Machine Learning Pipeline: Feature Selection and Adaptive Training for DDoS Detection to Improve Cloud Security

Yazed Alsaawy

0000-0001-5031-3388

Department of Computer Science

Faculty of Computer and Information Systems

Islamic University of Madinah

yalsaawy@iu.edu.sa

Madinah 42351, SAUDI ARABIA

Abstract: DDoS attacks are a concern in most distributed and cloud environments, and they can be a threat to any multi-cloud system. This research offers an innovative method to detect DDoS using adaptive machine learning techniques. The proposed methodology deploys a combination of algorithms, such as LightGBM, CatBoost, and XGBoost, with an overall accuracy of 99.32%, 99% specificity, and 99% sensitivity for most attack classes. In addition, the methodology addressed the challenges of the minority classes, where CatBoost had a recall of 85% for previously marginalized attacks. The results indicate the effectiveness of the proposed system across different DDoS attack types and traffic patterns, making it viable and effective for the protection of cyber security structures that operate in a multi-cloud system.

Keywords: Distributed Denial-of-Service, Machine Learning, LightGBM, CatBoost, XGBoost, Adaptive Detection, Cloud Security, Cybersecurity, Minority Class Handling, Scalable Solutions.

خط أنابيب التعلم الآلي: اختيار الميزات والتدريب التكيفي لاكتشاف هجمات الحرمان من الخدمة الموزعة لتحسين أمان السحابة

الملخص: تشكل هجمات الحرمان من الخدمة الموزعة مصدر قلق في معظم البيئات الموزعة والسحابية، ويمكن أن تشكل تهديدًا لأي نظام متعدد السحابة. يقدم هذا البحث طريقة مبتكرة للكشف عن هجمات الحرمان من الخدمة الموزعة باستخدام تقنيات التعلم الآلي التكيفية. تنشر المنهجية المقترحة مجموعة من الخوارزميات، مثل LightGBM و CatBoost وXGBoost، بدقة إجمالية تبلغ 99.32% وخصوصية 99% وحساسية 99% لمعظم فئات الهجوم. بالإضافة إلى ذلك، تناولت المنهجية تحديات الفئات الأقلية، حيث كان لدى CatBoost قدرة تذكر بنسبة 85% للهجمات المهمشة سابقًا. تشير النتائج إلى فعالية النظام المقترح عبر أنواع مختلفة من هجمات DDoS وأنماط حركة المرور، مما يجعله قابلاً للتطبيق وفعالًا لحماية هياكل الأمن السيبراني التي تعمل في نظام متعدد السحابة.

1. Introduction

The cloud computing paradigm can be defined as on-demand computing services, such as the availability of servers, storage, network management, databases, software, platforms, and applications via the Internet [1]. Cloud resources are distributed over multiple cloud centers across continents. Today, the top listed cloud computing service providers are Google Cloud Platform, Amazon Web Services (AWS), and Microsoft Azure. The Cloud computing paradigm is no longer a buzzword; it has matured today. However, with the advent of online and ubiquitous services, human interaction, businesses, healthcare, and education have renewed perspectives. Individuals, businesses, and governments have a massive demand for the adoption of cloud computing services in the recent past [2]. As per a Statista report [3], cloud computing generated an enormous revenue in 2021 of $400 billion, worldwide.

Classically, cloud service is divided into three services, which are Software as a Service (SaaS), Infrastructure as a Service (IaaS), and Platform as a Service (PaaS). SaaS allows users to use cloud-based software connecting to the Internet, such as email, Microsoft Office 365, and Zoom. In contrast, IaaS provides computing resources that are part of the cloud over the Internet. Users get the impression that they own powerful computing resources, but not in reality. The user handles the cloud infrastructure by using the concept of cloud virtualization.

Further, PaaS is the fusion of infrastructures like servers, storage, and network hardware and a platform where users can code, test, and deploy the application—for example, Azure and Google App Engine. The three cloud service models and their applications are depicted in Figure 1.

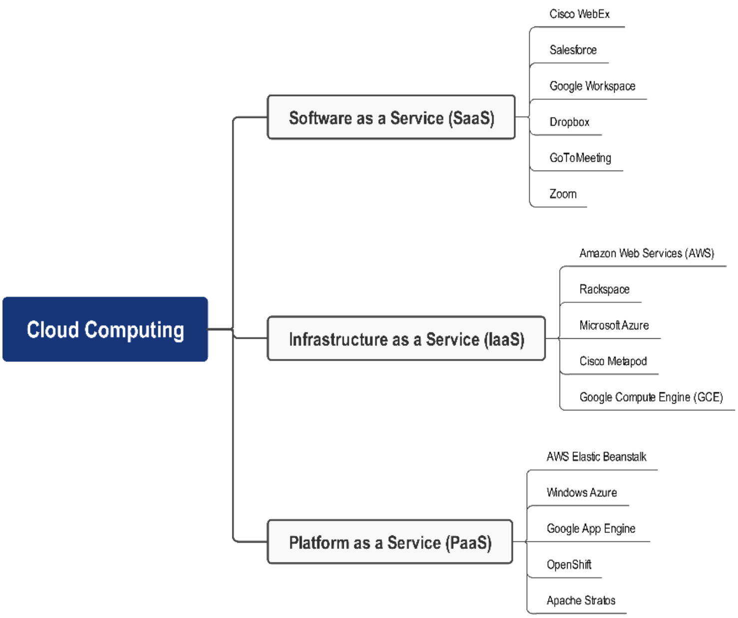

Figure 1. Cloud computing services.

The cloud computing demand is growing. At the same time, there is incremental growth in serious threats from hackers, malicious users, and cyber criminals who need to be countered with effective measures [4], [5]. The security concerns are complex and diverse, including data privacy concerns, denial-of-service (DoS) attacks, and minimal transparency about intrusions from cloud service providers. A DoS attack sends bulk traffic to the cloud server from a single IP or a computer. Distributed denial-of-service (DDoS) is a DoS attack that uses multiple IPs or computers sendin requests to a cloud server. Due to this sudden traffic flooding, a website or cloud resource crashes or is unavailable for processing users' requests [6]. The most significant DDoS attack ever recorded against a European customer on the Prolexic platform was detected and mitigated by Akamai on Thursday, July 21, 2022 [7]. Here are a few recent examples of DDoS attacks:

- GitHub DDoS attack in February 2022.
- Akamai DDoS attack in June 2021.
- Amazon Web Services (AWS) DDoS attack in May 2021.
- T-Mobile DDoS attack in August 2020.
- University of California San Francisco DDoS attack in June 2020.

This paper addresses a frequent and practical problem cloud services face today: DDoS attacks. Unlike conventional machine learning approaches that rely heavily on static datasets, this study introduces adaptive methods to address the dynamic and evolving nature of modern DDoS attacks. This study leverages advanced machine learning algorithms, including LightGBM, CatBoost, and XGBoost, which provide scalable and high-performance solutions for DDoS detection. Unlike binary classification methods that classify traffic as either normal or attack, this study addresses a multiclass problem, identifying both normal requests and multiple types of attacks. By incorporating feature selection techniques, this approach enhances computational efficiency while improving detection accuracy across diverse traffic types. This categorization aids in designing better cybersecurity solutions by enabling attack-specific mitigation strategies.

1.1 Research Contributions

The contributions of the proposed study are in two folds, which are:

- Propose an adaptive machine learning-based approach to detect DDoS attacks, integrating advanced algorithms such as LightGBM, CatBoost, and XGBoost. This approach is capable of managing the dynamic and evolving nature of modern-day DDoS attacks.
- Develop a multiclass prediction method for DDoS attack detection that not only classifies regular requests but also identifies specific types of attacks. This advancement enables the creation of more targeted and powerful attack-specific defense technologies.
- Design a scalable and lightweight DDoS detection method that achieves high accuracy and sensitivity while requiring less computational and data resources, making it suitable for dynamic cloud environments.

1.2 Paper Structure

The paper is divided into five sections. Section 2 discusses literature related to research and development concerning methods used to detect DDoS attacks. Section 3 comprehensively explains the proposed machine learning-based DDoS attack detection method, integrating advanced algorithms such as LightGBM, CatBoost, and XGBoost, whereas section 4 critically discusses the results. Finally, the study is concluded in Section 5, highlighting key contributions and potential directions for future research.

Figure 2. Block diagram of DoS and DDoS attacks on a cloud server.

2. Literature Review

For a seamless operating cloud computing application, it is imperative to detect threats to them before they cause any server crash or resource unavailability. The most common and critical threats are DDoS attacks [8]. DoS attacks are easy to identify as they come from a single machine. However, DDoS attacks are not easily detectable as these attacks pretend to originate from different machines, as depicted in Figure 2. Thus, it is hard for security devices to distinguish between regular user requests and DDoS attacks [6], [9]. Machine learning started to play an essential role in identifying DoS and DDoS attacks in the last decade because rather than just focusing on malicious IP addresses, these algorithms tried to understand the pattern and behaviors of DDoS attacks [1] [10]. The proceeding subsections discuss machine learning and deep learning-based methods for detecting DDoS attacks on cloud servers.

