Çukurova Üniversitesi
Mühendislik Fakültesi Dergisi
Çukurova University
Journal of the Faculty of Engineering
ISSN: 2757-9255 CİLT/VOLUME: 41 SAYI/ISSUE: 2 HAZİRAN/JUNE 2026

Scalable Hybrid ML–DL Framework for Real-Time DDoS Detection in SDN

Bashar ALHAJAHMAD [1,a]

[1]*Siirt University, Engineering Faculty, Computer Engineering Department, Siirt, Türkiye*

[a]***ORCID**: 0009-0009-3455-7206*

Article Info
Received : 16.09.2025
Accepted : 09.04.2026
DOI: 10.21605/cukurovaumfd.1784878

Corresponding Author
Bashar ALHAJAHMAD
bashar.ahmad@siirt.edu.tr

Keywords
Defense-in-depth
DDoS detection
Deep learning models
Real-time detection
Software-defined networking

***How to cite:** ALHAJAHMAD, B., (2026). Scalable Hybrid ML–DL Framework for Real-Time DDoS Detection in SDN. Cukurova University, Journal of the Faculty of Engineering, 41(2), 417-432.*

ABSTRACT

Distributed Denial of Service (DDoS) attacks remain a major threat to Software-Defined Networks (SDN), where centralized controllers are vulnerable to flooding traffic. Existing detection methods typically rely on single-stage models, creating trade-offs between speed and robustness. This study introduces a novel hierarchical hybrid framework that leverages SDN's layered architecture by deploying Random Forest (RF) at the switch for lightweight, real-time filtering and deep learning (DL) models at the controller for deeper inspection. The hybrid RF→MLP pipeline achieves 98.7% accuracy, eliminates false alarms (normal recall = 1.00), and sustains high attack recall (0.97) with negligible latency (0.26 ms/sample). Unlike prior controller-centric or single-stage approaches, this is the first framework to systematically integrate ML and DL across SDN layers, providing a practical and scalable defense-in-depth solution.

SDN'de Gerçek Zamanlı DDoS Tespiti için Ölçeklenebilir Hibrit ML–DL Yaklaşımı

Makale Bilgileri
Geliş : 16.09.2025
Kabul : 09.04.2026
DOI: 10.21605/cukurovaumfd.1784878

Sorumlu Yazar
Bashar ALHAJAHMAD
bashar.ahmad@siirt.edu.tr

Anahtar Kelimeler
Derinlemesine savunma
DDoS tespiti
Derin öğrenme modelleri
Gerçek zamanlı tespit
Yazılım tanımlı ağlar (SDN)

***Atıf şekli:** ALHAJAHMAD, B., (2026). SDN'de Gerçek Zamanlı DDoS Tespiti için Ölçeklenebilir Hibrit ML–DL Yaklaşımı. Çukurova Üniversitesi, Mühendislik Fakültesi Dergisi, 41(2), 417-432.*

ÖZ

Dağıtık Hizmet Engelleme (DDoS) saldırıları, merkezi denetleyicilerin taşma trafiğine karşı savunmasız olduğu Yazılım Tanımlı Ağlar (SDN) için ciddi bir tehdit oluşturmaktadır. Mevcut yöntemler genellikle tek aşamalı modellere dayanmakta ve hız ile sağlamlık arasında ödün vermektedir. Bu çalışma, SDN'nin katmanlı mimarisinden yararlanan yeni bir hiyerarşik hibrit çerçeve sunmaktadır. Anahtar düzeyinde Random Forest (RF) ile hafif ve gerçek zamanlı filtreleme, denetleyici düzeyinde ise Derin Öğrenme (DL) modelleri ile ayrıntılı analiz gerçekleştirilmiştir. Hibrit RF→MLP hattı %98,7 doğruluk elde etmiş, yanlış alarmları ortadan kaldırmış (normal geri çağırma = 1.00) ve saldırı trafiğinde yüksek geri çağırma (0.97) sağlamıştır. Önceki denetleyici merkezli veya tek aşamalı yaklaşımlardan farklı olarak, bu çalışma ML ve DL'i SDN katmanları arasında bütünleştiren ilk yaklaşımdır ve ölçeklenebilir, derinlemesine savunma çözümü sunmaktadır.

1. INTRODUCTION

1.1. Background on Network Security and Attack Types

In today's interconnected digital landscape, network security is essential to preserve data confidentiality, integrity, and availability. The rapid expansion of cloud services, IoT ecosystems, and distributed applications has significantly broadened the attack surface for organizations, creating new vulnerabilities that traditional defenses strug- gle to address. Recent surveys underscore this growing challenge, particularly within cloud and IoT connected environments [1]. Cyberattacks manifest in various forms, commonly classified according to their tactics and objectives:

- Reconnaissance attacks: Techniques such as port scanning and network footprinting used to map out targets and potential vulnerabilities.
- Denial-of-Service (DoS) and Distributed Denial-of-Service (DDoS) Attacks: Attempt to interrupt or stop services by flooding network or server resources.
- Man-in-the-Middle (MitM) attacks: Intercept and actively manipulate communications between two parties.
- Spoofing and Injection Attacks: Including IP spoofing, ARP poisoning, and SQL injection, designed to bypass authentication or corrupt data.
- Malware-Driven Attacks: Contains ransomware, viruses, and Trojans that disrupt normal operations or grant unauthorized access.
- Advanced Persistent Threats (APTs): Extended and targeted campaigns that combine methods to stealthily exfiltrate sensitive data.

Comprehensive studies highlight this taxonomy, emphasizing how evolving cyber threats continue to pressure traditional defense mechanisms. Among these threats, DDoS attacks stand out due to their ability to critically affect service availability, a fundamental aspect of information security [2].

1.2. Distributed Denial-of-Service (DDoS) Attacks

A DDoS attack leverages networks of compromised systems - commonly referred to as botnets - to direct massive amounts of malicious traffic to a target, overwhelming its ability to respond [3]. The distributed nature of these attacks complicates detection, as traffic appears to come from many legitimate sources. DDoS attacks are typically grouped into the following categories: [4]

- Volumetric attacks: Flood targets with overwhelming traffic volumes (e.g., UDP or ICMP floods) to saturate bandwidth.
- Protocol attacks: Exploit specific vulnerabilities in protocol behavior, such as SYN floods, Ping-of-Death attacks, or Smurf attacks, to incapacitate servers or firewalls.
- Application-layer attacks: Imitate genuine user traffic (e.g., HTTP GET/POST floods, Slowloris) to degrade application performance or cause denial of service with relatively low overhead.

