Journal of Engineering and Technology for Industrial Applications

ITEGAM-JETIA

Manaus, v.12 n.57, p. 637-647. January/February, 2026.

DOI: https://doi.org/10.5935/jetia. v12i57.2929

ISSN ONLINE: 2447-0228

RESEARCH ARTICLE OPEN ACCESS

A NOVEL ADAPTIVE ENSEMBLE INTELLIGENCE FRAMEWORK FOR STRENGTHENING CLOUD SECURITY AGAINST DDOS THREATS FOR ACCURATE DETECTION AND MITIGATION

Abida.T[1]*, M. Shanmugapriya [2]

[1]Research Scholar, Department of Computer Science, Park's College (Autonomous), Tirupur, Tamil Nadu, India.
[2]Associate Professor, Department of Computer Science, Kongu Arts and Science College (Autonomous), Erode, Tamil Nadu, India

*[1]http://orcid.org/0009-0006-0433-0375, [2]http://orcid.org/0009-0008-7075-3098

Email: *aabitha93@gmail.com, priyasathyan@gmail.com.

ARTICLE INFO

Article History
Received: November 12, 2025
Revised: December 20, 2025
Accepted: January 15, 2026
Published: February 28, 2026

Keywords:
Cloud Security
Distributed Denial-of-Service
Deep Neural Networks, Quadratic Discriminant Analysis
Ensemble Learning
Attack Mitigation.

ABSTRACT

The high adoption of cloud computing has thoroughly increased the need to have good security controls in order to ensure the flexibility and uniformity of the services. One of the most disruptive categories of emerging cyber threats is the DDoS attacks that are capable of flooding system resources with very large volumes of traffic and advanced obfuscation methods. Traditional machine learning algorithms like K-Nearest Neighbors, Support Vector Machines, Decision Trees, and Logistic Regression will not perform well when exposed to high-dimensional, noisy, as well as imbalanced network data. To overcome these deficiencies, a novel ensemble deep learning structure along with a superior Quadratic Discriminant Analysis (QDA) is introduced in this paper to apply in the decision-level fusion. Deep neural networks are used to extract the hierarchical and discriminative aspects of traffic, and the classification under the overlapping feature distribution is optimized with QDA. Moreover, an adaptive ensemble stacking algorithm combines the results of several base learners, which optimizes predictions that are based on confidence-based weighting. The results indicate that traditional models achieved accuracies between 92% and 97%, while deep neural networks improved to 99.18% accuracy with 0.87% FPR. In contrast, the proposed Adaptive Stacking + QDA framework performed very well when compared with all baselines, reaching 99.43% accuracy, 99.36% precision, 99.25% recall, 99.30% F1-score, and the lowest false positive rate of 0.62%. These findings confirm that the proposed framework has the robustness and reliability for real-time cloud security. Experimental assessments of benchmark cloud-based DDoS datasets show that there are some effective enhancements in detection accuracy, precision, and false-positive reduction, which thoroughly highlight the framework's efficiency in real-time cloud security applications.

I. INTRODUCTION

Cloud computing marks a major shift toward delivering computing services in a centralized manner, where numerous services are hosted remotely on shared infrastructure. This model involves relocating data and applications from local systems to the "cloud," offering highly scalable, flexible, and cost-efficient resources under a "pay-as-you-go" model. It has gained widespread adoption across industries due to its ability to minimize infrastructure costs, reduce deployment time, and offer on-demand services. The core service models—Software as a Service (SaaS), Platform as a Service (PaaS), and Infrastructure as a Service (IaaS)—enable users to leverage cloud functionalities without direct hardware management.However, with the rise in cloud adoption, there has been a concurrent increase in sophisticated cyberattacks targeting cloud-based infrastructure. Among these, Sharma, [1] Distributed Denial-of-Service (DDoS) attacks are particularly damaging, affecting the Internet of Things (IoT), Vishwakarma, [2] smart cities, healthcare systems, and enterprise platforms.

DDoS attacks exploit the very nature of shared and scalable cloud services by flooding target servers or applications with massive traffic, this leads to exhaustion of computational resources, which makes services inaccessible for genuine users. DDoS attacks are classified into two main categories: (i) Reflection-based attacks, where attackers send large volumes of request traffic using spoofed IP addresses through intermediary servers, which unknowingly redirect the traffic to the victim; and (ii) Direct flood attacks, where massive traffic is sent directly from compromised devices under the attacker's control Sallam, [3]. Common techniques include TCP SYN flooding and spoofing, which result in the victim server exhausting its capacity to respond to legitimate requests. As the scale and frequency of DDoS attacks increase, so does their complexity. Modern DDoS threats are dynamic and capable of adapting to countermeasures, making real-time detection and mitigation increasingly difficult. Based on these issues, machine learning (ML) approaches have been considered to detect DDoS at very high-speed cloud traffic with a minimum of false positives Tahsien, [4]. K-Nearest Neighbors, Support Vector machines, Decision Trees, and Logistic Regression algorithms have proved to be very useful in studying traffic trends. Nonetheless, high-dimensional data, imbalanced classes, and dynamic attack patterns that are common in cloud settings, which tend to be a challenge to these models, deep learning (DL) methods have become more resilient to these challenges.

Deep learning involves the use of several nonlinear layers to automatically extract hierarchical features in some of the very important input data, like traffic logs, system signals, or metadata. Architectures such as Convolutional Neural Networks (CNNs) have been shown to be very useful in more complex pattern recognition problems, such as image classification and anomaly detection. The proposed work presents a new and refined DDoS detection model that thoroughly integrates ensembled deep learning with Quadratic Discriminant Analysis (QDA) to perform a strong classification. Deep networks are used to extract highly complex features, but QDA increases class separability, particularly with overlapping traffic patterns. Moreover, an adaptive ensemble stacking mechanism is introduced in this study, in which multiple base learners are dynamically added to the final decision, according to real-time performance measures. Such a combination guarantees down-weighting of poorly-performing classifiers, and dominance of strong learners in the final output, which thoroughly enhances overall accuracy and responsiveness to novel threats. Since cloud-based services are dynamic and at large scale, the combination of deep learning, QDA, and adaptive ensemble methods is a promising way to achieve scalable as well as intelligent DDoS detection systems.

I.1 CLOUD CHARACTERISTICS

Infrastructure as a Service (IaaS) allows users to fully access and control virtualized computing resources via the internet. IaaS allows the user to dynamically provision resources and to configure software environments without investing in physical infrastructure as well. This pay-as-you-use model allows organizations to scale services in an effective manner, which pays only based on the virtual machine space or time used. Notably, Platform as a Service (PaaS) is enhanced by providing a complete software development environment, such as operating systems, development tools, and database management systems as well. Although cloud services are very beneficial, users develop and deploy applications through secure channels such as VPNs and private networks. Cloud services expose users to certain vulnerabilities. An example of direct DDoS attacks is when compromised devices transmit traffic directly to a target server and flood the server with packets. The traffic can be of numerous different sources with virus carriers, but the point of attack is centralized disruption, commonly through flaws in protocols like the TCP SYN flood. In such an approach, the attacker will use TCP handshake to ping the server with a lot of SYN packets, which thoroughly triggers the server to wait until the connection is acknowledged, which never happens, and thereby fills its connection queue. Spoofing techniques are also employed to redirect responses, leading to further confusion and traffic buildup.