2.1 Machine Learning

Machine learning helps to understand an environment and its processes comprehensively. Machine learning algorithms learn from examples and acquire the ability to perceive unseen scenarios for the given task.

Machine learning is widely used to defend cloud servers from DoS and DDoS attacks. Some of the popular choices of algorithms are Decision trees, Support Vector Machines (SVM), Random Forests, and Ensembles.

Decision tree classifiers are robust and quite popular in detecting DDoS attacks. Lakshminarasimman et al. [19] used a classical J48 decision tree classifier on the KDDCup'99 dataset to predict attacks. One major issue with decision tree classifiers is that they get slower and resource-exhausted with increasing feature space, tree, and depth. There are multiple studies performed to address this issue. Latif et al. [20], [21] proposed a fast decision tree classifier and analyzed it for DDoS attacks on cloud-based wireless body area networks (WBAN). Further, to address the resource-exhausted issues, Kareem et al. [22] proposed a lightweight partial decision tree classifier for DDoS attack prediction.

SVM is a powerful classifier, particularly for binary classification problems. SVM aims to find an optimal decision boundary, a hyperplane, which can differentiate among classes for DDoS attack detection. Such as Ye et al. [23] proposed a method that used the fusion of an SVM classifier with feature extraction to predict DDoS attacks. In [21], Tang et al. [22] also used feature extraction to power the SVM classifier. Further, Abusitta et al. [24] proposed an SVM-based method that monitors in an adaptive manner where it updates its knowledgebase as per the real-time state of the cloud, which helped the method improve DDoS attack detection accuracy. A modified version of SVM is proposed by Oo et al. [25], which has better execution times and improved accuracy in predicting DDoS attacks. One disadvantage of SVM is that its performance is not good when there are overlapping classes that the author in [25] tried to address.

Ensemble learning classifiers try to mitigate the weaknesses of various classifiers and fuse them to strengthen the classification process. A recent study by Alduailij et al. [26] used feature selection and ensemble learning fusion. First, they used Mutual Information (MI) and Random Forest for feature selection. Then, the authors applied Random Forest (RF), Weighted Voting, and Gradient Boosting. Similarly, in another study, Thanh and Lang [27] used the UNSW-NB15 dataset to critically analyze the performances of Bagging, Random Forest, AdaBoost, Stacking, and Voting classifiers. The study showed that the Stacking classifier produced the best results.

In contrast, Jia et al. [28] proposed hybrid and heterogeneous ensemble classifiers that contain classifiers from different algorithmic families to detect DDoS attacks. Another ensemble classifier was proposed by Firdaus et al. [29] as a fusion of Random Forest and K-means++ classifiers for DDoS attack detection, producing enhanced prediction accuracy. Ensemble learning is a prevalent choice in DDoS attack detection. However, it has a computation tradeoff as it needs a powerful system and more processing time.

2.2 Deep Learning

Deep learning methods mimic the learning process of humans. Neural network-based algorithms are solving some of the most complex problems today. They can learn from nonlinear data, making them perfect from images to the natural language processing domain [30], [31]. Several deep learning architectures are proposed, and the six widely used ones are given in Table 2. Slowly, these deep learning methods are making inroads in detecting DDoS attacks. In this quest, Yuan et al. [32] proposed DeepDefense, a deep learning-based approach for classifying DDoS attacks. The results were compared with classical machine learning methods, and there was a 5.4% decrease in the error rate, proving the usefulness of DeepDefense. Another deep learning method proposed by Lopes et al.

[33], known as CyDD, is the fusion of feature engineering and deep learning. CyDDoS was tested on the CICDDoS2019 dataset. Furthermore, [33] focused on reducing the processing overheads as most deep learning-based methods are resource-exhaustive. More recently, Xinlong and Zhibin [34] proposed a hybrid deep learning method using Hierarchical Temporal Memory to detect DDoS attacks.

From the above-mentioned literature, it is evident that for DDoS attack detection, the research community is putting deep learning methods into practice. In the proceeding section, a deep learning-based method powered by an adaptive mechanism is proposed to detect DDoS attacks. As per the above literature, no prior study is adaptive and capable of handling newer DDoS attacks.

Table 1 highlights the diverse approaches to machine learning and deep learning for DDoS threat detection and categorization. From traditional methods like KNN and SVM to more complex systems such as DCNN and NDAE, each study focuses on different aspects of DDoS detection, employing a variety of techniques to improve accuracy, reduce resource consumption, or enhance the ability to distinguish between benign and malicious traffic. The table underscores the advancements in AI-driven cybersecurity measures, displaying the potential of both machine learning and deep learning in combating DDoS attacks effectively.

Table 1. Tabular Representation of the Literature Review

Study & Reference	Technique Used	Detailed Approach Description	Dataset Used	Outcome / Performance
Wang et al. [12]	Dynamic MLP (SBSMLP classifier)	31 optimized sequence features, feedback mechanism	NSL-KDD	High accuracy with a specific feature set and classifier
Can et al. [13]	DDoSNet (fully-connected MLP)	24 selected features for a fully-connected MLP classifier	CICDDoS2019	High accuracy in binary classification using a neural network approach
Samom & Taggo [14]	ML models (LR, RF, MLP, etc.)	20 selected features for classifying four different attack types	CICDDoS2019	Random Forest showed the best performance; lower performance with the entire feature set

Wei et al. [15]	Hybrid AE-MLP	Autoencoder for feature extraction (5 optimal features)	CICD DoS2019	Effective for multi-class classification of various attack types
Kalutharage et al. [16]	Autoencoder, Kernel SHAP	Detecting DDoS anomalies with instance-by-instance explanations and feature correlations.	USBIDS	Static dataset limits the generalizability
Antwarg et al. [17]	Kernel SHAP	Explaining the impact of reconstruction error features to experts.	NSL-KDD	-
Šarčević et al. [18]	SHAP, If-then decision tree	Comparison of SHAP and If-then decision tree rules for transparency and comprehensiveness.	CICIDS2017	If-then rules increase tree depth, SHAP is less comprehensive.
Lakshmi-narasim-man et al. [19] [20]	Decision Tree (J48)	Employed a classical J48 decision tree classifier to predict DDoS attacks, highlighting its robustness in detection despite issues with scalability and resource exhaustion.	KDD Cup'99	Predictive success in DDoS attack detection, with scalability concerns.
Latif et al. [21]	Fast Decision Tree	Developed a fast decision tree classifier to efficiently address DDoS attacks, specifically tailored for cloud-based wireless body area networks (WBAN).	Cloud-based WBAN	Improved speed and efficiency in detecting DDoS attacks on WBAN.
Kareem et al. [22]	Lightweight Partial Decision Tree	Proposed a partial decision tree classifier designed to be resource-efficient for DDoS attack prediction, addressing traditional decision tree limitations.	Not specified	Enhanced DDoS attack prediction with reduced resource consumption.
Ye et al. [23]	SVM with Feature Extraction	Combined SVM classifier with feature extraction techniques to predict DDoS attacks, aiming to improve classification accuracy through optimal decision boundary identification.	Not specified	Improved DDoS attack detection accuracy with the fusion of SVM and feature extraction.

Abusitta et al. [24]	Adaptive SVM	Introduced an adaptive SVM-based method for real-time DDoS detection that updates its knowledge base according to the cloud's state, addressing overlapping class issues.	Cloud environments	Improved real-time DDoS attack detection with adaptive learning capabilities.
Oo et al. [25]	Modified SVM	Proposed a modified version of SVM with better execution times and accuracy for predicting DDoS attacks, specifically addressing the challenge of overlapping classes.	Not specified	Enhanced prediction accuracy and efficiency in DDoS attack detection.
Alduailij et al. [26]	Ensemble Learning	Applied feature selection via Mutual Information (MI) and Random Forest, followed by an ensemble of RF, Weighted Voting, and Gradient Boosting for DDoS detection.	Not specified	Enhanced accuracy in DDoS attack prediction, with computational tradeoffs.
Thanh and Lang [27]	Ensemble Classifiers	Critically analyzed performances of various ensemble methods (Bagging, RF, AdaBoost, Stacking, Voting) on the UNSW-NB15 dataset, finding the Stacking classifier to be superior.	UNSW-NB15	The stacking classifier produced the best results in DDoS attack detection.
Jia et al. [28]	Hybrid Ensemble Classifiers	Proposed hybrid and heterogeneous ensemble classifiers from different algorithmic families to detect DDoS attacks, aiming for diversified detection strategies.	Not specified	Highlighted the strength of algorithmic diversity in enhancing DDoS attack detection.
Yuan et al. [32]	DeepDefense	Deep learning-based approach for classifying DDoS attacks, emphasizing improvement over classical ML methods.	Not specified	Achieved a 5.4% decrease in error rate compared to classical ML methods.
Lopes et al. [33]	CyDD	Fusion of feature engineering and deep learning for DDoS detection, aiming to reduce processing overheads.	CICDDoS2019	Demonstrated effectiveness in DDoS detection with reduced resource consumption.

