this study, including CICDDoS2019, NSL-KDD, and UNSW-NB15, are publicly available through their respective repositories. The implementation details, preprocessing procedures, model configurations, and experimental settings are described within the manuscript to facilitate reproducibility. The source code and trained model configurations are available from the corresponding author upon reasonable request and will be considered for public repository release in future work to further enhance research transparency and reproducibility or can be accessed at https://www.kaggle.com/datasets/dhoogla/cicddos2019, https://www.kaggle.com/datasets/hassan06/nslkdd, and https://www.kaggle.com/datasets/mrwellsdavid/unsw-nb15.

Author Contribution Statement

Sujit Sutradhar: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Resources, Data curation, Writing – original draft, Writing – review & editing, Visualization, Project administration. **Joy Lal Sarkar:** Writing – review & editing, Visualization, Supervision, Project administration. **Abhijit Biswas:** Writing – review & editing, Supervision, Project administration.

References

[1] Roopesh, M., Nishat, N., Rasetti, S., & Rahaman, M. A. (2024). A review of machine learning and feature selection techniques for cybersecurity attack detection with a focus on DDOS attacks. *Academic Journal on Science, Technology, Engineering & Mathematics Education*, *4*(3), 178–194. https://doi.org/10.69593/ajsteme.v4i03.105

[2] Ma, C. (2021). Smart city and cyber-security; technologies used, leading challenges and future recommendations. *Energy Reports*, *7*, 7999–8012. https://doi.org/10.1016/j.egyr.2021.08.124

[3] Sullivan, S., Brighente, A., Kumar, S. A. P., & Conti, M. (2021). 5G security challenges and solutions: A review by OSI layers. *IEEE Access*, *9*, 116294–116314. https://doi.org/10.1109/ACCESS.2021.3105396

[4] Karnani, S., Agrawal, N., & Kumar, R. (2023). A comprehensive survey on low-rate and high-rate DDoS defense approaches in SDN: Taxonomy, research challenges, and opportunities. *Multimedia Tools and Applications*, *83*(12), 35253–35306. https://doi.org/10.1007/s11042-023-16781-0

[5] López, P. (2025). The national security framework as a cybersecurity reference for information cryptosystems. In C. Pastor Sempere (Ed.), *Governance and control of data and digital economy in the European single market: Legal framework for new digital assets, identities and data spaces* (pp. 125–144). Springer. https://doi.org/10.1007/978-3-031-74889-9_6

[6] Markopoulou, D., Papakonstantinou, V., & de Hert, P. (2019). The new EU cybersecurity framework: The NIS Directive, ENISA's role and the General Data Protection Regulation. *Computer Law & Security Review*, *35*(6), 105336. https://doi.org/10.1016/j.clsr.2019.06.007

[7] Bhayo, J., Shah, S. A., Hameed, S., Ahmed, A., Nasir, J., & Draheim, D. (2023). Towards a machine learning-based framework for DDOS attack detection in software-defined IoT (SD-IoT) networks. *Engineering Applications of Artificial Intelligence*, *123*, 106432. https://doi.org/10.1016/j.engappai.2023.106432

[8] Oladosu, S. A., Ike, C. C., Adepoju, P. A., Afolabi, A. I., Ige, A. B., & Amoo, O. O. (2021). Advancing cloud networking security models: Conceptualizing a unified framework for hybrid cloud and on-premise integrations. *Magna Scientia Advanced Research and Reviews*, *3*(1), 079–090. https://doi.org/10.30574/msarr.2021.3.1.0076

[9] Rejimol Robinson, R. R., & Thomas, C. (2021). Low rate multi-vector DDoS attack detection using information gain based feature selection. In *Computer Networks, Big Data and IoT: Proceedings of ICCBI 2020*, 685–696. https://doi.org/10.1007/978-981-16-0965-7_53

[10] Xiao, Y., Shi, G., Li, Y., Saad, W., & Poor, H. V. (2020). Toward self-learning edge intelligence in 6G. *IEEE Communications Magazine*, *58*(12), 34–40. https://doi.org/10.1109/MCOM.001.2000388

[11] Mishra, N., & Pandya, S. (2021). Internet of things applications, security challenges, attacks, intrusion detection, and future visions: A systematic review. *IEEE Access*, *9*, 59353–59377. https://doi.org/10.1109/ACCESS.2021.3073408

[12] Patsakis, C., Casino, F., & Katos, V. (2020). Encrypted and covert DNS queries for botnets: Challenges and countermeasures. *Computers & Security*, *88*, 101614. https://doi.org/10.1016/j.cose.2019.101614

[13] Al-Garadi, M. A., Mohamed, A., Al-Ali, A. K., Du, X., Ali, I., & Guizani, M. (2020). A survey of machine and deep learning methods for internet of things (IoT) security. *IEEE Communications Surveys & Tutorials*, *22*(3), 1646–1685. https://doi.org/10.1109/COMST.2020.2988293

[14] Shaukat, S. U., Khan, S., & Parkinson, S. (2025). A review on multi-step attack detection. *IEEE Access*, *13*, 161779–161805. https://doi.org/10.1109/ACCESS.2025.3607497

[15] Khraisat, A., & Alazab, A. (2021). A critical review of intrusion detection systems in the internet of things: Techniques, deployment strategy, validation strategy, attacks,

public datasets and challenges. *Cybersecurity*, *4*(1), 18. https://doi.org/10.1186/s42400-021-00077-7