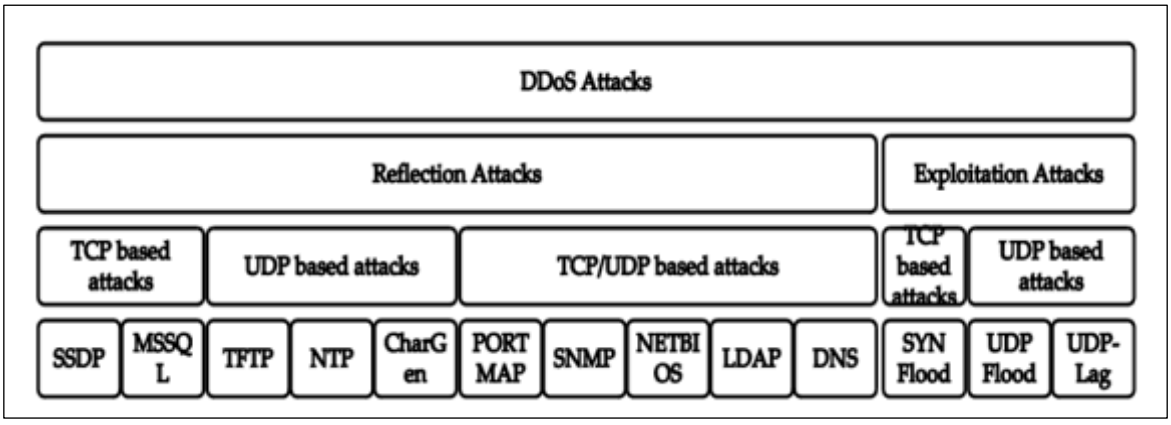

Figure 1: Taxonomy of DDoS Attack.
Source: [5].

As cloud applications continue to scale, the importance of intelligent detection systems becomes paramount. Traditional rule-based security mechanisms fall short in distinguishing between legitimate spikes in traffic and malicious floods. Deep learning, through its layered architecture of input, hidden, and output layers, has proven effective for automated feature extraction and classification from large-scale data, including network traffic. CNNs and other deep models can process packet-level details and derive abstract representations for anomaly detection. In the proposed system, we integrate de nep learning with QDA as a fusion classifier, combining the strength of deep models in feature learning with QDA's capability to model class variance. Moreover, an adaptive ensemble stacking framework is incorporated, allowing the system to adaptively weigh and combine predictions from multiple models based on their real-time performance. This thoroughly helps to improve the system's resilience against evolving DDoS strategies and reduces false alarms, which makes it very effective for cloud-based DDoS detection.

II. RELATED WORKS

Within the past ten years, the usage of deep learning (DL) methods within networking and cybersecurity has experienced significant velocity. DDoS attacks are a very significant issue in this space, mainly because of the challenge of distinguishing between bad traffic and normal traffic in a dynamic cloud traffic environment. Deep learning-based classifiers have demonstrated the capability to automatically discover the appropriate features in raw network traffic, as well as differentiate between benign and malicious traffic. A number of studies have used classical machine learning (ML) algorithms like Naive Bayes and Random Forest to identify and characterize DDoS traffic patterns [6]. Although these techniques are very effective with structured data, they are ineffective in data dimensionality and non-linear relationships settings. DDoS detection is also a common area of research by Artificial Neural Networks (ANNs). As an example, a model created by [7] was able to identify various categories of DDoS attacks, but with lower accuracy, especially when it was applied to the classification of UDP-based floods because it did not allow capturing complicated time-related traits of the data. [8] suggested a traffic analysis method based on the features of Time to Live (TTL), protocols, source ports, and IP addresses.

Their model, trained on an outdated dataset, provided initial insights into static feature-based detection but lacked adaptability to newer attack vectors. In further work, [8] also introduced an edge-centric detection model involving Long Short-Term Memory (LSTM) and Convolutional Neural Network (CNN) architectures. These models achieved high accuracy—98.9% for LSTM and 99.9% for CNN—by learning temporal and spatial features in traffic, and were optimized for deployment on resource-constrained edge devices to reduce operational latency. Entropy-based detectors integrated with Software Defined Networking (SDN) have also been explored to identify anomalies in IoT-driven cloud ecosystems. The system proposed by Galeano-Brajones [8] achieved a detection confidence ranging from 68% to 99.7% based on network state monitoring. This reflects the growing trend of combining intelligent classifiers with programmable networks for adaptive DDoS mitigation. Despite the progress, most existing approaches rely on single-model classifiers or traditional stacking techniques that do not adjust dynamically to varying data distributions.

As attacks evolve, fixed models are prone to concept drift, leading to degraded detection performance. [9] noted that the increasing scale and complexity of modern DDoS attacks outpace static detection frameworks, particularly in IoT and smart environments where devices have limited processing power. The large-scale DDoS attack in 2016, which disrupted major ISPs and impacted services like Netflix, CNN, and Twitter, is a strong reminder of the financial and reputational risks posed by these threats [10]. Several works have achieved strong results using deep learning. In a comparative analysis, [11] demonstrated that SVM outperformed traditional models in accuracy. Similarly, [12] evaluated five different classifiers in IoT-based networks and achieved near-perfect accuracy (>99.9%). However, the study acknowledged that models trained on offline datasets still struggled with unseen traffic features in real-time deployment. [13] developed an AI-based DDoS detection model that employed feature extraction and machine learning to detect malicious traffic. It was providing a high degree of accuracy although it was based on offline training and fixed features.

This restricts its flexibility to changing attacks resulting in the possible inefficiency and increased false positives in changing networks. [14] employed an ANN-based model that was trained on a limited feature set (about five features), resulting in inconsistent classification across different DDoS variants. Neural Network (NN) and Naive Bayes (NB) methods to DDoS classification proposed by Yudhana, I. [15] were used to analyze network forensics, and their performance was superior to other classification methods. The method is however constrained by high computation cost and low flexibility to changing patterns of attack in real-time networks. A comprehensive survey by [16] revealed that while multiple ML techniques could detect DDoS attacks with good accuracy, no single model consistently outperformed the others across all datasets. Additionally, their study lacked detail on the specific TCP/IP and ICMP-level features used for training. Thapngam [17] also found over 98% accuracy in their ML models but noted the limitation of relying on outdated and overly feature-rich datasets. Similarly, [5] demonstrated in an SDN environment that classifier accuracy often drops significantly when tested on live traffic compared to offline datasets.