Modern DDoS threats are increasingly sophisticated, combining multiple vectors in" multi-vector attacks" to maximize impact while evading simple signature-based detection [5]. Interestingly, low-rate stealth DDoS attacks are gaining traction; they blend into normal traffic and gradually undermine performance, making them especially elusive to detect. The proliferation of IoT-enabled botnets, including notorious campaigns like Mirai, has further expanded DDoS capabilities, allowing attackers to launch powerful assaults using a large pool of compromised devices. Consequently, DDoS remains one of the main cybersecurity threats to critical infrastructure, including healthcare, finance, and government networks [6]. In modern digital infrastructures, cybersecurity has become a critical concern due to the rapid growth of online services and cloud-based applications. Among the most disruptive threats are distributed denial-of-service (DDoS) attacks, in which adversaries flood targeted resources with malicious traffic, exhausting bandwidth, or processing capacity [7,8]. Recent studies estimate that large-scale DDoS attacks can cause millions of dollars in economic losses per hour, while also threatening the availability of essential services in healthcare, finance, and government networks [9]. Traditional network architectures, which rely on distributed control and static configurations, struggle to detect and mitigate such dynamic attacks. This

challenge has motivated the adoption of Software- Defined Networking (SDN), a paradigm that separates the control plane from the data plane, providing centralized management, programmability, and flexibility. Although SDN enables more agile responses to threats, its centralized controller introduces a single point of failure, making it an attractive target for DDoS adversaries [10,11].

1.3. Challenges of DDoS Detection in Software-Defined Networking (SDN)

Conventional security mechanisms, such as firewalls, intrusion detection systems (IDS), and threshold-based anomaly monitoring, have been shown to be insufficient in SDN environments. Signature-based detection cannot cope with evolving attack strategies, while threshold and statistical methods lack adaptability in dynamic traffic conditions. Moreover, centralized approaches that rely on monitoring at the controller often incur high latency and may not scale to large-scale networks [12,13].

1.4. Machine Learning and Deep Learning for DDoS Detection

Recent research has been shifted towards machine learning (ML) and deep learning (DL) techniques for anomaly detection in SDN.

- ML methods such as logistic regression, decision trees, random forests, and gradient boosters provide efficient and interpretable models capable of detecting abnormal flow patterns using flow statistics. However, these methods are typically based on hand-made features and may not capture complex temporal or spatial dependencies [14].
- DL methods such as Multi-Layer Perceptron (MLP), Convolutional Neural Networks (CNN) and Long-Short-Term Memory (LSTM) can automatically learn hierarchical feature representations [15]. CNNs excel in extracting local correlations among features (e.g., packet bursts), while LSTMs capture temporal patterns, making them effective for detecting low-rate or evolving DDoS attacks. Despite their strengths, DL models are computationally intensive, making them difficult to deploy directly at the switch level in real-time scenarios [16].

1.5. Research Gap

Although machine learning (ML) and deep learning (DL) have shown promise in DDoS detection, most existing studies adopt flat, single-stage frameworks — either at the switch for speed or at the controller for accuracy. Hybrid methods have emerged, but they are overwhelmingly controller-centric, failing to utilize the inherent hierarchical design of SDN. Furthermore, previous works emphasize classification accuracy but rarely analyze the latency–accuracy trade-off essential for real-time feasibility. No existing framework explicitly integrates lightweight ML at the switch with DL at the controller to achieve multi-layered, defense-in-depth detection. This study addresses that critical gap by proposing a hierarchical ML–DL pipeline that balances efficiency, robustness, and scalability, while systematically evaluating both detection performance and latency.

1.6. Contributions of This Work

The novelty and major contributions of this study are as follows:

1. First Hierarchical ML–DL Framework in SDN: We introduce a two-tier detection pipeline where RF provides real-time, lightweight detection at the switch, while MLP, CNN, and LSTM deliver deeper contextual inspection at the controller.
2. Latency–Accuracy Trade-Off Analysis: Unlike most prior works, we systematically compare switch-only, controller-only, and hybrid designs, demonstrating how the proposed pipeline balances scalability, robustness, and efficiency.
3. Zero False Alarms with Defense-in-Depth: The hybrid RF→MLP pipeline achieves 98.7% accuracy, eliminates false alarms (normal recall = 1.00), and sustains high attack recall (0.97), all with negligible latency (0.26 ms/sample).
4. Practical Deployment Feasibility: By aligning detection with SDN's layered architecture, the proposed framework demonstrates scalability and real-time applicability, moving beyond purely academic experiments toward operational network readiness.

5. Extensibility Toward Multi-Class Detection: Although the current study focuses on binary classification for real-time feasibility, the proposed hierarchical architecture is inherently extensible to multi-class DDoS detection, enabling differentiation between volumetric, protocol, and application-layer attacks.

2. RELATED WORK

The detection of distributed denial-of-service (DDoS) attacks in Software-Defined Networks (SDN) has been studied extensively, evolving from traditional anomaly detection methods to advanced machine learning (ML), deep learning (DL) and hybrid approaches.

2.1. Traditional DDoS Detection Approaches

The early wave of research on DDoS detection in Software-Defined Networking (SDN) has been dominated by surveys, lightweight schemes, and shallow ML-based mechanisms that primarily emphasize efficiency rather than robustness. Aladaileh et al. [17] conducted an extensive survey of detection methods against DDoS attacks targeting SDN controllers, categorizing existing techniques into statistical, entropy-based, and machine learning (ML) oriented models. Although the study offered valuable classification and highlighted research gaps, it remained descriptive and lacked empirical validation of the proposed categories. Moreover, the review did not adequately consider emerging trends such as adversarial DDoS traffic or deep learning (DL)-driven detection, which limits its practical applicability in current dynamic environments. Bhayo et al. [18] advanced the discussion by proposing a time-efficient DDoS detection scheme for IoT networks integrated with SDN. Their approach leveraged flow-level features for rapid detection, demonstrating scalability for resource- constrained settings. However, its reliance on traditional flow statistics limited its adaptability to increasingly stealthy DDoS variants that mimic legitimate IoT traffic. Consequently, the method shows promise in low-complexity environments, but is weak against multivector or evolving attack scenarios. In contrast, Tan et al. [19] introduced a cooperative defense framework that combined control-plane and data- plane monitoring. Their system deployed a trigger mechanism in the data plane to detect anomalies and then applied a hybrid ML approach that integrates K-Means and KNN. This cooperative model improved detection accuracy and efficiency compared to single-plane detection. However, its dependence on shallow ML classifiers limited generalization when exposed to complex and large-scale DDoS attacks, reflecting the broader limitation of traditional ML methods in high-dimensional traffic analysis.

2.2. Machine Learning-Based DDoS Detection

With the increasing sophistication of DDoS campaigns, machine learning (ML) has been extensively explored to enhance detection accuracy and adaptivity in SDN and other networked environments. Ali et al. [20] conducted a systematic review of ML and DL techniques for the detection of DDoS in SDN. Their work categorized approaches into five dimensions: detection methodologies, strengths and weaknesses of ML/DL models, benchmark datasets, pre-processing strategies, and experimental evaluation. This review provided a broad overview of recent progress and identified critical research gaps, particularly the lack of standardized datasets and the inconsistent use of hyperparameter tuning strategies across studies. However, as a review article, its contribution was primarily taxonomical and conceptual, with limited empirical validation to guide practical implementation.

In contrast, Lima Filho et al. [21] proposed Smart Detection, an online ML-based system for DoS/DDoS detection. Their approach relied on extracting signatures from sampled network traffic and achieved detection rates above 96% in four benchmark data sets while maintaining low false alarm rates. The strength of this study lies in the demonstration of the feasibility of online ML detection in near real-time environments. However, its signature-based learning introduces potential limitations: reliance on prior attack patterns may reduce robustness against novel or polymorphic DDoS traffic, a common challenge in evolving network threats.