[16] Kheddar, H., Dawoud, D. W., Awad, A. I., Himeur, Y., & Khan, M. K. (2025). Reinforcement-learning-based intrusion detection in communication networks: A review. *IEEE Communications Surveys & Tutorials*, *27*(4), 2420–2469. https://doi.org/10.1109/COMST.2024.3484491

[17] Khan, Q. W. (2024). Exploring Markov decision processes: A comprehensive survey of optimization applications and techniques. *Igmin Research*, *2*(7), 508–517. https://doi.org/10.61927/igmin210

[18] Nguyen, H. T., Nguyen, M. T., Do, H. T., Hua, H. T., & Nguyen, C. V. (2021). DRL-based intelligent resource allocation for diverse QoS in 5G and toward 6G vehicular networks: A comprehensive survey. *Wireless Communications and Mobile Computing*, *2021*(1), 5051328. https://doi.org/10.1155/2021/5051328

[19] Wu, Z., Li, W., Liu, L., & Yue, M. (2020). Low-rate DoS attacks, detection, defense, and challenges: A survey. *IEEE Access*, *8*, 43920–43943. https://doi.org/10.1109/ACCESS.2020.2976609

[20] Güllü, M., & Barişçi, N. (2026). Visual question answering for intelligent communication systems: A systematic review. *IEEE Access*, *14*, 11607–11630. https://doi.org/10.1109/ACCESS.2026.3654676

[21] Hossain, M. A. (2025). Deep Q-learning intrusion detection system (DQ-IDS): A novel reinforcement learning approach for adaptive and self-learning cybersecurity. *ICT Express*, *11*(5), 875–880. https://doi.org/10.1016/j.icte.2025.05.007

[22] Bakro, M., Kumar, R. R., Husain, M., Ashraf, Z., Ali, A., Yaqoob, S. I., . . . , & Parveen, N. (2024). Building a cloud-IDS by hybrid bio-inspired feature selection algorithms along with random forest model. *IEEE Access*, *12*, 8846–8874. https://doi.org/10.1109/ACCESS.2024.3353055

[23] Doğan, S. M., Koçak, A., & Alkan, M. (2025). Detection and mitigation of cyber-attacks in software defined networks using machine learning/deep learning: A systematic literature review, research challenges and future directions. *International Journal of Information Security*, *24*(5), 209. https://doi.org/10.1007/s10207-025-01114-z

[24] Manokaran, J., & Vairavel, G. (2024). DL-ADS: Improved grey wolf optimization enabled AE-LSTM technique for efficient network anomaly detection in internet of thing edge computing. *IEEE Access*, *12*, 75983–76002. https://doi.org/10.1109/ACCESS.2024.3405628

[25] Nomikos, N., Zoupanos, S., Charalambous, T., & Krikidis, I. (2022). A survey on reinforcement learning-aided caching in heterogeneous mobile edge networks. *IEEE Access*, *10*, 4380–4413. https://doi.org/10.1109/ACCESS.2022.3140719

[26] Hady, M. A., Hu, S., Pratama, M., Cao, Z., & Kowalczyk, R. (2025). Multi-agent reinforcement learning for resources allocation optimization: A survey. *Artificial Intelligence Review*, *58*(11), 354. https://doi.org/10.1007/s10462-025-11340-5

[27] Jiao, L., Shao, Y., Sun, L., Liu, F., Yang, S., Ma, W., . . . , & Guo, Y. (2024). Advanced deep learning models for 6G: Overview, opportunities, and challenges. *IEEE Access*, *12*, 133245–133314. https://doi.org/10.1109/ACCESS.2024.3418900

[28] Yuan, H., & Li, G. (2021). A survey of traffic prediction: From spatio-temporal data to intelligent transportation. *Data Science and Engineering*, *6*(1), 63–85. https://doi.org/10.1007/s41019-020-00151-z

[29] Sangoleye, F., Johnson, J., & Eleni Tsiropoulou, E. (2024). Intrusion detection in industrial control systems based on deep reinforcement learning. *IEEE Access*, *12*, 151444–151459. https://doi.org/10.1109/ACCESS.2024.3477415

How to Cite: Sutradhar, S., Sarkar, J. L., & Biswas, A. (2026). A Deep Reinforcement Learning–Based Adaptive Framework for Early DDoS Attack Detection and Prevention in Heterogeneous Networks. *Journal of Data Science and Intelligent Systems*. https://doi.org/10.47852/bonviewJDSIS62029543

Appendix

Abbreviation	Full Form
DRL	Deep Reinforcement Learning
DQN	Deep Q-Network
DDoS	Distributed Denial of Service
IDS	Intrusion Detection System
ML	Machine Learning
DL	Deep Learning
MDP	Markov Decision Process
SDN	Software Defined Networking
IoT	Internet of Things
MEC	Multi-access Edge Computing
CNN	Convolutional Neural Network
LSTM	Long Short-Term Memory
GRU	Gated Recurrent Unit
PPO	Proximal Policy Optimization
A3C	Asynchronous Advantage Actor–Critic
SAC	Soft Actor–Critic
SVM	Support Vector Machine
MLP	Multilayer Perceptron
QoS	Quality of Service
DNS	Domain Name System
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
UDP	User Datagram Protocol
ICMP	Internet Control Message Protocol
IP	Internet Protocol
FTP	File Transfer Protocol
SSH	Secure Shell
XSS	Cross-Site Scripting
RTX	Ray Tracing Texel eXtreme (NVIDIA RTX Series)
NIS	Network and Information Security