[18] introduced the use of machine learning to detect DDoS attacks in cloud computing configuration whereby a greater accuracy of detection was attained, however; it experienced difficulties in the ability to deal with changing attack patterns dynamically.The research by Mouli, [19] on the systematic literature review of web services attacks and security discusses typical vulnerabilities and vectors of attacks, but this study has not been effective in the practical implementation and assessment of the defense mechanisms. The security of web service frameworks against Denial-of-Service attacks were evaluated by Oliveira, [20], which is a vast evaluation of the weaknesses of frameworks; the study did not however offer automated detection propositions. [21] studied the effect of the announcement of DDoS attacks on stock values of the victims, which provided information about the economic effects, but did not consider technical methods of detection and prevention. Subbulakshmi, [22] was able to come up with Enhanced Support Vector Machines (SVM) and a real-time generated data to be used in detecting DDoS attacks, which had a better classification but consumed many computational resources and could not handle large-scale traffic.

[23] discussed the problems and issues in countering DDoS attacks and found that there were major gaps in the current security practices, but it had no specific implementation plans. Samtani, [24] investigated how Artificial Intelligence (AI) could be used in cybersecurity, which formed a basis to be able to detect attacks intelligently, but the methods proved to be theoretical and could not adapt to the real-time environment. Zarpelão, [25] surveyed intrusion detection in Internet of things (IoT) noting various methods of detection, but could not be scaled and limited resources. [26] suggested an intrusion detection technology based on immunity of IoT, which provided the adaptive detection performance but had the drawback of high complexity and implementation in non-homogeneous devices. In 6LoWPAN-based IoT networks, Denial-of-Service detection was proposed by [27]. Kasinathan, P et al., and was able to detect efficiently in a constrained setting, however, it could not cope with changing attack patterns and network dynamics. Given these limitations, the proposed work extends existing research by introducing an adaptive ensemble-based approach that not only employs deep learning for rich feature extraction but also integrates Quadratic Discriminant Analysis (QDA) to improve classification boundaries for overlapping traffic patterns.

The novelty is in the application of adaptive ensemble stacking, which dynamically assesses and weights base models upon real-time detection confidence as well as accuracy. This provides a great enhancement over previous fixed-model techniques and also provides very high resilience in changing cloud-based DDoS threat environments.

III. METHODOLOGY

In this section, the different machine learning and deep learning models have been discussed, which are employed in the proposed model for identifying Distributed Denial-of-Service (DDoS) attacks within a cloud setup. It starts by describing the individual traditional classifiers as base learners. Next, the suggested hybrid model combining a Deep Neural Network (DNN) and Quadratic Discriminant Analysis (QDA) with the adaptive stacking mechanism was presented. This hybrid architecture thoroughly ensures the successful learning of complex and overlapping traffic patterns and adaptively responds to changing attack behavior as well.

III.1 TRADITIONAL CLASSIFIERS

In order to cover a wide range of traffic patterns and provide model diversity, we include multiple classical machine learning classifiers in the bottom layer of the ensemble. These models offer different decision boundaries and generalization facilities, which are very essential when dealing with dynamic and imbalanced network traffic information. The Naive Bayes classifier is a probabilistic learning algorithm that uses the Bayes theorem, subject to the assumption that the features are individually conditionally independent. It approximates the posterior probability of classes using previous probabilities and probability based on the training data. Naive Bayes is computationally inexpensive and has a good performance in those cases where the distribution of the features fits its assumption of independence. But its power to isolate subtle traffic variations is not so much, particularly in situations where correlated network traits are at stake. A Decision Tree classifier is a supervised learning algorithm that recursively splits the data set depending on the values of features to create a tree-like model.

Internal nodes indicate decisions that are made on feature attributes, and the leaf nodes are the class label. Decision Trees are also very useful in finding interpretable rules based on traffic data. Nevertheless, they tend to overfit, particularly when trained on noisy or highly skewed datasets, which is very common in the real-world cloud setup. The K-Nearest Neighbors (KNN) algorithm is a non-parametric classifier that thoroughly helps to classify a given input in the feature space by the majority of the closest K neighbors. KNN is effective in capturing local data variations and works well in distinguishing between different stages of DDoS attacks, such as flooding and probing. Nevertheless, its performance degrades significantly in high-dimensional settings, and it is computationally inefficient for real-time classification tasks. The Support Vector Machine (SVM) is a powerful binary classification technique that constructs an optimal hyperplane to separate classes with the maximum margin.

It utilizes kernel functions to transform non-linearly separable data into higher dimensions where linear separation becomes possible. SVM is very accurate on balanced data sets with well-defined boundaries, but has been found to be difficult to scale and adapt to changing or noisy attacks. Random Forest classifier is an ensemble algorithm that consists of many decision trees, where each decision tree is trained on a bootstrap sample, and features are selected randomly. Majority voting on all trees makes predictions. This model is better at generalization and overfitting and works very well on high-dimensional network traffic. Random Forests are able to identify a wide variety of attack patterns, though they can lose interpretability with more trees. The Stochastic Gradient Descent (SGD) classifier is a linear model whose optimization is based on gradient descent. It is specifically adapted to large-scale problems and to online education as well.

SGD modifies model weights on a per-training instance or mini-batch basis, and therefore can be used to learn streaming data effectively. But it is also sensitive to feature scaling, and it needs to be hyperparameter-tuned for convergence and stability. The Deep Neural Network (DNN) is the deep learning part of our base learners. It comprises multiple hidden layers with non-linear activation functions, enabling hierarchical feature extraction from raw input data. The DNN model learns complex, non-obvious relationships in network traffic, allowing it to generalize across a wide variety of known and unknown DDoS attack types. Backpropagation and optimization techniques such as Adam or RMSProp are used to train the model effectively. However, without proper regularization, DNNs may overfit the training data, particularly in scenarios with class imbalance.

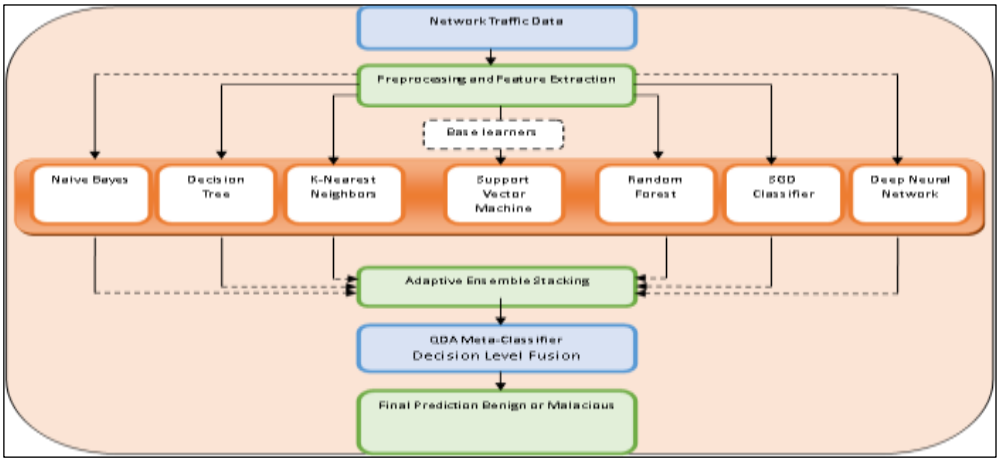

Figure 2: Architecture of the Proposed DDoS Detection Framework using Deep Learning, Adaptive Ensemble Stacking, and QDA Fusion (Self-sourced).
Source: Authors, (2026).