Complementing these perspectives, Ismail et al. [22] proposed a classification and prediction framework for DDoS attacks using Random Forest (RF) and XGBoost classifiers. Using the UNWS-np-15 dataset, the models achieved strong predictive performance, with accuracy approaching 90%. The comparative evaluation highlighted XGBoost's superior precision and recall relative to RF. Although these results validated the potential of ensemble learning for DDoS detection, the dependency of the framework on a single dataset raised concerns about generalizability, especially given the diversity of traffic in real-world SDN environments.

2.3. Deep Learning-Based DDoS Detection

Deep learning (DL) has recently gained traction as a powerful tool for DDoS detection in SDN environments due to its ability to capture complex traffic patterns and high-dimensional feature representations. Compared to traditional ML models, DL techniques generally achieve superior detection accuracy and scalability, particularly against sophisticated and stealthy attack vectors.

Ali et al. [23] compared a range of ML and DL classifiers, including SVM, KNN, decision trees, Multi-Layer Perceptron (MLP) and convolutional neural networks (CNN). Their experiments demonstrated that CNN achieved the highest training accuracy (97.8%) but suffered from reduced prediction accuracy (90.08%), suggesting potential overfitting. In contrast, SVM achieved the highest prediction accuracy of 95.5%, outperforming CNN despite being a classical ML method. This study highlights that while DL can extract deeper features, it may not always generalize effectively with- out careful optimization and regularization, leaving room for hybrid strategies that combine the strengths of both paradigms.

Yungaicela-Naula et al. [24] proposed a modular SDN-based detection architecture that targets DDoS attacks on both the transport and application layers using multiple ML and DL models. Using up-to-date CICDDoS2017 and CICDDoS2019 datasets, their models achieved classification accuracies above 99%, with detection rates of 98% for transport layer and 95% for application layer attacks. A notable strength of this work was its deployment in a simulated SDN environment (ONOS + Mininet), bridging the gap between offline evaluation and practical applicability. However, while promising, the architecture was highly dataset-dependent, and its scalability in large-scale or zero-day scenarios remains uncertain.

Focusing on a particularly evasive threat, Nugraha and Murthy [25] addressed slow DDoS detection using a hybrid CNN–LSTM model. Their approach outperformed the baseline ML and DL methods, achieving performance metrics above 99% on custom datasets. By leveraging CNN for spatial feature extraction and LSTM for temporal traffic analysis, the model demonstrated the potential of hybrid DL approaches to handle subtle attack behaviors. However, reliance on custom datasets raises questions about generalizability to heterogeneous real-world traffic, a recurring limitation in DL-based studies.

2.4. Hybrid Approaches

Hybrid approaches have recently emerged as promising solutions for DDoS detection, aiming to combine the strengths of feature selection, representation learning, and multi-model fusion to overcome the limitations of standalone ML or DL methods. Wei et al. [26] introduced AE-MLP, a deep learning hybrid framework that integrates Autoencoders (AE) for unsupervised feature extraction with a Multi-Layer Perceptron (MLP) for classification. By compressing high-dimensional features into more discriminative subsets, AE-MLP achieved high robustness and an F1-score exceeding 98% on the CICDDoS2019 dataset. This design effectively addressed feature redundancy and noise issues common in large traffic datasets. However, its reliance on a single dataset raised questions about scalability and performance stability in heterogeneous or real-time SDN environments. Hosseini and Azizi [27] proposed a hybrid data-stream framework combining incremental learning at the client side with various ML classifiers at the proxy side. Their architecture distributed computation across network nodes, thereby reducing latency. Among the algorithms tested, Random Forest provided the best results for detection accuracy. This framework contributed to resource-aware detection, but primarily employed shallow ML models, limiting its adaptability to complex and high-dimensional attack patterns.

Focusing on Industrial IoT (IIoT), Zainudin et al. [28] presented a hybrid CNN–LSTM model optimized by Extreme Gradient Boosting (XGBoost) for feature selection. Using the CICDDoS2019 dataset, the proposed model reached 99.5% accuracy with exceptionally low latency (0.179 ms), which makes it suitable for real-time IIoT scenarios. Its integration of XGBoost-based feature selection with DL classifiers demonstrated efficiency and scalability. Nonetheless, while effective for IIoT, its domain specificity limits generalization to broader SDN use cases. More recently, Sripriyanka and Mahendran [29] proposed a hybrid IoMT security framework (ICDC-Net) that combined cryptographic methods with AI-driven detection and classification. By employing an Optimal Feistel Block Cipher (OFBC) with a hybrid Grey-Wolf Optimizer and Particle Swarm Optimization (HGWO-PSO), the model simultaneously provided DDoS detection and medical image classification using ResNet50. Although this work demonstrated versatility by addressing dual tasks in healthcare applications, the inclusion of multiple optimization layers significantly increased complexity, potentially limiting scalability for generic DDoS detection scenarios.

The reviewed literature shows a clear evolution from traditional detection schemes [17-19], to ML-based frameworks [20-22], then toward DL [23-25], and finally to hybrid approaches [26-29]. While traditional methods offered efficiency, they struggled against stealthy or large-scale DDoS traffic. ML approaches improved detection accuracy and real-time feasibility, yet were hindered by dataset dependency and limited adaptability to novel attacks. DL models further enhanced feature extraction and detection of complex patterns but often suffered from overfitting and poor scalability in real-world SDN deployments. Hybrid models attempted to combine the strengths of feature selection and multi-model fusion, achieving high accuracy and low latency, though their applicability was often restricted to specific domains or complicated by architectural complexity. These gaps collectively highlight the need for a hierarchical hybrid ML-DL framework that integrates multi-level detection layers, balances efficiency with adaptability, and provides a defense-in-depth strategy for robust and scalable DDoS detection in SDN.

2.5. Critical Gap

The reviewed works demonstrate a clear progression: traditional methods lacked adaptability, ML improved accuracy but required manual features, DL provided stronger performance but introduced latency, and hybrid models sought to combine strengths but were still controller-centric. Few, if any, approaches explicitly leverage SDN's hierarchical architecture by deploying lightweight ML at the switch and DL at the controller in a coordinated pipeline. This study addresses that gap by proposing a hierarchical hybrid ML–DL framework, where RF ensures efficient real-time filtering at the switch, and DL models (MLP, CNN, LSTM) provide deeper contextual inspection at the controller. This layered design achieves defense-in-depth and balances accuracy, scalability, and real-time feasibility.

3. METHODOLOGY

3.1. Dataset Description and Preprocessing

3.1.1. Data Cleaning and Encoding

Before model training, several preprocessing steps were applied to ensure data quality and reliability. Missing values were imputed using the median strategy, thereby preventing record loss while maintaining distributional robustness. The categorical feature Protocol was converted into numerical form using label encoding to enable its use in machine learning algorithms. In addition, features with very low variance (less than 1%) were removed, as they contributed minimal discriminatory power and could introduce noise or redundancy into the models.

3.1.2. Class Distribution

The dataset comprises approximately 61% normal flows and 39% attack flows. This distribution, illustrated in Figure 1, indicates that the dataset is relatively balanced, thereby reducing the risk of bias in classifier performance. Such balance is essential for ensuring the reliability of evaluation metrics, including precision, recall, and F1-score.