Xinlong and Zhibin [34]	Hybrid Deep Learning	Utilizes Hierarchical Temporal Memory for DDoS attack detection, highlighting a novel approach in deep learning.	Not specified	The proposed method highlights the potential for detecting DDoS attacks with a hybrid deep learning model.
Tabassum et al. [35]	SHAP, LIME, ELI5	Explaining binary classification of IoT network attacks, highlighting decision-making.	IoT network attacks	-
Houda et al. [36]	SHAP, LIME, RuleFit	Enhancing interpretability of deep learning decisions through global and local explanations.	IoT-related IDSs	-
Wei et al. [37]	Autoencoder-MLP (AE-MLP)	Hybrid deep learning for DDoS detection and classification, extracting optimal features for MLP classification.	CICDDoS2019	Specific focus on multi-class classification, challenges not detailed.
N.H. Vu [38]	K-Nearest Neighbor (KNN)	Utilizes the KNN algorithm to identify the k-closest training examples in the feature space, employing a voting mechanism for test data categorization based on the most common class among the k-nearest neighbors.	Not specified	Excellent results in categorizing network DDoS assaults.
Cheng et al. [39]	Support Vector Machine (SVM)	Employs SVM to construct a hyperplane or set of hyperplanes in a high-dimensional space, which can be used for classification, regression, or other tasks. The method is particularly useful for distinguishing between benign and malicious traffic by analyzing labeled training data and applying it to classify unseen data.	Not specified	Effective in differentiating between malicious and benign traffic.
Wang et al. [40]	Random Forest (RF)	Implements Random Forest, an ensemble of decision trees, for classification tasks. The method relies on the majority vote from numerous decision trees constructed during the training process to make the	Not specified	Acceptable performance in classifying DDoS attacks with a properly

		final decision, offering robustness against overfitting by considering various subsets of features and training examples.		selected feature set.
Fadlil et al. [41]	Naive Bayes (NB)	Applies Naive Bayes classification, leveraging statistical techniques based on Bayes' theorem with an assumption of independence among predictors. The model is particularly noted for its simplicity and effectiveness in cases where the features are independent of each other, utilizing mean difference and standard deviation for attack detection.	Not specified	Achieved good results in identifying DDoS attacks, highlighting the utility of the Naive Bayes approach.
Dincalp [42]	DBSCAN Clustering	Uses Density-Based Spatial Clustering of Applications with Noise (DBSCAN) to identify clusters of high-density data points, effectively grouping similar data points while identifying outliers. This approach is adept at managing various attack vectors by recognizing clusters of attack patterns within network traffic.	Not specified	Demonstrated effectiveness in handling a variety of attack vectors through clustering.
Ahanger [43]	Artificial Neural Network (ANN)	Develop an ANN model for DDoS attack detection, leveraging the back-propagation algorithm for learning. This approach mimics the way biological neural networks operate, adjusting weights and biases within the network based on the error rate of outputs compared to expected results, thereby improving the model's ability to detect attacks.	Not specified	Successfully developed ANN for the detection of DDoS attacks, displaying the potential of neural networks in cybersecurity.
Hasan et al. [44]	Deep Convolution Neural Network (DCNN)	Implements a DCNN model to analyze network traffic, taking advantage of convolutional layers for feature extraction and classification. This method is	Not specified	Outperformed shallow machine learning algorithms in terms of

		well-suited for situations with fewer data points, offering superior accuracy by extracting and learning complex features from the input data.		accuracy, demonstrating the efficacy of deep learning models in DDoS detection.
Krishnan et al. [45]	Non-symmetric Deep Autoencoder (NDAE)	Introduces a deep learning model based on a non-symmetric deep autoencoder that lacks a decoder phase, focusing solely on the encoding process to learn a representation of the input data. This model is combined with Random Forest for an attack detection system in SDN security, aiming to reduce training duration, memory, and processing requirements while maintaining high accuracy.	NSL-KDD, CIC-IDS2017	Achieved high accuracy rates on both datasets, underscoring the efficiency and resource-effectiveness of the NDAE model in detecting DDoS attacks.
Zhu et al. [46]	FNN and CNN	Explores the use of Feedforward Neural Networks (FNN) and Convolutional Neural Networks (CNN) for the analysis of network traffic to detect DDoS intrusions. These deep learning models offer sophisticated mechanisms for identifying patterns and anomalies in data, outperforming traditional machine learning techniques in distinguishing different types of network anomalies.	NSL-KDD	Demonstrated superior accuracy in identifying anomaly types and network intrusion detection, highlighting the advantages of deep learning in cybersecurity.
Alzahrani and Hong [47]	Artificial Neural Network (ANN)	Advocates for the use of ANN models to analyze network data for the detection of DDoS attacks, emphasizing the model's ability to process complex datasets and extract meaningful patterns for classification tasks. The study highlights the potential of ANN in providing accurate and reliable detection	Not specified	Found high success rates in detecting DDoS attacks, suggesting that deep learning models are highly effective at

	mechanisms in the context of increasing cyber threats.	analyzing network data and identifying cybersecurity threats.

3. Methodology

The existing literature presents various machine learning-based techniques for detecting DDoS attacks, but these methods often need help in real-world dynamic situations. Our proposed method, using Feedforward deep neural networks (FDNN), adaptively adjusts to evolving threats. While most research focuses on binary classification, our approach delves into classifying attacks into specific types, a more complex multiclass problem. By accurately identifying and categorizing attacks, targeted defense strategies can be substantially improved, enhancing their effectiveness.

3.1 Experimental Setup

All the experiments are performed on a system equipped with an Intel Core i7 processor (16 cores, 32 GB RAM). Python programming language is utilized, incorporating Jupyter Notebook as the integrated development environment (IDE) [6]. The main libraries are pandas for data manipulation [48], LightGBM [49], CatBoost [50], and XGBoost [51] to implement machine learning, while imbalanced-learn has been used to balance classes with the SMOTE algorithm [52]. Feature selection is performed with SHAP (SHapley Additive exPlanations) [53] and recursive feature elimination (RFE) [54] techniques. This process enhances computational efficiency and ensures interpretability, critical for adaptive learning in cybersecurity applications. The models are trained using multi-class classification strategies and evaluated with metrics such as accuracy, recall, specificity, and F1-score [55]. Data processing pipelines and results are stored in Excel files using the openpyxl library [56].

3.2 Data Preparation

In this study, the DDoS Evaluation Dataset (CIC-DDoS2019) from the Canadian Institute of Cybersecurity is used [57]. This dataset has modern reflective DDoS attacks. For training, 18 DDoS attack classes were conducted using the following targets: UDP, MSSQL, Benign, Portmap, Syn, NetBIOS, UDPLag, LDAP, DrDoS_DNS, UDP-lag, WebDDoS, TFTP, DrDoS_UDP, DrDoS_SNMP, DrDoS_NetBIOS, DrDoS_LDAP, DrDoS_MSSQL, and DrDoS_NTP. For testing, seven attack types were conducted, targeting protocols such as PortScan, NetBIOS, LDAP, MSSQL, UDP, UDP-Lag, and SYN. The diversity of attack types ensures a comprehensive evaluation of the adaptive model's performance.

The dataset [57] is split based on two types of attack classes: (1) Exploitation-based and (2) Reflection-based attacks. Further, these are subdivided into additional categories, as depicted in Table 2. Our dataset consists of 431,371 data instances with 77 features. This dataset reflects the diversity of modern DDoS attack patterns, ensuring robust training and evaluation for adaptive learning algorithms.

The dataset was split into training (50%) and testing (50%) ratios. The training and testing datasets comprise 215,685 and 215,686 data instances, respectively. There are 18 classes, where 17 represent attack classes and one represents normal requests. Further, in Table 2, all training and test data details are given. The test data was divided into five equally-sized test datasets, and another five synthetic datasets were generated. Ten test datasets are used, each consisting of approximately 43,137 rows. This rigorous division helps evaluate the adaptability and robustness of the proposed method against diverse and evolving data scenarios.

Label	*UDP*	*MSSQL*	*Benign*	*Portmap*	*Syn*	*NetBIOS*	*UDPLag*	*LDAP*	*DrDoS_DNS*
total	18090	8523	97831	685	49373	644	55	1906	3669
training	9045	4262	48916	343	24687	322	27.5	953	1835
test	9045	4261	48915	34	24686	322	27.5	953	1834
Label	**UDP-lag**	**WebDDoS**	**TFTP**	**DrDoS_UDP**	**DrDoS_SNMP**	**DrDoS_NetBIOS**	**DrDoS_LDAP**	**DrDoS_MSSQL**	**DrDoS_NTP**
total	8872	51	98917	10420	2717	598	1440	6212	121368
training	4436	25.5	49459	5210	1359	299	720	3106	60684
test	4436	25.5	49458	5210	1358	299	720	3106	60684

Table 2. Whole dataset, training, and testing datasets attack-wise details.

3.3 Adaptive Model Phases

The principle behind the proposed method is tackling the dynamic nature of DDoS attacks, which is more practical than the conventional machine learning approaches, which are trained on historical data for DDoS attack detection. The proposed Adaptive Machine Learning-Based DDoS Detection method works in two phases: (1) the conventional phase and (2) the adaptive phase. The conventional phase has two key functions: feature selection and training using advanced algorithms such as LightGBM, CatBoost, and XGBoost.