III.2 ADAPTIVE DEEP ENSEMBLE FRAMEWORK (DNN–QDA INTEGRATION) FOR CLOUD DDOS DETECTION

A novel hybrid framework is introduced in this study that integrates Deep Neural Networks (DNN) with Quadratic Discriminant Analysis (QDA) in an adaptive ensemble stacking mechanism. The DNN extracts very complex hierarchical traffic characteristics, whereas QDA guarantees the fine-grained decision fusion by extracting the class-specific variances. The adaptive stacking layer is dynamic with regard to changing model weights based on unlabeled traffic and thus completely self-learns and resists the changing format of attacks as well. Combining deep learning with adaptive statistical decision fusion, the proposed system demonstrates very high accuracy, real-time scalability, and false alarms when detecting cloud-based DDoS.

Algorithm 1: Adaptive Deep Ensemble Framework (DNN–QDA Integration) for Cloud DDoS Detection.

Input:

- Training dataset $D = (x1, y1), (x2, y2), \ldots, (xn, yn)$
- Base learners $B1, B2, \ldots, Bm$ (Naïve Bayes, SVM, RF, etc.)
- Deep Neural Network (DNN)
- Unlabeled traffic data U
- Small labeled dataset L

Output:

- Final predicted class label $\hat{y}$
- Updated labeled dataset L

Steps:

1. **Initialization:**

Partition dataset: $L\leftarrow$ small labeled subset, $U\leftarrow$ unlabeled data.
Pre-train DNN on L to extract hierarchical traffic features.

2. **Base Learner Training:**

Train each base learner B_i on feature space from DNN + raw features.

3. **Prediction on Unlabeled Data:**

For each instance $x \in U$:
a. Obtain predictions $p_i(x)$and confidence scores $c_i(x)$ from each B_i.
b. Normalize scores to compute adaptive weights $w_i(x$.

4. **Adaptive Ensemble Aggregation:**

Compute weighted prediction:

$$y_{agg}(x) = \sum_i w_i(x) \cdot p_i(x)$$

5. **Meta-Classification with QDA:**

Apply Quadratic Discriminant Analysis (QDA) on $y_{agg}(x))$ to estimate posterior probabilities.
Assign final label:

$$\hat{y} = argmax_k [\, P(y = k|x)]$$

6. **Self-Training Update:**

If $\hat{y}(x)$is predicted with high confidence, move xxx from U to L with label $\hat{y}(x)$
Retrain base learners and update DNN feature space using expanded L.

7. **Iteration:**

Repeat Steps 3–6 until stopping criteria (no more confident predictions or convergence).

Return: Final prediction $\hat{y}$ and updated labeled dataset L.

Source: Authors, (2026).

This algorithm unifies deep feature extraction (DNN), statistical decision fusion (QDA), as well as adaptive ensemble stacking with self-training, which thoroughly enables robust as well as dynamic DDoS detection in cloud environments.

III.3 QUADRATIC DISCRIMINANT ANALYSIS (QDA)

Quadratic Discriminant Analysis (QDA) is a supervised learning algorithm used in our framework as a meta-classifier to perform decision-level fusion. Unlike Linear Discriminant Analysis (LDA), which assumes equal covariance among classes, QDA permits each class to have its own covariance structure. This flexibility makes it especially effective in handling non-linear and overlapping feature distributions—an essential property for classifying cloud network traffic where benign and malicious behavior often intersect.QDA operates under the assumption that the data from each class follows a Gaussian distribution, characterized by its own mean vector and covariance matrix. It calculates the posterior probability of each class for a given input and assigns the label with the highest probability. In our architecture, QDA plays a critical role in combining the outputs from multiple base classifiers, leveraging its capacity to distinguish subtle variations in feature distributions caused by evolving attack patterns.

III.3.1 QDA Mathematical Formulation

The QDA classifier relies on three core equations:
Class-conditional likelihood (Gaussian distribution)

$$P(^{x}/_{y}=k)=\frac{1}{(2\pi)^{d/2}|\Sigma k|^{1/2}}exp\left(-\frac{1}{2}(x-\mu_k)^T\sum_{\mathrm{k}}^{-1}(x-\mu_k)\right) \quad (1)$$

Posterior probability (Bayes' theorem)

$$P(y=k/x)=\frac{P(x|y=k)*P(y=k)}{\sum_{=1}^{\mathrm{k}}P(x|y=j)*P(y=j)} \quad (2)$$

Decision rule (QDA classifier)

$$\hat{y}=\ argmax_k[\ P(y=k|x)] \quad (3)$$

Equations (1), (2), and (3) ensure that QDA captures the unique variance of each class, improving classification accuracy when working with overlapping or imbalanced cloud traffic features.

III.3.2 QDA Algorithm

The algorithm below outlines the QDA implementation used for decision fusion in our model.

Algorithm 2: Quadratic Discriminant Analysis (QDA) for Decision Fusion.

Input:
- *Training dataset* $D\ =\ \{(x1,y1),(x2,y2),\dots,(xn,yn)\}$
- *Prediction outputs from base classifiers*
- *New input instance* x

Output:
- *Final predicted class label* $\hat{y}$

Steps:

1. *For each class* k *in* $\{1,2,\dots,K\}$:
   a. *Compute class mean vector* μk *from training data*
   b. *Compute class covariance matrix* Σ_k
   c. *Estimate prior probability* $P(y=k)$
2. *For the new input instance* x:
   a. *For each class* k:
      i. *Compute class − conditional likelihood*:
      $P(x|y=k)=\ Gaussian\_PDF(x;\ \mu_k,\Sigma_k)$
      ii. *Compute posterior probability*:

$$P(y=k|x)=\frac{[P(x|y=k)*\ P(y=k)]}{\sum_{=1}^{\mathrm{k}}[P(x|y=j)*\ P(y=j)]}$$

3. *Assign final label*:

$$\hat{y}=\ argmax_k[\ P(y=k|x)]$$

4. *Return* $\hat{y}$ *as the fused decision.*

Source: Authors, (2026).