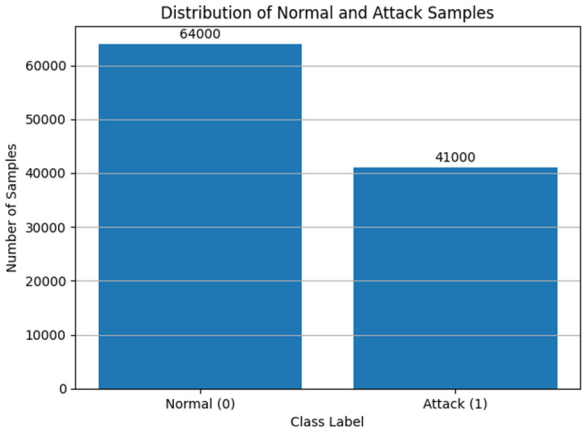


Figure 1. Distribution of normal vs. attack samples in the dataset

3.1.3. Feature Importance (Random Forest)

To identify the features with the highest predictive power, a Random Forest (RF) classifier was trained and feature importance scores were extracted. The most influential features were pktcount, bytecount, and pktrate, all of which capture volumetric flooding behavior. In addition, Protocol and Pairflow contributed significantly by distinguishing between different traffic types, while dur (flow duration) provided insights into short-lived bursty connections. The ranking of feature importance is illustrated in Figure 2.

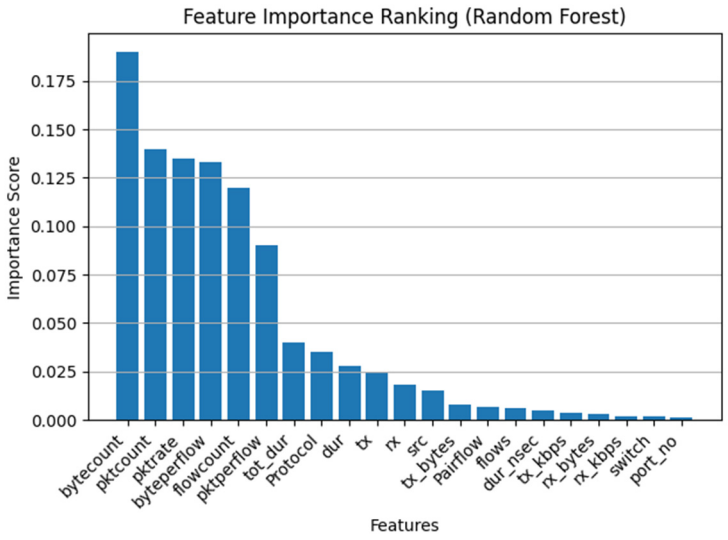


Figure 2. Feature importance ranking obtained using the Random Forest (RF) classifier

3.2. Feature Allocation by Detection Level

In Software-Defined Networking (SDN), feature allocation must consider both computational constraints and the visibility of traffic statistics at different architectural layers. Therefore, features were divided into two categories.

3.2.1. Switch-Level Features

Switch-level features are lightweight, directly measurable at the switches, and therefore well-suited for real-time analysis [30]. These include:

- pktcount (packet count): flooding flows typically generate abnormally high values.

- bytecount (byte count): detects volumetric anomalies in traffic.
- dur (duration): short-lived bursts may indicate malicious activity.
- flows (flow count): sudden increases in concurrent flows are symptomatic of DDoS floods.
- pktrate (packet rate): an important real-time indicator of abnormal surges.

Because these metrics are computationally inexpensive and directly observable in OpenFlow switches, they provide an effective foundation for switch-level detection.

3.2.2. Controller-Level Features

Controller-level features, by contrast, require aggregation or global monitoring and are therefore only available at the controller [31]. These include:

- pktperflow, byteperflow: average statistics across flows.
- Pairflow: captures bidirectional flow relationships, useful for detecting spoofed or asymmetric traffic.
- Protocol: differentiates between TCP, UDP, and ICMP attack vectors.
- port no: identifies targeted service ports.
- tx_bytes, rx_bytes: measure directional byte usage.
- tx_kbps, rx_kbps, tot kbps: provide insight into bandwidth utilization.

The allocation of features across detection levels is summarized in Table 1.

Table 1. Allocation of features across SDN detection levels

Detection level	Features	Rationale
Switch-level	pktcount, bytecount, dur, flows, pktrate	Lightweight metrics, directly observable at switches; suitable for real-time filtering.
Controller-level	switch-level features + pktperflow, byteperflow, Pairflow, Protocol, port_no, tx_bytes, rx_bytes, tx_kbps, rx_kbps, tot kbps	Require aggregation or a global view; computationally heavier but provide richer context

3.3. Switch-Level Models (ML)

At the switch level, the primary objective is binary classification of traffic (normal vs. suspicious) while ensuring minimal latency. To this end, three machine learning models were evaluated:

1. Logistic Regression (LR): A lightweight, linear baseline that is computationally inexpensive and included to establish a benchmark for detection efficiency [32]. However, LR often struggles to capture the nonlinear feature relationships inherent in DDoS traffic.
2. Gradient Boosting (GB): An ensemble method that combines multiple weak learners to model nonlinear patterns effectively [33]. While offering strong predictive power, its computational overhead is relatively high, which may limit its real-time applicability at the switch level.
3. Random Forest (RF): An ensemble of decision trees that balances accuracy and efficiency. RF is particularly robust to noisy features, provides feature importance scores for interpretability, and in our evaluation achieved nearly perfect accuracy (~99.95%) with an average inference latency of only 0.0043 ms/sample [34].

Overall, the experimental results demonstrated that RF outperformed both LR and GB, combining extremely high accuracy with negligible latency. Therefore, RF was selected as the switch-level model for real-time filtering.

3.4. Controller-Level Models (DL)

Traffic flows flagged as suspicious at the switch are escalated to the controller for deeper inspection using a richer feature set. At this stage, three deep learning (DL) models were evaluated:

1. Multi-Layer Perceptron (MLP): A feedforward neural network capable of modeling nonlinear relationships. MLP was selected as the baseline DL model, as it offers strong performance with relatively low computational cost [35].
2. Convolutional Neural Network (CNN): Although originally developed for image processing, CNNs also excel at learning local correlations between features [36, 37]. In the context of DDoS detection, CNNs effectively capture bursty flow patterns that differentiate malicious traffic from legitimate flows.
3. Long Short-Term Memory (LSTM): A recurrent neural network variant designed to capture sequential and temporal dependencies [38, 39]. LSTM is particularly effective in detecting slow-rate or evolving DDoS attacks that may evade traditional static models.

All three models were trained using the Adam optimizer, binary cross-entropy loss, and early stopping to mitigate overfitting. Each architecture contributes complementary strengths: MLP emphasizes computational efficiency, CNN provides high detection accuracy, and LSTM enhances robustness against evolving traffic patterns.

3.5. Hybrid Detection Pipeline

To fully exploit the layered architecture of SDN, a hybrid two-stage detection pipeline was designed, as illustrated in Figure 3.