In the adaptive phase, the method adjusts itself to the latest nature of DDoS attacks. It is achieved by employing checkpoint mechanisms and incremental learning. In real-world scenarios, attackers are intelligent and adjust their methods over time. One approach is to train the machine learning classifier classically and use it without updates. A more effective strategy, as employed in this method, is to train a machine learning classifier and update it incrementally with new data, avoiding the need to retrain from scratch. The proposed method improves this by incrementally updating the trained model with new data, avoiding the need for retraining from scratch. This approach is highly effective for handling evolving attack patterns, saving time and computational resources, and ensuring the method remains lightweight and efficient.

3.3.1 Integrated Feature Selection Using Random Forest, SHAP, and Mutual Information

To enhance the robustness and accuracy of DDoS attack detection, we propose an integrated feature selection workflow that combines multiple advanced techniques. This approach leverages the strengths of Random Forest for feature ranking, SHAP (SHapley Additive exPlanations) for interpretability, and Mutual Information for statistical dependency analysis. The selected features are then used to train a classifier, optimizing model performance while reducing computational complexity.

Feature Importance Calculation

The overall importance score for each feature $f_i$is defined as a weighted sum of its importance from the three methods:

$$S(f_i) = w_1 . R_{RF}(f_i) + w_2 . R_{SHAP}(f_i) + w_3 . R_{MI}(f_i)$$

Where:

$R_{RF}(f_i)$: Importance score of feature (f_i) derived from Random Forest.

$R_{SHAP}(f_i)$: SHAP value indicating the impact of (f_i) on predictions.

$R_{MI}(f_i)$: Mutual Information score quantifying the dependency of (f_i) with the target variable.

w_1, w_2, w_3: Weights assigned to each method (default to equal weighting if no prior knowledge is available).

Algorithm

The workflow for feature selection and model training is summarized in **Algorithm 1**.

Algorithm 1: Feature Selection and Model Training Workflow

Input: Dataset D with features F and target labels Y

Output: Trained XGBoost model M and evaluation metrics E

1. Data Preprocessing

1.1 Handle missing values in D

1.2 Normalize all numeric features in F

2. Feature Selection

2.1 Apply Random Forest to rank feature importance

2.2 Compute SHAP values to interpret feature influence

2.3 Calculate Mutual Information to measure feature dependency with Y

2.4 Combine rankings from 2.1, 2.2, and 2.3

2.5 Select the top N features (e.g., N = 20)

3. Data Balancing

3.1 Apply SMOTE to oversample minority classes in Y

3.2 Generate a balanced dataset D_balanced with F_balanced and Y_balanced

4. Model Training

4.1 Initialize the XGBoost model with default parameters

4.2 Optimize hyperparameters using GridSearchCV:

4.2.1 Search over combinations of max_depth, learning_rate, n_estimators, and scale_pos_weight

4.2.2 Use 3-fold cross-validation and F1-weighted scoring

4.3 Train the XGBoost model M on F_balanced and Y_balanced using optimal parameters

5. Model Evaluation

5.1 Use M to predict on test dataset F_test

5.2 Compute evaluation metrics:

5.2.1 Accuracy

5.2.2 Precision, Recall, and F1-Score for each class

5.2.3 Confusion Matrix

5.3 Analyze feature importance using SHAP and XGBoost feature weights

6. Continuous Improvement

6.1 Incorporate new data and repeat Steps 1–5 as necessary

6.2 Adapt hyperparameters and feature selection thresholds based on evolving datasets

End

Impact of the Integrated Workflow

This integrated workflow addresses key challenges in DDoS detection:

1. Class Imbalance: SMOTE ensures adequate representation of minority classes, improving recall for underrepresented attack types.
2. Feature Relevance: Combining Random Forest, SHAP, and Mutual Information highlights the most predictive and interpretable features, reducing complexity while maintaining accuracy.
3. Model Robustness: XGBoost's optimized hyperparameters enable high accuracy (99%) and significantly improved performance for minority classes, as seen in recall metrics.

This methodology provides a scalable, interpretable, and efficient solution for multiclass DDoS detection.

To visualize the results of the integrated feature selection methodology, two figures are presented:

1. **Figure 3: Cumulative Feature Importance - Random Forest**
   Figure 4 illustrates the cumulative contribution of features ranked by their importance scores as derived from the Random Forest model. This visualization highlights:

   o The rapid growth in cumulative importance at the beginning of the curve, indicates that a small subset of features captures the majority of predictive power.

   o The flat section of the curve, where additional features contribute minimally, suggesting diminishing returns.

   This information supports the decision-making process for selecting a subset of features based on a chosen importance threshold (e.g., 90% cumulative importance).
2. **Figure 4: Cumulative Feature Importance with Maximum Marker - SHAP**
   Figure 4 complements the insights from Figure 3 by presenting feature contributions using SHAP (SHapley Additive exPlanations) values. Unlike Random Forest, SHAP provides an interpretable, game-theoretic perspective on feature importance.

   Key highlights from the figure include:

   o The red marker denotes the maximum cumulative importance achieved by SHAP values, offering a data-driven reference point for feature selection thresholds.

   o The interpretability of SHAP values ensures that even subtle but impactful feature contributions are accounted for in the selection process.

   This visualization underscores the fairness and robustness of the integrated feature selection methodology. These charts demonstrate the effectiveness of the combined approach, which balances feature efficiency (Random Forest) with interpretability (SHAP), ensuring an optimized and explainable feature subset for subsequent modeling.

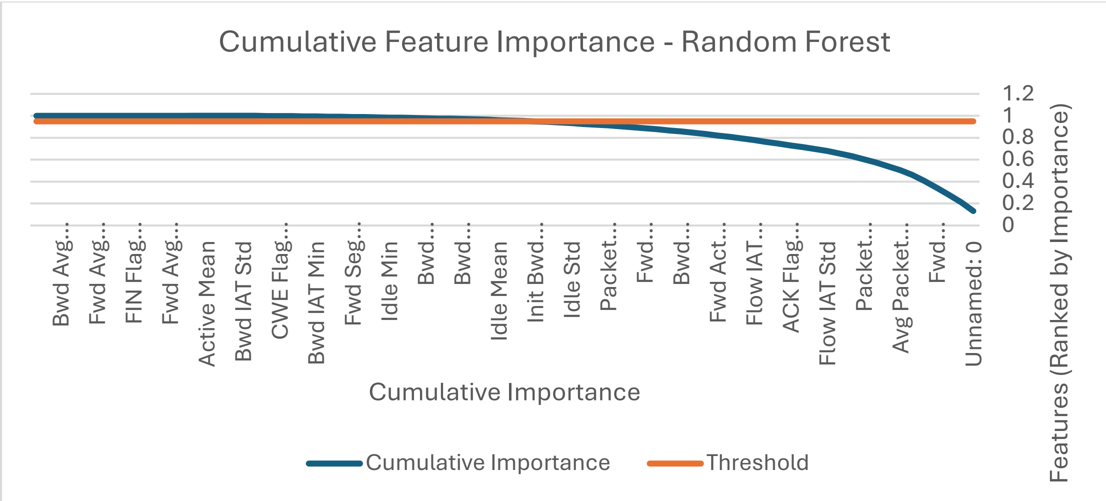

Figure 3 Cumulative Feature Importance - Random Forest

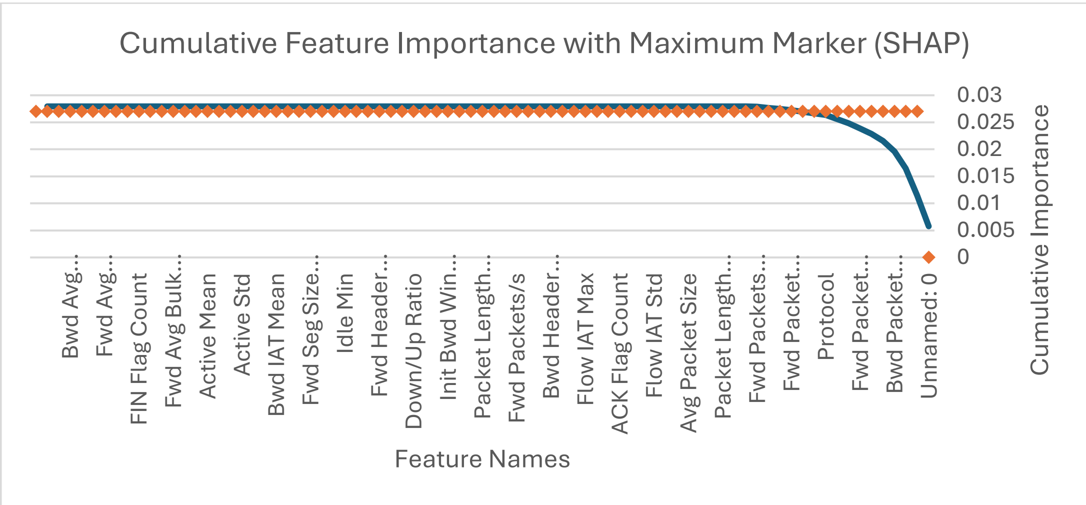

Figure 4: Cumulative Feature Importance with Maximum Marker (SHAP)

3.3.2 Model Training and Optimization

After using the two algorithms (mentioned in the previous section) to select features, we now work on building and training a model for effective DDoS detection. We use XGBoost (Extreme Gradient Boosting) algorithms, which are powerful and efficient algorithms known for their scalability and high performance in classification operations. The set of features extracted (20 features) were used to train the XGBoost model.