This algorithm ensures that QDA captures the unique variance of each class and yields high discrimination accuracy when combined with diverse base classifiers.

III.4 ADAPTIVE ENSEMBLE STACKING

To overcome the limitations of static ensemble methods and improve robustness, we introduce an adaptive ensemble stacking mechanism. In traditional stacking, base learners are trained independently, and a meta-learner is used to combine their outputs using fixed weights. However, this approach does not account for variations in model performance across different traffic scenarios. Our adaptive stacking framework addresses this issue by dynamically adjusting the contribution of each base classifier based on real-time confidence metrics such as accuracy, false positive rate, and output certainty. The adaptive ensemble stacking module continuously monitors the performance of base learners on recent data and assigns them confidence scores accordingly. These scores are normalized and used as dynamic weights for combining predictions. This strategy allows the ensemble to prioritize base models that are currently more reliable, thus improving overall decision accuracy, especially when facing previously unseen or evolving attack vectors. The final fusion is handled by the QDA meta-classifier, which takes these weighted predictions as input and learns to make the final classification based on the underlying data distribution. The interaction between adaptive weighting and QDA fusion ensures that the model remains flexible, accurate, and robust under changing network conditions. The algorithm below describes the adaptive stacking mechanism integrated into our DDoS detection framework.

Algorithm 3: Adaptive Ensemble Stacking for DDoS Detection.

Input:

- P: Predictions from base learners
- U: Unlabeled traffic data
- L: Small labeled dataset

Output:

- Final class prediction
- Updated labeled dataset L

Steps:

1. **Initialization:**
Set $L \leftarrow$ small labeled set
Set $U \leftarrow$ remaining unlabeled instances
2. **Base Learner Training:**
Train each base learner B_i on L.
3. **Prediction on Unlabeled Data:**
For each instance $x \in U$:
Obtain prediction $pi(x)$ and confidence score $ci(x)$ from each base learner B_i
4. **Adaptive Weight Computation:**
Normalize confidence scores $ci(x)$ to obtain adaptive weights $wi(x)$
5. **Ensemble Aggregation:**
Compute weighted prediction:

$$y_{agg}(x) = \sum_i w_i(x) \cdot p_i(x)$$

6. **Meta-Classification:**
Use QDA (Quadratic Discriminant Analysis) meta-classifier on $y_{agg}(x)$ to predict final class label $y(_x)$
7. **Self-Training Update:**
If $y(_x)$is predicted with high confidence:
Move x from U to L with assigned label $y(x)$
8. **Iteration:**
Update base learner confidence profiles.
Retrain base learners on expanded L.
Repeat steps **3–8** until stopping criteria are met (e.g., no more confident predictions or convergence).

Source: Authors, (2026).

By repeating these steps iteratively, the model continuously improves as more data is labeled and fed into the system. This adaptive process ensures real-time responsiveness to traffic distribution changes while maintaining high detection accuracy and minimizing false positives.

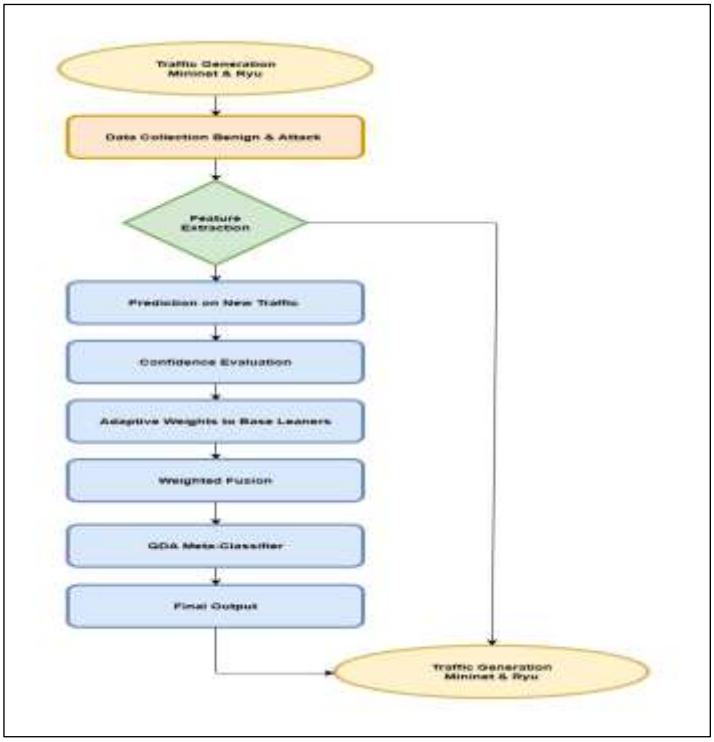

Figure 3: Workflow of the DDoS Detection Framework from Traffic Collection to Final Classification (Self-sourced).
Source: Authors, (2026).

IV. IMPLEMENTATION RESULTS

To evaluate the effectiveness of the proposed adaptive ensemble framework for DDoS detection, we conducted extensive experimentation using a Software Defined Networking (SDN)-based dataset developed through the Mininet emulator. The dataset was specifically tailored to replicate realistic network conditions and enable robust machine learning and deep learning-based classification of traffic flows. A total of ten distinct network topologies were configured in the Mininet environment, each managed by a centralized Ryu controller interfacing with OpenFlow switches. Both benign and malicious traffic types were generated within these topologies. Legitimate traffic included UDP, ICMP, and TCP protocols, while malicious traffic comprised TCP SYN flood, ICMP flood, and UDP flood attacks. The inclusion of diverse attack types ensured the dataset's relevance for detecting complex DDoS behavior patterns. Each traffic instance in the dataset includes 23 carefully selected features. These features were either directly collected from the SDN controller or derived through computed metrics to provide a granular view of network activity. Core features such as packet_count, byte_count, switch_id, duration_sec, source_ip, destination_ip, tx_bytes, and rx_bytes were combined with higher-level indicators like byte_per_flow, packet_rate, flow_table_entries, tx_kbps, port_bandwidth, and others. These features capture both volumetric and behavioral aspects of the network flows. The dataset is labeled such that a class value of '0' represents benign traffic and '1' indicates malicious activity, establishing a binary classification problem. Over a 250-minute simulation period, the environment generated a comprehensive dataset comprising 104,345 traffic samples.