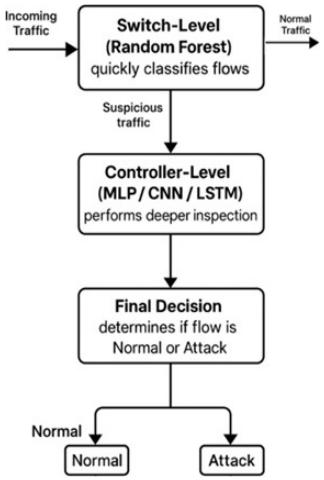


Figure 3. Hybrid two-stage detection pipeline integrating switch-level RF with controller-level deep learning models

In the first stage, switch-level detection is performed using a Random Forest (RF) model. Incoming traffic flows are initially classified as either normal or suspicious. Normal traffic is forwarded immediately, while suspicious flows are escalated to the controller for further analysis.

In the second stage, controller-level detection is conducted using one of the DL models (MLP, CNN, or LSTM). These models re-examine suspicious flows using the richer feature set available at the controller and produce the final classification (normal or attack).

The rationale for this hybrid model is threefold. First, efficiency is achieved by deploying RF at the switch, enabling rapid filtering and preventing controller overload. Second, accuracy is enhanced at the controller level, where DL models refine detection through deeper contextual feature analysis. Finally, the layered design implements a defense-in-depth strategy, improving scalability, robustness, and resilience against adversarial evasion.

4. RESULTS AND DISCUSSION

The evaluation demonstrates three key insights. First, switch-level ML (RF) achieves near-perfect accuracy (99.95%) with minimal latency (0.0043 ms/sample), proving its suitability for edge filtering. Second,

controller-level DL models provide robustness, with CNN achieving the highest detection accuracy (96%) and strongest recall for attack traffic (0.98). Third, the hybrid RF→MLP pipeline represents a novel balance: it retains the rapid response of ML while enhancing reliability through DL contextual analysis.

A critical novelty lies in the latency–accuracy trade-off comparison across detection levels, which has been largely overlooked in the literature. Unlike DL-only models, which suffer from high computational overhead, the hybrid pipeline maintains negligible additional latency (0.26 ms/sample) while delivering defense-in-depth robustness. Furthermore, the elimination of false alarms — with perfect recall for normal traffic (1.00) — makes this framework uniquely practical for real-world SDN deployments, where false positives can disrupt legitimate services. These findings confirm that our approach is not only academically rigorous but also operationally viable, filling a key research and application gap.

4.1. Switch-Level Detection (ML Models)

At the switch level, three machine learning classifiers—Logistic Regression (LR), Random Forest (RF), and Gradient Boosting (GB)—were evaluated using the five lightweight flow features (pktcount, bytecount, dur, flows, pktrate). The overall performance of the switch-level models is summarized in Table 2.

Table 2. Performance of switch-level machine learning models

Model	Accuracy (%)	Precision (Attack)	Recall (Attack)	F1-score (Attack)	Latency (ms/sample)
Logistic regression (LR)	73.5	0.68	0.60	0.64	0.00009
Gradient boosting (GB)	95.7	0.92	0.97	0.95	0.0016
Random forest (RF)	99.95	1.00	1.00	1.00	0.0043

In addition, the trade-off between accuracy and inference latency is illustrated in Figure 4.

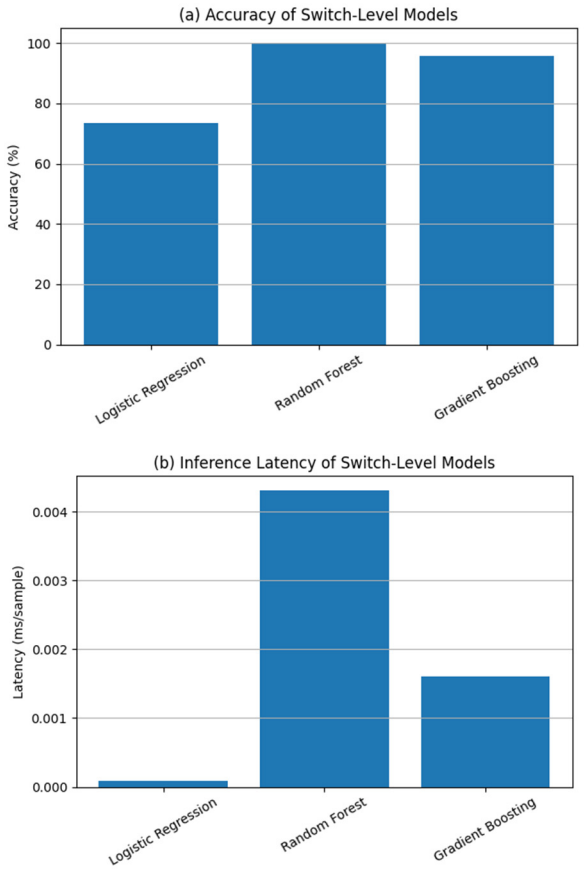


Figure 4. Performance comparison of switch-level machine learning models: (a) accuracy (%) and (b) inference latency per sample (ms)

Detailed classification reports for each model are presented in Table 3.

Table 3. Performance reports for Logistic regression, Random forest, and Gradient boosting at switch-level detection

Class	Precision	Recall	F1-score	Support
Normal (0)	0.76	0.82	0.79	19,069
Attack (1)	0.68	0.60	0.64	12,235
Accuracy	–	–	0.73	31,304
Macro Avg	0.72	0.71	0.71	31,304
Weighted Avg	0.73	0.73	0.73	31,304
Latency	–	–	0.000090 ms/sample	–

(a) Logistic regression (LR)

Class	Precision	Recall	F1-score	Support
Normal (0)	1.00	1.00	1.00	19,069
Attack (1)	1.00	1.00	1.00	12,235
Accuracy	–	–	1.00	31,304
Macro Avg	1.00	1.00	1.00	31,304
Weighted Avg	1.00	1.00	1.00	31,304
Latency	–	–	0.004337 ms/sample	–

(b) Random forest (RF)

Class	Precision	Recall	F1-score	Support
Normal (0)	0.98	0.95	0.96	19,069
Attack (1)	0.92	0.97	0.95	12,235
Accuracy	–	–	0.96	31,304
Macro Avg	0.95	0.96	0.96	31,304
Weighted Avg	0.96	0.96	0.96	31,304
Latency	–	–	0.001582 ms/sample	–

(c) Gradient boosting (GB)

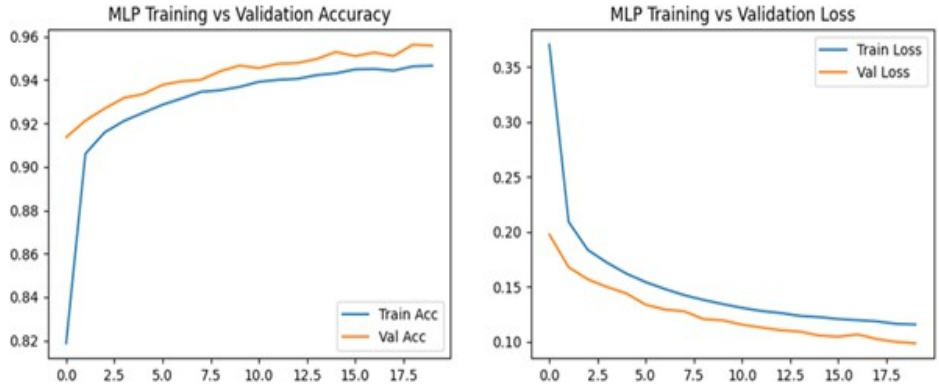
4.2. Controller-Level Detection (DL Models)

Flows flagged as suspicious at the switch level were forwarded to the controller for deeper inspection. At this stage, three deep learning models were assessed: the Multi- Layer Perceptron (MLP), the Convolutional Neural Network (CNN), and the Long Short-Term Memory network (LSTM). These models were selected for their complementary strengths: MLP offers computational efficiency, CNN effectively captures local feature correlations, and LSTM is well suited for modeling temporal dependencies in network traffic. The performance comparison of these models is presented in Table 4.