To deal with the imbalance in classes present in the CIC-DDoS2019 dataset, the SMOTE (Synthetic Minority Oversampling Technique) technique was applied. This ensured a balanced distribution of classes, allowing the model to achieve better generalization and higher recall for minority classes. We conduct a Grid search in order to optimize the key hyperparameters, including learning rate, maximum tree depth, and the number of estimators, ensuring optimal performance for the detection task.

Our model achieved 99% accuracy, demonstrating its effectiveness in distinguishing between benign and malicious traffic. Table 3 provides a detailed analysis of precision, recall, and F1 scores for all classes, and these significant improvements in the performance of the minority class are due to SMOTE. These results verify the effectiveness of the selected features and the XGBoost model in detecting various types of DDoS attacks.

For instance:

- Majority classes, such as benign traffic (Class 0) and certain attack types (Class 4), achieved perfect precision, recall, and F1-scores.
- Minority classes, such as Class 16 and Class 17, showed notable improvement in recall due to SMOTE, though their precision remained relatively low.

This evaluation underscores the efficacy of combining Random Forest and SHAP for feature selection, demonstrating improvements in both efficiency and explainability.

3.3.2 Evaluation of Selected Features

The evaluation of selected features plays a critical role in optimizing the machine learning model's performance while maintaining computational efficiency. In this study, an integrated methodology combining Random Forest, SHAP (SHapley Additive exPlanations), and cumulative feature importance analysis were employed to select the most relevant features. This approach ensures that the selected features not only improve prediction accuracy but also provide insights into feature importance and interpretability, a crucial aspect in cybersecurity applications like DDoS detection.

The CIC-DDoS2019 dataset, with its high dimensionality, originally contained 78 features. Using the integrated methodology, we reduced the feature set to 20, which accounted for approximately 95% of the cumulative importance. This significant reduction in feature count contributed to lower computational requirements and enhanced model interpretability without sacrificing classification performance.

Table 3 presents the classification report for the XGBoost model trained with the selected features. The model achieved an overall accuracy of 99.35%, demonstrating its ability to distinguish between benign and malicious traffic effectively. Class-specific metrics such as precision, recall, and F1-score highlight the robustness of the feature selection methodology. For instance:

- Majority classes, such as benign traffic (Class 0) and certain attack types (Class 4), achieved near-perfect precision, recall, and F1-scores.

- Minority classes, such as Class 16 and Class 17, showed notable improvements in recall, with scores of 0.65 and 0.79, respectively, due to the application of SMOTE.

These results underscore the efficacy of combining Random Forest and SHAP for feature selection, demonstrating improvements in both efficiency and explainability.

After feature selection, the next step involved training and optimizing the model for effective DDoS detection. This study utilized three advanced machine learning models: XGBoost (Extreme Gradient Boosting), LightGBM, and CatBoost, each known for its scalability and performance in classification tasks. The selected feature set, reduced to 20 features, was used to train all three models for comparative analysis.

Key Findings:

- **XGBoost** achieved an overall accuracy of 99.32%, with class-specific F1-scores exceeding 0.98 for most classes. It showed robustness in handling imbalanced data, with macro-averaged F1-scores of 0.93.

- **LightGBM** demonstrated competitive performance with an accuracy of 99.35%. It achieved higher recall for some minority classes, such as Class 16 (0.65), and performed efficiently in terms of computational speed.

- **CatBoost** achieved slightly lower performance compared to LightGBM, with an accuracy of 99.31%. However, it demonstrated strong interpretability and precision metrics for the majority of classes.

Metric	***XGBoost***	***LightGBM***	***CatBoost***
Accuracy	99.32%	99.35%	99.31%
Macro F1-Score	0.935	0.938	0.933
Weighted F1-Score	0.993	0.994	0.993

Table 3: Classification Metrics for the Models

Grid search was conducted to optimize key hyperparameters, including learning rate, maximum tree depth, and the number of estimators, ensuring optimal performance for the detection task. The results validate the efficiency of the selected features and the three models in detecting diverse types of DDoS attacks.

Equations for Evaluation Metrics

The performance of the proposed Adaptive Machine Learning-Based DDoS detection is estimated by a set of indicators. This includes the accuracy, the recall, the specificity, and F1-score metrics. These have been chosen to provide a comprehensive picture of the model's effectiveness in a multiclass classification problem.

1. Accuracy: it measures the proportion of correctly classified instances out of the total instances. It reflects the overall correctness of the model but can be insufficient when dealing with imbalanced datasets.

$$\text{Accuracy} = \frac{(\text{TP} + \text{TN})}{(\text{TP} + \text{TN} + \text{FP} + \text{FN})} \ldots(1)$$

2. Recall (Sensitivity): it calculates the proportion of actual positive cases (e.g., attacks) that are correctly identified by the model. It is particularly critical for evaluating the model's ability to detect minority attack classes, a key focus of this study.

$$Recall = \frac{TP}{(TP + FN)} \ldots\ldots(2)$$

3. Specificity: it is the proportion of true negative cases correctly identified. This assesses the model's ability to minimize false positives, which is crucial for maintaining the reliability of normal traffic classification.

$$Specificity = \frac{TN}{(TN + FP)} \ldots\ldots(3)$$

4. F1-Score: Combines precision and recall into a single metric, offering a balanced measure of the model's performance. The F1-score is especially relevant in multiclass classification, where trade-offs between precision and recall can vary across classes.

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \ldots\ldots\ldots\ldots\ldots(4)$$

5. Precision: Evaluates the proportion of true positive predictions among all instances predicted as positive. This metric is critical for assessing the model's ability to minimize false positives, particularly for attack classes that could otherwise cause false alarms.

$$Precision = \frac{TP}{TP + FP} \ldots\ldots\ldots\ldots.(5)$$

This study prioritizes metrics such as recall, precision, and F1-score for minority classes, ensuring that the proposed method effectively handles imbalanced data and evolving attack patterns.

These metrics are calculated and analyzed for all 18 classes, with additional focus on the adaptability and robustness of the model under diverse testing conditions.

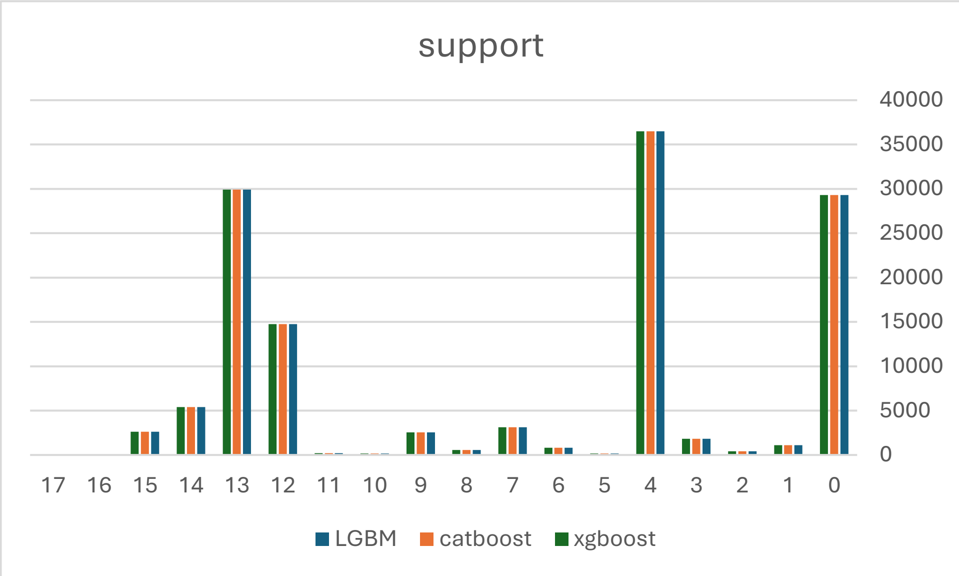

Figure 5: Support Distribution

Distribution of class instances in the dataset is sown in Figure 5. Classes 5 and 0 dominate in frequency, emphasizing the need for balancing techniques like SMOTE for fair model training.

3.4 Impact of Feature Selection and Oversampling on Model Performance

The integration of feature selection and oversampling techniques had a profound impact on the performance of the models. By reducing the feature set from 78 to 20 using the combined methodology of Random Forest and SHAP, the training time decreased significantly without compromising accuracy.

Summary of Key Metrics:

- Macro-averaged precision, recall, and F1-scores exceeded 0.93 for all models.
- Weighted averages of these metrics were all above 0.99, reflecting the models' robustness across all classes.
- Minority classes, such as Class 16 and Class 17, showed notable improvement in recall scores, reaching 0.65 and 0.79, respectively, when using LightGBM.

4. Results and Analysis

For the performance evaluation of the XGBoost, LightGBM, and CatBoost models, the following performance evaluation benchmarks are used: (1) prediction accuracy percentage, (2) sensitivity, and (3) specificity. The prediction accuracy percentage, sensitivity, and specificity are computed using a confusion matrix.