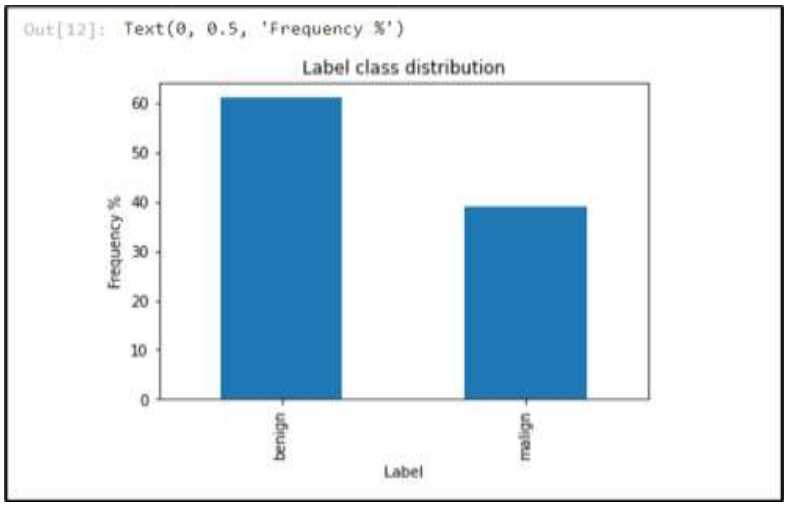

Figure 4: Class distribution of DDos Attacks.
Source: Authors, (2026).

Prior to training, the dataset underwent preprocessing that included data normalization, feature scaling, and timestamp alignment. Several base classifiers, including Random Forest, Support Vector Machine (SVM), Decision Tree, K-Nearest Neighbors (KNN), Stochastic Gradient Descent (SGD), and Deep Neural Network (DNN), were independently trained as a very first step to give initial predictions after preprocessing. These classifiers generated prediction scores and confidence values, which in turn were passed on to the adaptive ensemble stacking module. The adaptive stacking layer dynamically evaluated the reliability of every base learner using real-time confidence measurements (extracted accuracy, false positivity, and output confidence). Classifiers with very high and consistent performance were given more weight in the process of aggregation. This confidence-weighted system thoroughly enabled the ensemble to be sensitive to network behavioral changes and also provided resistance to distribution drift and novel attack patterns. At the last prediction stage, a Quadratic Discriminant Analysis (QDA) meta-classifier was used to fuse the prediction outputs of the adaptively weighted base learners. The use of QDA enabled the model to deal with overlapping feature distribution in a very effective manner through modeling each class with its own covariance structure. This had been especially useful in cloud traffic, where both malicious and benign flows can frequently exhibit similar properties. The nonlinear decision boundaries that QDA was able to build also increased the accuracy of the ensemble in the classification.

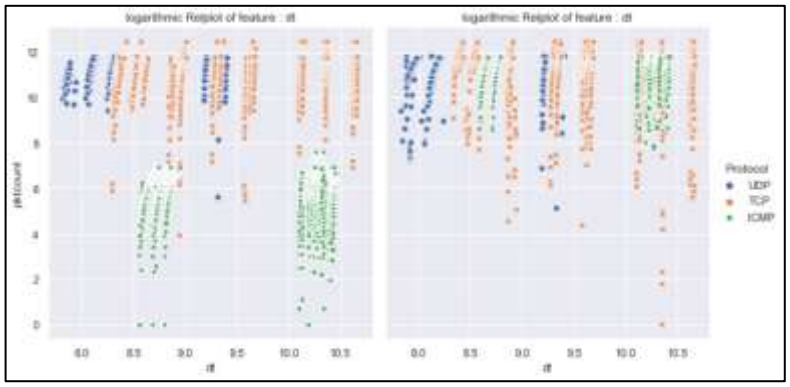

Figure 5: The distribution of continuous features with respect to packet count, protocol, and type of attack.
Source: Authors, (2026).

The Deep Neural Network (DNN) baseline was fine-tuned by hyperparameter optimization and architecture tuning. Activation functions, learning rates, batch sizes, and layer depths were iteratively tested to identify the optimal configuration. The DNN achieved a precision of 99.18%, making it the most performant standalone classifier in our experiments and establishing a strong lower bound for the ensembles expected performance.

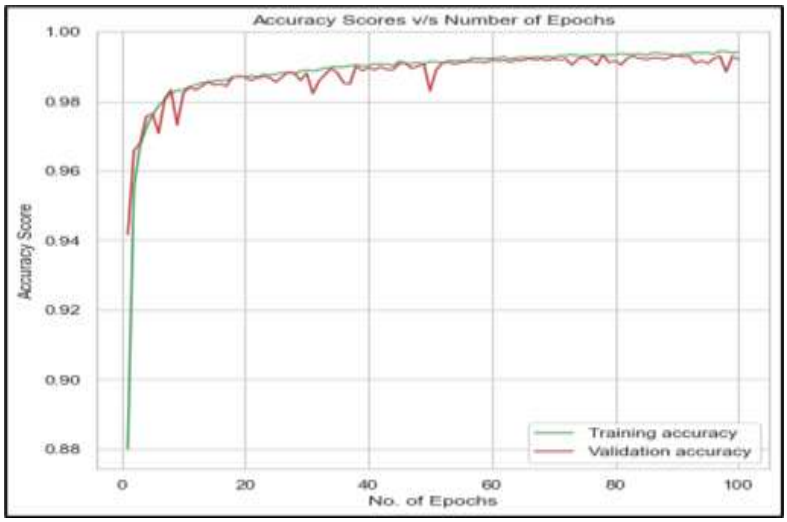

Figure 6: Plotting accuracy Vs epochs.
Source: Authors, (2026).

To clearly demonstrate the advantage of our adaptive ensemble stacking with QDA, we compare its performance against each individual base learner and traditional ensemble methods in Table I. The proposed framework consistently outperforms the baselines in all key metrics, especially in reducing the false positive rate and improving the F1-score, two of the most critical aspects in DDoS detection.

Table 1: Performance Comparison of Classifiers and Proposed Model.

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	False Positive Rate (%)
Naïve Bayes	92.67	90.21	89.75	89.98	6.40
Decision Tree	94.38	93.10	91.85	92.47	4.88
K-Nearest Neighbors (KNN)	95.12	93.56	92.33	92.94	3.89
SVM	96.45	95.20	94.81	95.00	2.95
Random Forest	97.82	96.70	96.12	96.41	1.98
SGD Classifier	95.87	94.01	93.48	93.74	3.12
Deep Neural Network (DNN)	99.18	99.18	98.75	98.96	0.87
Proposed Model (Adaptive Stacking + QDA)	**99.43**	**99.36**	**99.25**	**99.30**	**0.62**

Source: Authors, (2026).

As shown in Table I, the proposed adaptive stacking model with QDA yields the highest overall accuracy and the lowest false positive rate among all models evaluated. While the standalone DNN achieved impressive performance, it still exhibited slightly higher false positives in borderline cases. The ensemble, by contrast, leverages the strengths of multiple learners while mitigating individual weaknesses through adaptive confidence weighting and discriminative fusion via QDA.