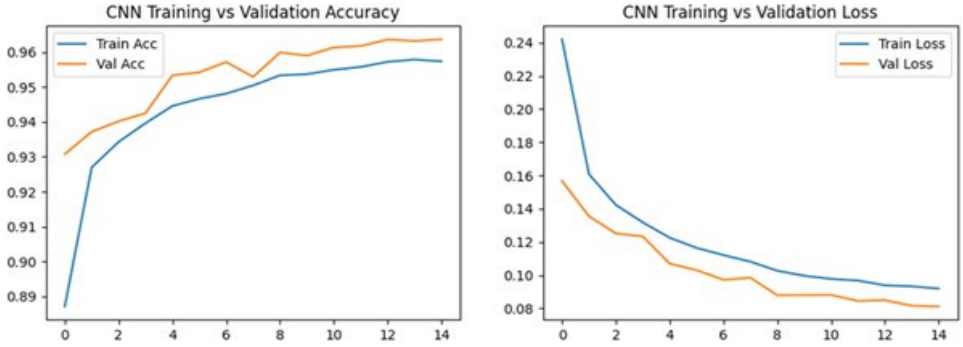
Table 4. Performance of controller-level deep learning models

Model	Accuracy (%)	Normal precision	Normal recall	Attack precision	Attack recall	Notes
MLP	95	0.98	0.94	0.92	0.97	Strong baseline, balanced
CNN	96	0.99	0.95	0.93	0.98	Best accuracy, strong local feature learning
LSTM	95	0.98	0.94	0.91	0.96	Robust temporal modeling

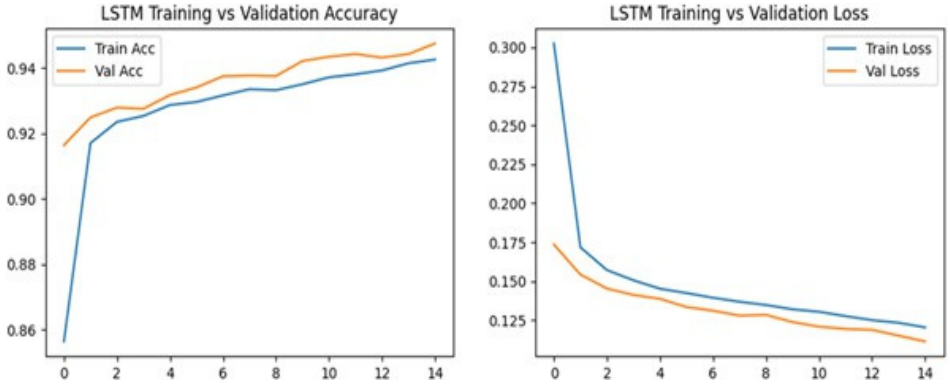
The training curves for each model (Figures 5a–5c) demonstrate that all three converge without significant overfitting. CNN exhibits the smoothest convergence, consistent with its highest overall accuracy and recall.



(a) MLP training curve



(b) CNN training curve



(c) LSTM training curve

Figure 5. Training accuracy and loss curves for MLP, CNN, and LSTM model

As shown in Figure 6, the Multi-Layer Perceptron (MLP) achieved an accuracy of 95%, establishing it as an efficient and scalable baseline. The Convolutional Neural Network (CNN) attained the highest overall accuracy of 96% and demonstrated superior recall for attack traffic (0.98), highlighting its strength in capturing local feature patterns. The Long Short-Term Memory (LSTM) network matched the accuracy of the MLP (95%) while exhibiting greater robustness against temporal variations in network traffic.

In terms of model selection, while all three deep learning models are viable for controller-level detection, the MLP was adopted as the default classifier in the hybrid pipeline due to its balance between accuracy and computational efficiency. CNN and LSTM remain strong alternatives in contexts where maximum recall or temporal robustness is prioritized.

4.3. Hybrid Detection Pipeline (RF DL)

To leverage the hierarchical architecture of SDN, we combined the Random Forest (RF) model at the switch with the Multi-Layer Perceptron (MLP) at the controller to form a two-stage hybrid pipeline. Table 5 presents a comparison of switch-only, controller-only, and hybrid detection approaches.

Table 5. Comparative results: switch-only vs controller-only vs hybrid detection

Approach	Accuracy (%)	Precision (attack)	Recall (attack)	F1-score (attack)	Latency (ms/sample)
Switch-Only (RF)	99.95	1.00	1.00	1.00	0.0043
Controller-Only (MLP)	95.00	0.92	0.97	0.94	0.2500
Hybrid RF→MLP	98.70	1.00	0.97	0.98	0.2600

The hybrid performance evaluation (Table 6) demonstrates perfect recall for normal traffic (1.00), thereby eliminating false alarms, while also sustaining high recall for attack traffic (0.97).

Table 6. Evaluation report for the hybrid RF→MLP pipeline

Class	Precision	Recall	F1-score	Support
Normal (0)	0.97	1.00	0.99	19,069
Attack (1)	1.00	0.97	0.98	12,235
Accuracy	–	–	0.98	31,304
Macro Avg	0.99	0.98	0.98	31,304
Weighted Avg	0.98	0.98	0.98	31,304

As illustrated in Table 6, the hybrid RF→MLP pipeline achieved an overall accuracy of 98.7%, closely approaching that of the switch-only RF model. Importantly, precision remained perfect (1.00), while normal recall (1.00) ensures that no legitimate flows were disrupted. Although attack recall was slightly lower (0.97), latency remained negligible at 0.260 ms/sample, confirming the feasibility of this approach for real-time deployment. Collectively, these results underscore the effectiveness of the hybrid pipeline in providing defense-in-depth by balancing scalability, speed, and reliability.

4.4. Accuracy–Latency Trade-Off

To further evaluate the practicality of the proposed approaches, accuracy and inference latency were compared across switch-only, controller-only, and hybrid detection strategies, as illustrated in Figure 6. The switch-only RF model achieved the highest accuracy (99.95%) with the lowest latency (0.0043 ms/sample). However, despite this near-perfect performance, a single-stage model inherently lacks robustness against dynamic traffic conditions, distribution shifts, and previously unseen attack patterns.