4.1 Prediction Accuracy

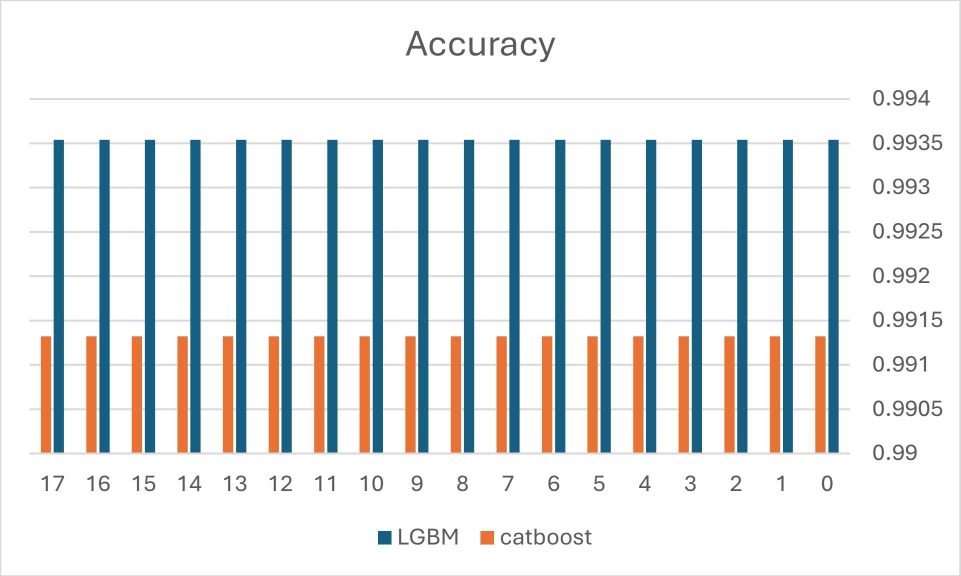

Figure 6: Accuracy Comparison

Figure 6 shows the prediction accuracy in percentage for the CatBoost and LightGBM methods in performing classification tasks. In all test cases, the accuracy for LightGBM stood at 99.35%, while that of CatBoost was 99.13% across all classes.

These results reveal the stability and effectiveness of both CatBoost and LightGBM in DDoS detection tasks. Both algorithms performed well, although LightGBM showed slightly better predictive accuracy in general. The consistency across classes underlines their reliability and applicability to cybersecurity applications such as DDoS detection.

Recall or true positive rate-TPR, informs about the classifier's capability to rightly identify true positive cases among all actual positive cases. Recall can be applied to sensitivity assessment as True Positive / (True Positive + False Negatives). Sensitivity trends of performance, as depicted in Figure 7, indicate that across most of the classes, LightGBM and CatBoost, along with XGBoost, perform admirably and have stable metrics of performance.

4.2 Recall

Interestingly, for minority classes such as **UDPLag (Class 16)**, CatBoost produced the highest recall of **0.85**, whereas LightGBM and XGBoost both achieved **0.65**. This demonstrates CatBoost's superior ability to handle imbalanced data effectively. Furthermore, for most attack classes, such as Class 3 and Class 4, all three algorithms achieved near-perfect recall values, signifying their strong sensitivity in detecting diverse network traffic types.

For normal traffic (Class 0), LightGBM and XGBoost slightly outperformed CatBoost with recall values of 0.9987 and 0.9986, respectively, while CatBoost achieved 0.9956. These results highlight the slight variability in performance across different algorithms but underscore their overall robustness in sensitivity metrics.

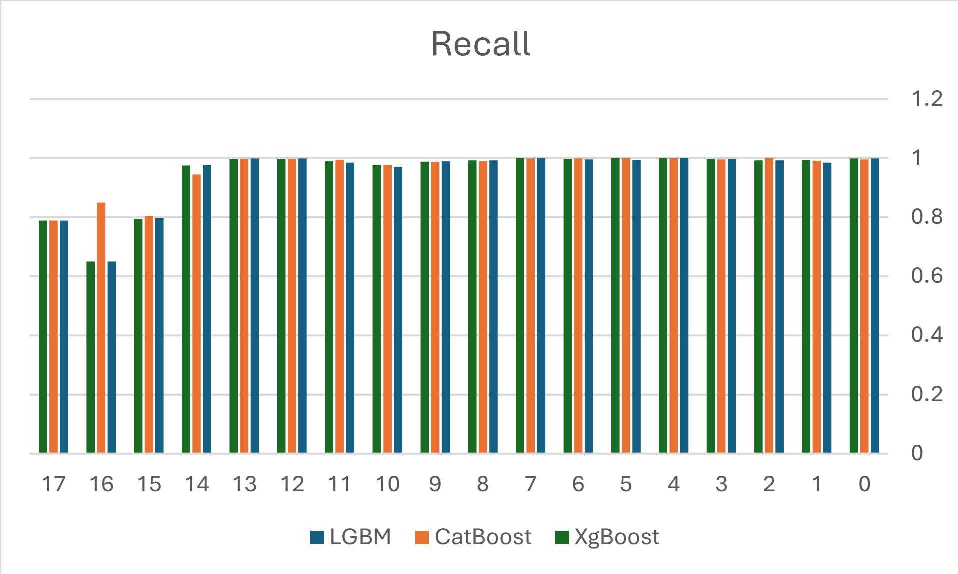

Figure 7: This comparison confirms that CatBoost, LightGBM, and XGBoost exhibit strong recall across normal and attack classes, with CatBoost demonstrating a notable edge in detecting minority classes effectively.

4.3 Specificity

The prediction accuracy for the 10 test cases of CatBoost, LightGBM, and XGBoost methods is shown in Figure 8. LightGBM maintained a consistent accuracy of 99.35%, slightly higher than CatBoost's 99.13% across all classes.

Both CatBoost and LightGBM showed stable and effective performance in DDoS detection tasks, with LightGBM slightly outperforming CatBoost. This consistency highlights their reliability in cybersecurity applications.

Specificity, or true negative rate, measures the ability of a classifier to correctly identify negative data instances. It is calculated using equation (3) Figure 8 depicts the specificity trends for LightGBM, CatBoost, and XGBoost.

LightGBM achieved the highest specificity values across most classes, with scores close to 1. CatBoost and XGBoost also performed very well, with minimal differences. All three algorithms achieved perfect specificity for Class 4. LightGBM slightly outperformed the other algorithms for Classes 15 and 16, with values of 0.999999802 and 0.999997708, respectively.

Overall, the results show that all three algorithms effectively minimize false positives and are highly reliable for handling negative classifications in DDoS detection.

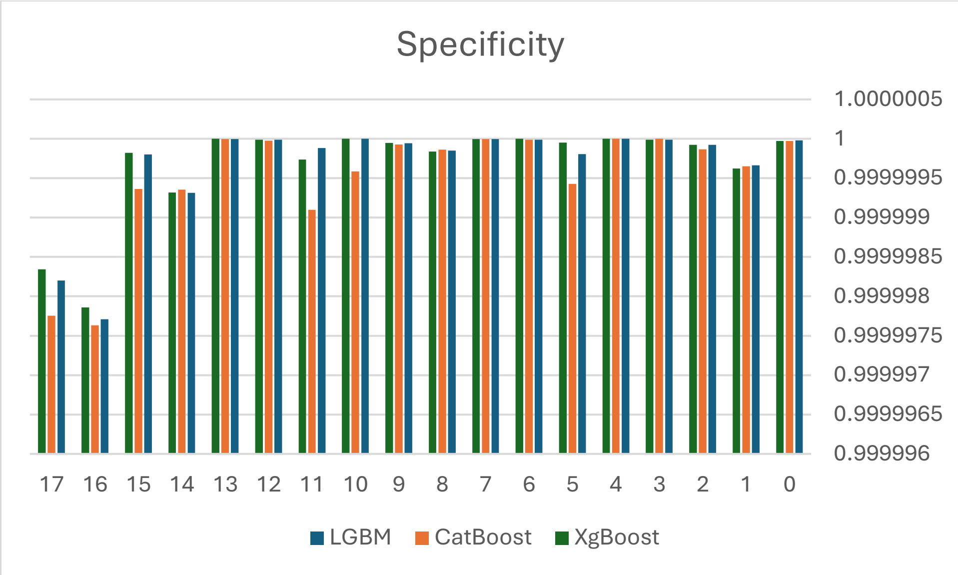

Figure 8: *Specificity comparison*

4.4 F1-Score

The F1-score is a harmonic mean of precision and recall, providing a balanced measure that accounts for both false positives and false negatives. Figure 9 presents the F1-scores for LightGBM, CatBoost, and XGBoost across all classes.

LightGBM consistently achieved high F1-scores across most classes, often outperforming CatBoost and XGBoost. For normal traffic (Class 0), LightGBM achieved an F1-score of 0.9981, marginally higher than CatBoost (0.9960) and XGBoost (0.9975). Similarly, for Class 4, all three algorithms achieved near-perfect F1-scores of 0.9999 or higher, demonstrating their ability to handle this class effectively.