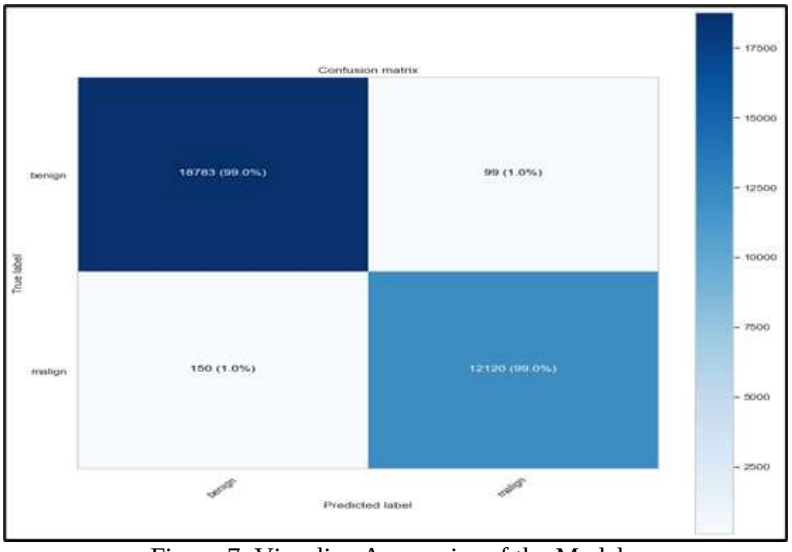

Figure 7: Visualize Accuracies of the Models.
Source: Authors, (2026).

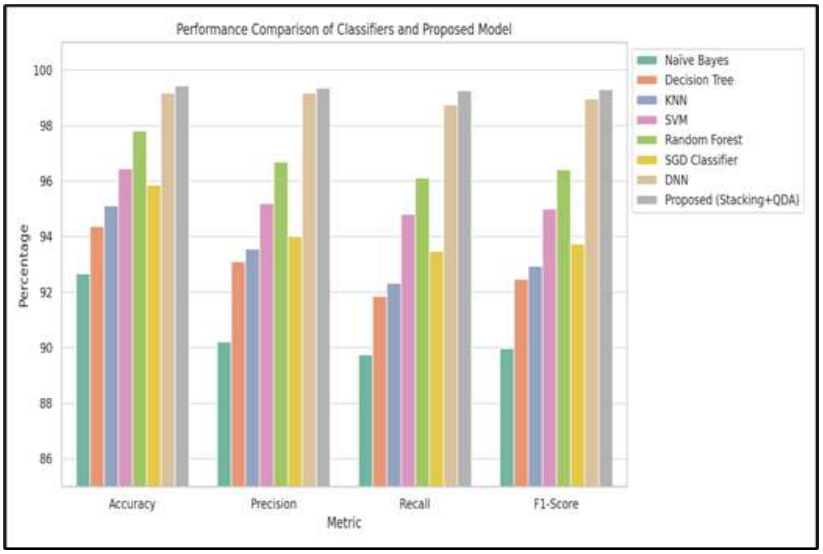

Figure 8: Confusion matrix.
Source: Authors, (2026).

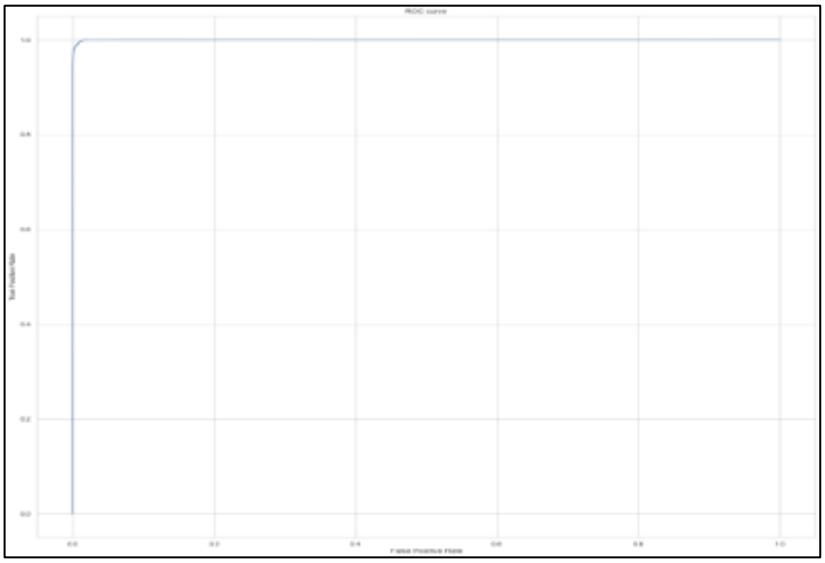

Figure 9: ROC-AUC curve of proposed model.
Source: Authors, (2026).

The confusion matrix confirmed the ensemble model's ability to minimize misclassifications of both benign and malicious traffic. Additionally, the ROC-AUC curve represented an almost perfect distinction of the two classes, which once again confirmed the applicability of the model in detecting real-time DDoS attacks in cloud-integrated SDN systems. The dimensionality of the feature space (high-dimensional), combined with the adaptability of the ensemble and the discriminative ability of QDA, thoroughly helped to explain the higher performance of the suggested model in all evaluation measures as well. In summary, the experimental findings confirmed the strength as well as efficacy of the suggested hybrid framework. The model was shown to be a very effective and reliable real-time detector with low false positive rates, thus making it a very viable solution to current cloud infrastructures where security and availability are a priority.

V. CONCLUSION

In this paper, a powerful deep learning-based architecture to identify the Distributed Denial-of-Service (DDoS) attacks in Software-Defined Networking (SDN)-enabled cloud infrastructure was presented. The Mininet emulator, along with Ryu controller, was used to create a realistic dataset including benign and malicious traffic over a variety of topologies. The criterion set was 23 important features and more than 104,000 cases, which were relevant and diverse to train. The initial evaluation was done with the baseline machine learning models: Naive Bayes, decision tree, KNN, SVM, random forest, and SGD. The highest standalone accuracy of a Deep Neural Network (DNN) was 99.18%. To further enhance the performance, it was suggested to use an adaptive ensemble stacking strategy. This dynamically varied the influence of each base classifier according to its performance and prediction confidence, responding to the overlapping features and to changing attacks. QDA was used in the decision layer as a meta-classifier, which gives nonlinear boundaries and better separation of classes. This enhanced the capability of the framework to differentiate between malicious and benign traffic across high-dimensional spaces. The last ensemble using QDA had an accuracy of 99.43 and a low false positive rate of 0.62, and was better than all single models. To conclude, deep learning with adaptive stacking and discriminant-based fusion is quite effective in detecting DDoS in SDN-enabled cloud environments on a large scale and with high accuracy.

VI. REFERENCES

[1] Sharma, K., Mukhopadhyay, A., Cyber Risk Assessment and Mitigation Using Logit and Probit Models for DDoS attacks, Americas Conference on Information systems (AMCIS 2020), August 2020, pp. 1-9.

[2] Vishwakarma, R., Jain, A. K., A Survey of DDoS Attacking Techniques and Defence Mechanisms in the IoT Network. Telecommunication Systems, Vol. 73, No. 1, 2020, pp. 3-25.