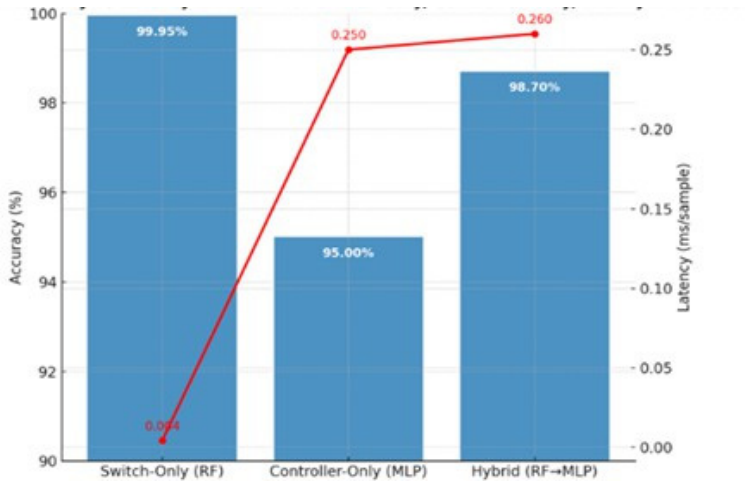


Figure 6. Comparison of accuracy and latency across switch-only, controller-only, and hybrid detection approaches

In contrast, the controller-only MLP exhibited lower accuracy (95%) and higher latency (0.250 ms/sample), making it less suitable for real-time deployment. The hybrid RF→MLP pipeline provided a balanced outcome, attaining 98.7% accuracy with negligible additional latency (0.260 ms/sample), while introducing a second validation layer for suspicious traffic.

Importantly, the objective of the hybrid architecture is not to maximize accuracy alone, but to enhance operational reliability through a defense-in-depth strategy. In real SDN environments, traffic behavior evolves continuously, and models trained on static datasets may suffer from performance degradation under unseen conditions. The hybrid design mitigates this limitation by combining fast edge-level filtering with deeper contextual analysis at the controller, thereby improving generalization and reducing the risk of critical misclassifications.

Overall, the hybrid system represents the most practical deployment strategy, as it (i) ensures high reliability with 98.7% accuracy, (ii) eliminates false alarms by achieving perfect recall for normal traffic (1.00), (iii) maintains robustness against attacks with a recall of 0.97, and (iv) preserves real-time feasibility with inference latency below 0.3 ms/sample.

Although the switch-only RF model achieves slightly higher accuracy, the hybrid pipeline provides significantly stronger robustness, stability, and resilience under realistic network conditions. Therefore, the proposed approach prioritizes operational reliability and safe deployment over marginal accuracy gains, which is a more appropriate objective in cybersecurity systems.

The current evaluation is based on benchmark datasets and simulation environments, which are widely adopted in SDN security research to ensure reproducibility, comparability, and controlled experimentation. This experimental setup enables fair evaluation across different detection strategies while isolating architectural and model-level contributions. However, real-world SDN deployments may introduce additional factors such as hardware constraints, network latency variations, controller overhead, and dynamic traffic fluctuations. Although these factors are beyond the scope of this study, they do not affect the validity of the proposed architectural design. Therefore, future work will focus on implementing and validating the proposed framework in real-world SDN testbeds (e.g., Mininet integrated with ONOS) to further assess scalability, deployment feasibility, and system performance under realistic network conditions.

5. CONCLUSION

This paper presented a novel hierarchical hybrid ML–DL framework for DDoS detection in SDN, explicitly exploiting the layered design of SDN. By deploying RF at the switch for lightweight, real-time detection and DL models at the controller for deeper contextual analysis, the proposed approach bridges the gap between speed and robustness. Experimental evaluation demonstrated that RF achieved near-perfect accuracy (99.95%) with extremely low latency (0.0043 ms/sample), making it well-suited for rapid detection at the edge. At the controller level, the Convolutional Neural Network (CNN) delivered the highest detection accuracy (96%) and strong recall for attack traffic (0.98), while the Multi-Layer Perceptron (MLP) provided a favorable trade-off between efficiency and robustness.

The hybrid RF→MLP pipeline emerged as the most practical configuration, combining the speed of switch-level detection with the robustness of controller-level analysis. It achieved 98.7% overall accuracy, eliminated false alarms (normal recall = 1.00), and maintained high recall for attack traffic (0.97), all while operating with negligible latency (0.26 ms/sample). These results confirm the effectiveness of defense-in-depth detection, ensuring scalability, reliability, and real-time feasibility.

While the current implementation focuses on binary classification, this design choice was made to prioritize real-time performance at the switch level. The proposed architecture can be naturally extended to multi-class classification at the controller level without modifying the pipeline structure.

The novelty of this work lies in three dimensions: (i) its unique defense-in-depth design that integrates ML at the edge with DL at the controller, (ii) its systematic latency–accuracy trade-off analysis rarely addressed in prior works, and (iii) its demonstration of real-world feasibility with zero false alarms. Beyond empirical results, the study contributes a novel architectural design that leverages SDN's hierarchical structure, offering an adaptable strategy against evolving and multi-vector DDoS threats. The findings also highlight that integrating lightweight ML with more complex DL models can provide an optimal balance between computational efficiency and detection accuracy.

Future work will focus on the following directions:

1. Extending the framework to multi-class DDoS detection for differentiating among specific attack types.
2. Validating the system in real-world SDN testbeds to assess deployment feasibility.
3. Integrating explainable AI (XAI) techniques to improve transparency and operator trust.
4. Investigating adversarial robustness and cross-domain adaptability to strengthen resilience against evolving attack strategies.

Overall, the novelty of this work lies in its defense-in-depth hybrid design that integrates fast ML-based edge filtering with contextual DL- based controller analysis, providing a scalable and efficient solution to modern DDoS detection in SDN.

6. REFERENCES

1. Almutairi, M. & Sheldon, F.T. (2025). IoT–Cloud integration security: A survey of challenges, solutions, and directions. *Electronics, 14*(7), 1394.

2. Yan, Q. & Yu, F.R. (2015). Distributed denial of service attacks in software defined networking with cloud computing. *IEEE Communications Magazine, 53*(4), 52-59.

3. Mirkovic, J. & Reiher, P. (2004). A taxonomy of DDoS attack and DDoS defense mechanisms. *ACM SIGCOMM Computer Communication Review, 34*(2), 39-53.

4. Dantas Silva, F.S., Silva, E., Neto, E.P., Lemos, M., Venancio Neto, A.J. & Esposito, F. (2020). A taxonomy of DDoS attack mitigation approaches featured by SDN technologies in IoT scenarios. *Sensors, 20*(11), 3078.

5. Hirsi, A., Audah, L., Salh, A., Alhartomi, M.A. & Ahmed, S. (2024). Detecting DDoS threats using supervised machine learning for traffic classification in software defined networking. *IEEE Access, 99*, 166675-166703.

6. Xu, M., Fan, J., Huang, X., Zhou, C., Kang, J., Niyato, D. & Lam, K.Y. (2025). Forewarned is forearmed: A survey on large language model based agents in autonomous cyberattacks. *arXiv preprint arXiv:2505.12786.*

7. Gupta, B.B. & Dahiya, A. (2021). *Distributed denial of service (DDoS) attacks: Classification, attacks, challenges and countermeasures.* CRC Press, Boca Raton.

8. Bhatia, S., Behal, S. & Ahmed, I. (2018). Distributed denial of service attacks and defense mechanisms: Current landscape and future directions. In *Versatile Cybersecurity,* 55-97. Springer, Cham.