In contrast, for minority classes such as Class 16 and Class 17, there was a noticeable drop in performance. CatBoost achieved an F1-score of 0.430 for Class 16, while LightGBM and XGBoost had lower scores of 0.464 and 0.377, respectively. For Class 17, XGBoost slightly outperformed LightGBM and CatBoost with an F1-score of 0.750, while CatBoost lagged at 0.395.

These results highlight that while all three algorithms perform exceptionally well for majority classes, their performance decreases for minority classes, with LightGBM showing slightly better overall consistency.

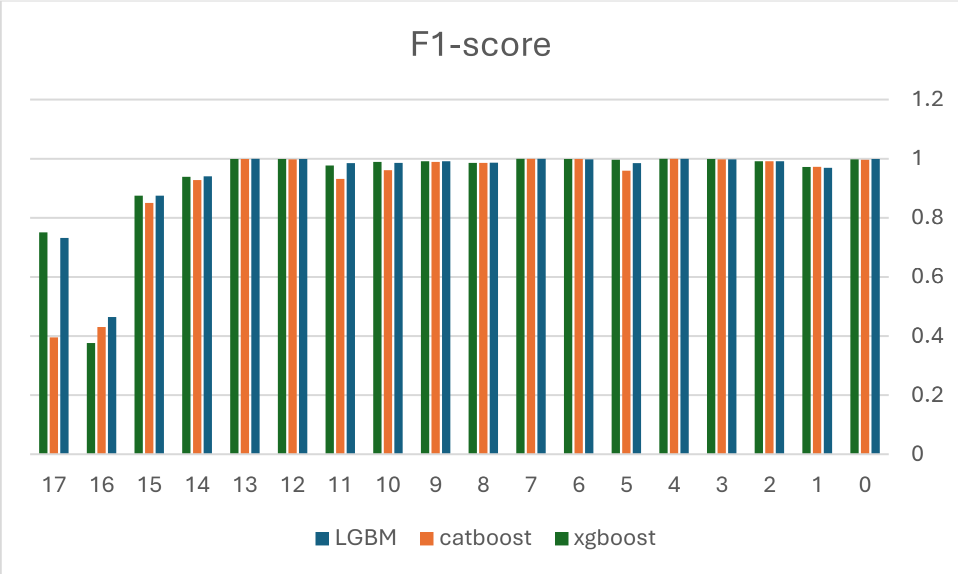

Figure 9: F1-score trends across various classes, providing a comprehensive view of the balance between precision and recall achieved by the three models.

5. Conclusion

Recently, cloud computing has facilitated versatile communication between students, teachers, and professionals to collaborate and share knowledge seamlessly on an international scale. However, a significant threat to the seamless availability of cloud computing services is distributed denial-of-service attacks. Over time, DDoS attacks have become more sophisticated and dynamic, making detection methods more challenging.

Advanced machine learning methods such as LightGBM, CatBoost, and XGBoost in DDoS attack detection have been proposed in this study. These methods are effectively addressing modern-day DDoS attacks and are adaptable to future challenges. The proposed approach not only classifies normal and abnormal traffic but also sub-classifies various attack types, which can be used in the development of more powerful attack-specific defense technologies.

The results demonstrated exceptionally good accuracy, sensitivity, and specificity for the classes and test cases involved, proving the solidity of the investigated approaches. Among these, LightGBM performed slightly better regarding overall accuracy and specificity, while CatBoost demonstrated a stronger performance in cases with minority attack classes.

Future work can be done regarding the feature aspects of these attacks in order to understand how features develop over time. This knowledge will further enhance the detection methods with better adaptability and efficiency against ever-evolving DDoS threats.

REFERENCES

[1] M. Jangjou and M. K. Sohrabi, “A Comprehensive Survey on Security Challenges in Different Network Layers in Cloud Computing,” Archives of Computational Methods in Engineering 2022, pp. 1–22, Jan. 2022, doi: 10.1007/S11831-022-09708-9.

[2] Gaurav Aggarwal, “How the Pandemic Has Accelerated Cloud Adoption,” Forbes. Accessed: Aug. 14, 2022. [Online]. Available: https://www.forbes.com/sites/forbestechcouncil/2021/01/15/how-the-pandemic-has-accelerated-cloud-adoption/?sh=463f9b836621

[3] Lionel Sujay Vailshery, “Cloud computing - Statistics & Facts,” Statista. Accessed: Aug. 14, 2022. [Online]. Available: https://www.statista.com/topics/1695/cloud-computing/#dossierKeyfigures

[4] E. Abdurachman, F. Lumban Gaol, and B. Soewito, “ScienceDirect ScienceDirect Survey on Threats and Risks in the Cloud Computing Environment,” Procedia Comput Sci, vol. 161, pp. 1325–1332, 2019, doi: 10.1016/j.procs.2019.11.248.

[5] Bob Violino, “Google, Microsoft ramp up cloud security as cyberattacks increase,” CNBC. Accessed: Aug. 14, 2022. [Online]. Available: https://www.cnbc.com/2022/03/29/google-microsoft-ramp-up-cloud-security-as-cyberattacks-increase.html

[6] M. Masdari and M. Jalali, “A survey and taxonomy of DoS attacks in cloud computing,” Security and Communication Networks, vol. 9, no. 16, pp. 3724–3751, Nov. 2016, doi: 10.1002/SEC.1539.

[7] Craig Sparling and Max Gebhardt, “Largest European DDoS Attack on Record,” Akamai Blog. Accessed: Mar. 29, 2023. [Online]. Available: https://www.akamai.com/blog/security/largest-european-ddos-attack-ever

[8] Z. Liu, B. Xu, B. Cheng, X. Hu, and M. Darbandi, “Intrusion detection systems in the cloud computing: A comprehensive and deep literature review,” Concurr Comput, vol. 34, no. 4, p. e6646, Feb. 2022, doi: 10.1002/CPE.6646.

[9] M. Darwish, A. Ouda, and L. Fernando Capretz, “Cloud-based DDoS Attacks and Defenses”.

[10] V. Chang et al., “A Survey on Intrusion Detection Systems for Fog and Cloud Computing,” Future Internet 2022, Vol. 14, Page 89, vol. 14, no. 3, p. 89, Mar. 2022, doi: 10.3390/FI14030089.

[11] N. Stephenson et al., “Survey of Machine Learning Techniques in Drug Discovery,” Curr Drug Metab, vol. 20, no. 3, pp. 185–193, Aug. 2019, doi: 10.2174/1389200219666180820112457.

[12] M. Wang, Y. Lu, and J. Qin, “A dynamic mlp-based ddos attack detection method using feature selection and feedback,” Computers & Security, Vol. 88, p. 101645, 2020.

[13] D.-C. Can, H.-Q. Le, and Q.-T. Ha, “Detection of distributed denial of service attacks using automatic feature selection with enhancement for imbalance dataset,” in Intelligent Information and Database Systems: 13th Asian Conference, ACIIDS 2021, Phuket, Thailand, April 7–10, 2021, Proceedings 13. Springer, 2021, pp. 386–398.

[14] P. Singh Samom and A. Taggu, “Distributed denial of service (ddos) attacks detection: A machine learning approach,” in Applied Soft Computing and Communication Networks: Proceedings of ACN 2020. Springer, 2021, pp. 75–87.