[3] Sallam, A. A., Kabir, M. N., Alginahi, Y. M., Jamal, A., Esmeel, T. K., IDS for improving DDoS Attack Recognition Based on Attack Profiles and Network Traffic Features. In 2020 16th IEEE International Colloquium on Signal Processing & Its Applications (CSPA), IEEE, February 2020, pp. 255-260.

[4] Tahsien, S. M., Karimipour, H., Spachos, P., Machine Learning based Solutions for Security of Internet of Things (IoT): A Survey. Journal of Network and Computer Applications, Vol. 161, 2020, art. No. 102630.

[5] Sahi, D. Lai, Y. Li, and M. J. I. A. Diykh, "An efficient DDoS TCP flood attack detection and prevention system in a cloud environment," vol. 5, pp. 6036-6048, 2017.

[6] Chen, Y. W., Sheu, J. P., Kuo, Y. C., Van Cuong, N., Design and Implementation of IoT DDoS Attacks Detection System based on Machine Learning. In 2020 European Conference on Networks and Communications (EuCNC), IEEE, June 2020, pp. 122-127.

[7] Jia, Y., Zhong, F., Alrawais, A., Gong, B., Cheng, X., Flowguard: An Intelligent Edge Defense Mechanism against IoT DDoS Attacks, IEEE Internet of Things Journal, Vol. 7, No. 10, May 2020, pp. 9552-9562.

[8] Galeano-Brajones, J., Carmona-Murillo, J., Valenzuela-Valdés, J. F., Luna-Valero, F., Detection and Mitigation of DoS and DDoS Attacks in IoT-based Stateful SDN: An Experimental Approach, Sensors, Vol. 20, No. 3, 2020, Art. No. 816.

[9] Bhardwaj, K., Miranda, J. C., Gavrilovska, A., Towards IoT-DDoS Prevention using Edge Computing, In {USENIX} Workshop on Hot Topics in Edge Computing (HotEdge 18),Boston, MA, USA, July 2018, pp. 1-7

[10] Kumar, R.; Kumar, P.; Tripathi, R.; Gupta, G.P.; Kumar, N.; Hassan, M.M. A Privacy-Preserving-Based Secure Framework Using Blockchain-Enabled Deep-Learning in Cooperative Intelligent Transport System. IEEE Trans. Intell. Transp. Syst. 2021.

[11] B. Nagpal, P. Sharma, N. Chauhan, and A. Panesar, "DDoS tools: Classification, analysis and comparison," in 2015 2nd International Conference on Computing for Sustainable Global Development (INDIACom), pp. 342-346, 2015.

[12] Y. S. Abu-Mostafa, M. Magdon-Ismail, and H.-T. Lin, Learning from data. AMLBook New York, NY, USA:, 2012.

[13] B. Zhang, T. Zhang, and Z. Yu, "DDoS detection and prevention based on artificial intelligence techniques," in 2017 3rd IEEE International Conference on Computer and Communications (ICCC), pp. 1276-1280, 2017.

[14] Alsirhani, S. Sampalli, and P. Bodorik, DDoS Detection System: Using a Set of Classification Algorithms Controlled by Fuzzy Logic System in Apache Spark (2019 IEEE Transactions on Network and Service Management). IEEE, pp. 1932-4537, 2019.

[15] Yudhana, I. Riadi, F. J. I. J. O. A. C. S. Ridho, and APPLICATIONS, "DDoS Classification Using Neural Network and Naïve Bayes Methods for Network Forensics," vol. 9, no. 11, pp. 177-183, 2018.

[16] D. Peraković, M. Periša, I. Cvitić, and S. J. T. J. Husnjak, "Model for detection and classification of DDoS traffic based on artificial neural network," vol. 9, no. 1, p. 26, 2017.

[17] T. Thapngam, S. Yu, W. Zhou, S. K. J. P.-t.-p. n. Makki, and applications, "Distributed Denial of Service (DDoS) detection by traffic pattern analysis," vol. 7, no. 4, pp. 346-358, 2014.

[18] M. Zekri, S. El Kafhali, N. Aboutabit, and Y. Saadi, "DDoS attack detection using machine learning techniques in cloud computing environments," in 2017 3rd International Conference of Cloud Computing Technologies and Applications (CloudTech), pp. 1-7, 2017

[19] Mouli, V.R.; Jevitha, K. Web Services Attacks and Security- A Systematic Literature Review. Procedia Comput. Sci. 2016, 93, 870–877.

[20] Oliveira, R.A.; Laranjeiro, N.; Vieira, M. Assessing the security of web service frameworks against Denial of Service attacks. J. Syst. Softw. 2015, 109, 18–31. Available online: https://www.sciencedirect.com/science/article/pii/S0164121215001454

[21] Abhishta; Joosten, R.; Nieuwenhuis, L.J.M. Analysing the Impact of a DDoS Attack Announcement on Victim Stock Prices. In Proceedings of the 2017 25th Euromicro International Conference on Parallel, Distributed and Network-Based Processing (PDP 2017), St. Petersburg, Russia, 6–8 March 2017; pp. 354–362.

[22] Subbulakshmi, T.; Balakrishnan, K.; Shalinie, S.M.; Anandkumar, D.; Ganapathisubramanian, V.; Kannathal, K. Detection of DDoS attacks using Enhanced Support Vector Machines with real time generated dataset. In Proceedings of the 3rd International Conference on Advanced Computing, ICoAC 2011, Chennai, India, 14–16 December 2011; pp. 17–22.

[23] Gupta, B.; Joshi, R.C.; Misra, M. Defending against Distributed Denial of Service Attacks: Issues and Challenges. Inf. Secur. J. A Glob. Perspect. 2009, 18, 224–247.

[24] Samtani, S.; Kantarcioglu, M.; Chen, H. Trailblazing the Artificial Intelligence for Cybersecurity Discipline. ACM Trans. Manag. Inf. Syst. 2020, 11, 1–19.

[25] Zarpelão, B.B.; Miani, R.S.; Kawakani, C.T.; de Alvarenga, S.C. A survey of intrusion detection in Internet of Things. J. Netw. Comput. Appl. 2017, 84, 25–37.

[26] Liu, C.; Yang, J.; Chen, R.; Zhang, Y.; Zeng, J. Research on immunity-based intrusion detection technology for the Internet of Things. In Proceedings of the 2011 7th International Conference on Natural Computation, ICNC 2011, Shanghai, China, 26–28 July 2011; Volume 1, pp. 212–216.

[27] Kasinathan, P.; Pastrone, C.; Spirito, M.A.; Vinkovits, M. Denial-of-Service detection in 6LoWPAN based Internet of Things. In Proceedings of the International Conference on Wireless and Mobile Computing, Networking and Communications, Lyon, France, 7–9 October 2013; pp. 600–607.