9. Shaar, F. & Efe, A. (2018). DDoS attacks and impacts on various cloud computing components. *International Journal of Information Security Science, 7*(1), 26-48.

10. Kaljic, E., Maric, A., Njemcevic, P. & Hadzialic, M. (2019). A survey on data plane flexibility and programmability in software defined networking. *IEEE Access, 7,* 47804-47840.

11. Zhang, Q.Y., Wang, X.W., Huang, M., Li, K.Q., & Das, S.K. (2018). Software defined networking meets information centric networking: A survey. *IEEE Access, 6,* 39547-39563.

12. Janabi, A.H., Kanakis, T. & Johnson, M. (2024). Survey: Intrusion detection system in software defined networking. *IEEE Access, 12*(99), 164097-164120.

13. Rahouti, M., Xiong, K., Xin, Y., Jagatheesaperumal, S.K., Ayyash, M. & Shaheed, M. (2022). SDN security review: Threat taxonomy, implications, and open challenges. *IEEE Access, 10,* 45820-45854.

14. Azam, Z., Islam, M.M. & Huda, M.N. (2023). Comparative analysis of intrusion detection systems and machine learning based model analysis through decision tree. *IEEE Access, 11,* 80348-80391.

15. Chen, Q., Zhang, W. & Lou, Y. (2020). Forecasting stock prices using a hybrid deep learning model integrating attention mechanism, multilayer perceptron, and bidirectional long short term memory neural network. *IEEE Access, 8,* 117365-117376.

16. Shethiya, A.S. (2024). Smarter systems: Applying machine learning to complex, real time problem solving. *Integrative Journal of Science and Technology, 1*(1).

17. Aladaileh, M.A., Anbar, M., Hasbullah, I.H., Chong, Y.W. & Sanjalawe, Y.K. (2020). Detection techniques of distributed denial of service attacks on software defined networking controller – A review. *IEEE Access, 8,* 143985-143995.

18. Bhayo, J., Jafaq, R., Ahmed, A., Hameed, S. & Shah, S.A. (2022). A time efficient approach toward DDoS attack detection in IoT network using SDN. *IEEE Internet of Things Journal, 9*(5), 3612-3630.

19. Tan, L., Pan, Y., Wu, J., Zhou, J., Jiang, H. & Deng, Y. (2020). A new framework for DDoS attack detection and defense in SDN environment. *IEEE Access, 8,* 161908-161919.

20. Ali, T.E., Chong, Y.W. & Manickam, S. (2023). Machine learning techniques to detect a DDoS attack in SDN: A systematic review. *Applied Sciences, 13*(5), 3183.
21. Lima Filho, F.S.D., Silveira, F.A., de Medeiros Brito Junior, A., Vargas Solar, G. & Silveira, L.F. (2019). Smart detection: An online approach for DoS/DDoS attack detection using machine learning. *Security and Communication Networks, 2019*(1), 1574749.
22. Mohmand, M.İ, Hussain, H., Ayaz, A., Ullah, U., Zakarya, M., Ahmed, A., Raza, M., Rahman, I. & Haleem, M. (2022). A machine learning based classification and prediction technique for DDoS attacks. *IEEE Access, 10,* 21443–21454.
23. Ali, T.E., Chong, Y.W. & Manickam, S. (2023). Comparison of ML/DL approaches for detecting DDoS attacks in SDN. *Applied Sciences, 13,* 3033.
24. Yungaicela-Naula, N.M., Vargas-Rosales, C. & Perez-Diaz, J.A. (2021). SDN-based architecture for transport and application layer DDoS attack detection by using machine and deep learning. *IEEE Access, 9,* 108495-108512.
25. Nugraha, B. & Murthy, R.N. (2020). Deep learning-based slow DDoS attack detection in SDN-based networks. In *Proceedings of IEEE NFV-SDN*, 51-56.
26. Wei, Y., Jang-Jaccard, J., Sabrina, F., Singh, A., Xu, W. & Camtepe, S. (2021). AE-MLP: A hybrid deep learning approach for DDoS detection and classification. *IEEE Access, 9,* 146810-146821.
27. Hosseini, S. & Azizi, M. (2019). The hybrid technique for DDoS detection with supervised learning algorithms. *Computer Networks, 158,* 35-45.
28. Zainudin, A., Ahakonye, L.A.C., Akter, R., Kim, D.S. & Lee, J.M. (2023). An efficient hybrid-DNN for DDoS detection and classification in software-defined IIoT networks. *IEEE Internet of Things Journal, 10*(10), 8491-8504.
29. Sripriyanka, I.G. & Mahendran, A. (2024). Securing IoMT: A hybrid model for DDoS attack detection and COVID-19 classification. *IEEE Access, 12,* 17328-17348.
30. McClurg, J. (2021). Correct-by-construction network programming for stateful dataplanes. In *Proceedings of the ACM SIGCOMM Symposium on SDN Research (SOSR)*, 66-79.
31. Ja'afreh, M.A., Adhami, H., Alchalabi, A.E., Hoda, M. & El Saddik, A. (2022). Toward integrating software defined networks with the Internet of Things: A review. *Cluster Computing, 25*(3), 1619-1636.
32. Vergara, M., Ramos, L., Rivera-Campoverde, N.D. & Rivas-Echeverría, F. (2023). Engine-FaultDB: A novel dataset for automotive engine fault classification and baseline results. *IEEE Access, 11,* 126155-126171.
33. Sahin, E.K. (2020). Assessing the predictive capability of ensemble tree methods for landslide susceptibility mapping using XGBoost, gradient boosting machine, and random forest. *SN Applied Sciences, 2*(7), 1308.
34. Ebrahimy, H., Mirbagheri, B., Matkan, A.A. & Azadbakht, M. (2022). Effectiveness of the integration of data balancing techniques and tree-based ensemble machine learning algorithms for spatially-explicit land cover accuracy prediction. *Remote Sensing Applications: Society and Environment, 27,* 100785.
35. Namdari, A. & Durrani, T.S. (2021). A multilayer feedforward perceptron model in neural networks for predicting stock market short-term trends. *Operations Research Forum, 2*(3), 38.
36. Liu, Y., Pu, H. & Sun, D.W. (2021). Efficient extraction of deep image features using convolutional neural network (CNN) for applications in detecting and analysing complex food matrices. *Trends in Food Science & Technology, 113,* 193-204.
37. Özdemir, C. (2023). Derin öğrenme modelleri ve veri ön işleme yöntemleri ile çeltik yaprak hastalıklarının erken teşhisi. *Çukurova Üniversitesi Mühendislik Fakültesi Dergisi*, *38*(3), 807-817.
38. Nguyen-Duc, P., Nguyen, H.D., Nguyen, Q.H., Phan-Van, T. & Pham-Thanh, H. (2024). Application of long short-term memory (LSTM) network for seasonal prediction of monthly rainfall across Vietnam. *Earth Science Informatics, 17*(5), 3925-3944.
39. Korkmaz, C. ve Kacar, İ. (2024). Zaman serisinin kestirimi için uzun-kısa süreli bellek ağı yaklaşımı. *Çukurova Üniversitesi Mühendislik Fakültesi Dergisi*, *39*(4), 1053-1066.