[15] Y. Wei, J. Jang-Jaccard, F. Sabrina, A. Singh, W. Xu, and S. Camtepe, "Ae-mlp: A hybrid deep learning approach for ddos detection and classification," IEEE Access, vol. 9, pp. 146 810–146 821, 2021.
[16] C. S. Kalutharage, X. Liu, C. Chrysoulas, N. Pitropakis, and P. Papadopoulos, "Explainable ai-based ddos attack identification method for iot networks," Computers, vol. 12, no. 2, p. 32, 2023.
[17] L. Antwarg, R. M. Miller, B. Shapira, and L. Rokach, "Explaining anomalies detected by autoencoders using Shapley additive explanations," Expert Systems with Applications, vol. 186, p. 115736, 2021.
[18] A. ˇ Sarˇcevi´c, D. Pintar, M. Vrani´c, and A. Krajna, "Cybersecurity knowledge extraction using XAI," Applied Sciences, vol. 12, no. 17, p. 8669, 2022.
[19] S. Lakshminarasimman, S. Ruswin, and K. Sundarakantham, "Detecting DDoS attacks using decision tree algorithm," 2017 4th International Conference on Signal Processing, Communication and Networking, ICSCN 2017, Oct. 2017, doi: 10.1109/ICSCN.2017.8085703.
[20] R. Latif, H. Abbas, S. Latif, and A. Masood, "EVFDT: An Enhanced Very Fast Decision Tree Algorithm for Detecting Distributed Denial of Service Attack in Cloud-Assisted Wireless Body Area Network," Mobile Information Systems, vol. 2015, Jan. 2015, doi: 10.1155/2015/260594.
[21] R. Latif, H. Abbas, S. Assar, and S. Latif, "Analyzing feasibility for deploying very fast decision tree for DDoS attack detection in cloud-assisted WBAN," Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 8588 LNCS, pp. 507–519, 2014, doi: 10.1007/978-3-319-09333-8_57/COVER.
[22] M. I. Kareem and M. N. Jasim, "DDOS Attack Detection Using Lightweight Partial Decision Tree algorithm," Proceedings of the 2nd 2022 International Conference on Computer Science and Software Engineering, CSASE 2022, pp. 362–367, 2022, doi: 10.1109/CSASE51777.2022.9759824.
[23] J. Ye, X. Cheng, J. Zhu, L. Feng, and L. Song, "A DDoS Attack Detection Method Based on SVM in Software Defined Network," Security and Communication Networks, vol. 2018, Apr. 2018, doi: 10.1155/2018/9804061.
[24] A. Abusitta, M. Bellaiche, and M. Dagenais, "An SVM-based framework for detecting DoS attacks in virtualized clouds under changing environment," vol. 7, p. 9, 2018, doi: 10.1186/s13677-018-0109-4.
[25] M. Myint Oo, S. Kamolphiwong, T. Kamolphiwong, and S. Vasupongayya, "Advanced Support Vector Machine-(ASVM-) based detection for Distributed Denial of Service (DDoS) attack on Software Defined Networking (SDN)," Journal of Computer Networks and Communications, vol. 2019, 2019, doi: 10.1155/2019/8012568.
[26] M. Alduailij, Q. Waqas Khan, M. Tahir, M. Sardaraz, M. Alduailij, and F. Malik, "Machine-Learning-Based DDoS Attack Detection Using Mutual Information and Random Forest Feature Importance Method," Symmetry 2022, Vol. 14, Page 1095, vol. 14, no. 6, p. 1095, May 2022, doi: 10.3390/SYM14061095.
[27] H. N. Thanh and T. Van Lang, "Use the ensemble methods when detecting DoS attacks in Network Intrusion Detection Systems," EAI Endorsed Transactions on Context-aware Systems and Applications, vol. "6," no. 19, p. 163484, Nov. 2019, doi: 10.4108/EAI.29-11-2019.163484.
[28] Bin Jia, Xiaohong Huang, Rujun Liu, and Yan Ma, "A DDoS Attack Detection Method Based on Hybrid Heterogeneous Multiclassifier Ensemble Learning," Journal of Electrical and Computer Engineering. Accessed: Aug. 17, 2022. [Online]. Available: https://www.hindawi.com/journals/jece/2017/4975343/

[29] D. Firdaus, R. Munadi, and Y. Purwanto, “DDoS Attack Detection in Software Defined Network using Ensemble K-means++ and Random Forest,” 2020 3rd International Seminar on Research of Information Technology and Intelligent Systems, ISRITI 2020, pp. 164–169, Dec. 2020, doi: 10.1109/ISRITI51436.2020.9315521.
[30] L. Alzubaidi et al., “Review of deep learning: concepts, CNN architectures, challenges, applications, future directions,” Journal of Big Data 2021 8:1, vol. 8, no. 1, pp. 1–74, Mar. 2021, doi: 10.1186/S40537-021-00444-8.
[31] S. Dong, P. Wang, and K. Abbas, “A survey on deep learning and its applications,” Comput Sci Rev, vol. 40, p. 100379, May 2021, doi: 10.1016/J.COSREV.2021.100379.
[32] X. Yuan, C. Li, and X. Li, “DeepDefense: Identifying DDoS Attack via Deep Learning,” 2017 IEEE International Conference on Smart Computing, SMARTCOMP 2017, Jun. 2017, doi: 10.1109/SMARTCOMP.2017.7946998.
[33] I. Ortet Lopes, D. Zou, F. A. Ruambo, S. Akbar, and B. Yuan, “Towards Effective Detection of Recent DDoS Attacks: A Deep Learning Approach,” Security and Communication Networks, vol. 2021, 2021, doi: 10.1155/2021/5710028.
[34] L. Xinlong and C. Zhibin, “DDoS Attack Detection by Hybrid Deep Learning Methodologies,” Security and Communication Networks, vol. 2022, pp. 1–7, May 2022, doi: 10.1155/2022/7866096.
[35] S. Tabassum, N. Parvin, N. Hossain, A. Tasnim, R. Rahman, and M. I. Hossain, “Iot network attack detection using xai and reliability analysis, “in 2022 25th International Conference on Computer and Information Technology (ICCIT). IEEE, 2022, pp. 176–181.
[36] Z. Abou El Houda, B. Brik, and L. Khoukhi, “why should I trust your ids?”: An explainable deep learning framework for intrusion detection systems in internet of things networks,” IEEE Open Journal of the Communications Society, vol. 3, pp. 1164–1176, 2022.
[37] Y. Wei, J. Jang-Jaccard, F. Sabrina, A. Singh, W. Xu, and S. Camtepe, “Ae-mlp: A hybrid deep learning approach for ddos detection and classification,” IEEE Access, vol. 9, pp. 146 810–146 821, 2021
[38] Vu, N.H. DDoS attack detection using K-Nearest Neighbor classifier method. In Proceedings of the International Conference on Telehealth/Assistive Technologies, Baltimore, Maryland, USA, 16–18 April 2008; IEEE: Piscataway Township, NJ, USA, 2008; pp. 248–253.
[39] Cheng, J.; Yin, J.; Liu, Y.; Cai, Z.; Wu, C. DDoS attack detection using IP address feature interaction. In Proceedings of the IEEE International Conference on Intelligent Networking and Collaborative Systems, Thessalonika, Greece, 24–26 November 2010; IEEE: Piscataway Township, NJ, USA, 2009; pp. 113–118.
[40] Wang, C.; Zheng, J.; Li, X. Research on DDoS attacks detection based on RDF-SVM. In Proceedings of the 10th International Conference on Intelligent Computation Technology and Automation, Changsha, China, 9–12 October 2017.
[41] Fadlil, A.; Riadi, I.; Aji, S. Review of detection DDoS attack detection using Naïve Bayes classifier for network forensics. Bull. Electr. Eng. Inform. 2017, 6, 140–148.
[42] Dincalp, U. Anomaly based distributed denial of service attack detection and prevention with machine learning. In Proceedings of the 2nd International Symposium on Multidisciplinary Studies and Innovative Technologies, Ankara, Turkey, 19–21 October 2018.
[43] Ahanger, T.A. An effective approach of detecting DDoS using artificial neural networks. In Proceedings of the 2017 International Conference on Wireless Communications, Signal Processing and Networking, Chennai, India, 22–24 March 2017; IEEE: Piscataway Township, NJ, USA, 2017; pp. 707–711.

[44] Zahid Hasan, Md., Zubair Hasan, K. M., & Sattar, Abdus (2018). Burst header packet flood detection in optical burst switching network using deep learning model. Procedia Computer Science, 143, 970–977.
[45] Krishnan, Prabhakar, Duttagupta, Subhasri, & Achuthan, Krishnashree (2019). VARMAN: Multi-plane security framework for software defined networks. Computer Communications, 148, 215–239.
[46] Zhu, M., Ye, K., & Xu, C. Z. (2018). Network anomaly detection and identification based on deep learning methods. In M. Luo, & L. J. Zhang (Eds.), Cloud Computing – CLOUD 2018. CLOUD 2018. Lecture Notes in Computer Science. Cham: Springer.
[47] Alzahrani, S., & Hong, L. (2018). Detection of distributed denial of service (DDoS) attacks using artificial intelligence on cloud. In 2018 IEEE World Congress on Services (SERVICES), San Francisco, CA (pp. 35–36).
[48]McKinney, W. (2010). Data Structures for Statistical Computing in Python. Proceedings of the 9th Python in Science Conference.
[49] Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., ... & Liu, T. Y. (2017). LightGBM: A highly efficient gradient boosting decision tree. Advances in Neural Information Processing Systems
[50] Prokhorenkova, L., Gusev, G., Vorobev, A., Dorogush, A. V., & Gulin, A. (2018). CatBoost: unbiased boosting with categorical features. Advances in Neural Information Processing Systems.
[51] Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining
[52] Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic Minority Over-sampling Technique. Journal of Artificial Intelligence Research
[53] Kluyver, T., Ragan-Kelley, B., Pérez, F., Granger, B., Bussonnier, M., Frederic, J., ... & Willing, C. (2016). Jupyter Notebooks - A publishing format for reproducible computational workflows. Proceedings of the 20th International Conference on Electronic Publishing
[54] Lundberg, S. M., & Lee, S. I. (2017). A Unified Approach to Interpretable Model Predictions. Advances in Neural Information Processing Systems
[55] Guyon, I., Weston, J., Barnhill, S., & Vapnik, V. (2002). Gene Selection for Cancer Classification using Support Vector Machines. Machine Learning
[56] Powers, D. M. W. (2011). Evaluation: From Precision, Recall, and F-Measure to ROC, Informedness, Markedness & Correlation. Journal of Machine Learning Technologies.
[10] Python Software Foundation. (2023). Openpyxl: A Python library to read/write Excel 2010 xlsx/xlsm files. Available at https://openpyxl.readthedocs.io/.
[57] I. Sharafaldin, A. H. Lashkari, S. Hakak, and A. A. Ghorbani, “Developing realistic distributed denial of service (DDoS) attack dataset and taxonomy,” Proceedings - International Carnahan Conference on Security Technology, vol. 2019-October, Oct. 2019, doi: 10.1109/CCST.2019.8888419.