Data-Driven Detection of Multi-vector IoT DDoS Attacks across Network Layers

Rania A. Al-Ali [1], Mohammad Alnabhan [1], and Qasem Abu Al-Haija [2,*]

[1] Department of Cybersecurity, Princess Sumaya University for Technology, Amman, Jordan
[2] Department of Cybersecurity, Jordan University of Science and Technology, Irbid, Jordan
Email: ran20238134@std.psut.edu.jo (R.A.A.-A); m.alnabhan@psut.edu.jo (M.A.); qsabuhaija@just.edu.jo (Q.A.A.-H)
*Corresponding author

***Abstract*—As cyberattacks targeting Internet of Things (IoT) networks grow more sophisticated, the demand for models capable of accurately detecting and mitigating these threats becomes increasingly urgent existing detection systems often concentrate on a single attack surface which leads to critical blind spots in IoT network monitoring. This research introduces an intensive IoT attack-detection framework that addresses internal and external attack sources, leveraging multi-layer attack vectors across the application, transport, network, and data link layers. The proposed framework was evaluated in two phases. In the first phase, three datasets that simulate a distinct attack source were created: outbound, including Dynamic Host Configuration Protocol (DHCP) amplification and DHCP starvation; inbound attacks, including application-layer attacks, Synchronize (SYN) flood, User Datagram Protocol (UDP) flood, Internet Control Message Protocol (ICMP) Smurf, and ICMP direct; and internal Address Resolution Protocol (ARP) spoofing. Three machine learning models; Random Forest, Light Gradient-Boosting Machine (LightGBM), and Categorical Boosting (CatBoost) were evaluated using accuracy, precision, recall, F1-Score, and Receiver Operating Characteristic (ROC) curve. In the second phase, a new dataset was generated by combining all datasets from the first phase. On the combined dataset, LightGBM achieved the highest performance, with accuracy: 94.08%, precision: 93.94%, recall: 94.08%, and F1-Score: 93.90%. Random Forest and CatBoost showed comparable performance, with all metrics ranging from approximately 93.2% to 93.9%. LightGBM demonstrates a slight edge in overall detection performance compared to other models, which highlights its effectiveness in detecting diverse and complex attack patterns across multiple traffic directions.**

***Keywords*—Internet of Things (IoT) security, Distributed Denial of Service (DDoS) attack detection, inbound traffic attacks, outbound traffic attacks, internal traffic attacks, machine learning, network layer attacks, Light Gradient-Boosting Machine (LightGBM)**

I. Introduction

With the growing number of cyberattacks targeting the Internet of Things (IoT), there is an urgent need to enhance security mechanisms that can effectively detect these threats. The widespread deployment of IoT devices, combined with their limited built-in security and high interconnectivity, makes these networks particularly vulnerable to Distributed Denial of Service (DDoS) attacks. These attacks can disrupt network availability, interrupt critical services, and compromise the integrity of entire IoT ecosystems.

This research focuses on detecting two categories of attack traffic within IoT networks: external and internal. External attacks include both inbound and outbound threats, underscoring the need for comprehensive monitoring in both directions. Outbound DDoS attacks include Dynamic Host Configuration Protocol (DHCP) starvation and DHCP amplification. In contrast, inbound attacks include application-layer attacks, Synchronize (SYN) floods, User Datagram Protocol (UDP) floods, Internet Control Message Protocol (ICMP) Smurf attacks, and ICMP direct attacks, all of which are designed to overwhelm network resources and degrade performance.

Internal threats, such as Address Resolution Protocol (ARP) spoofing, disrupt network communication and introducing significant security risks. To address these challenges, we propose a Machine Learning (ML) framework that enhances IoT security by analyzing network traffic and accurately classifying attacks. By leveraging data-driven techniques, the model effectively distinguishes between normal and malicious traffic, achieving high precision and reliability [1].

The goal of this research is to detect DDoS attacks using ML to improve IoT network security, strengthen resilience against inbound, outbound, and internal traffic attacks, and enhance the overall security and reliability of IoT networks.

Despite the increasing number of IoT DDoS detection models, most existing approaches focus on a single traffic direction or a limited set of attack types. Furthermore, many evaluations prioritize accuracy over other critical metrics such as precision, which is essential for minimizing false alarms in real-world deployments. Therefore, there is a need for a comprehensive ML-based model capable of detecting multiple attack types across inbound, outbound, and internal traffic.

Manuscript received January 23, 2026; revised March 18, 2026; accepted April 21, 2026; published July 28, 2026.

Main Contributions—The key contributions of this research are:

- Dataset Integration: Integrate two datasets to construct a comprehensive dataset: the INDDoS24 dataset [2], which contains only inbound attack traffic, and a Generative Artificial Intelligence (GenAI)-based dataset that includes outbound and internal attacks.
- Comprehensive Multi-Attack Detection: Develop a robust model capable of detecting eight distinct attack types—DHCP starvation, DHCP Amplification, ICMP Direct, ICMP Smurt, SYN Flood, UDP Flood, Application Layer attacks, and ARP Spoofing—in addition to normal traffic, covering inbound, outbound, and internal attack categories.
- Assessment Across Scenarios: Evaluate the model in two phases under four scenarios: outbound attacks, inbound attacks, internal attacks, and a combined dataset containing all attack types, to examine robustness and generalizability.

II. Background

A. IoT Attack Surfaces

IoT refers to interconnected devices found in home appliances, industrial sensors, and smart city systems that can collect, transmit, and process data over the Internet. Ensuring that these devices remain reliable, secure, and trustworthy has become a challenge due to their growing use, diversity, limited resources, and large scale.

Inbound attacks, including SYN floods, UDP floods, ICMP Smurf attacks, and application-layer exploits, originate outside the network. Outbound attacks, such as DHCP amplification and DHCP starvation, occur when compromised devices within the network are used to attack other systems. Internal attacks, such as ARP spoofing, exploit trust within the network, disrupting communications and compromising data integrity.

By developing IoT attack-detection frameworks, researchers can strengthen the resilience of these environments against increasingly sophisticated attacks [1, 3].

B. Machine learning(ML) for Intrusion Detection

ML is widely used to detect abnormal patterns and malicious activities in IoT networks automatically. Traditional security methods, such as Intrusion Detection Systems (IDS), are often insufficient for detecting anomalies in network traffic. Numerous studies have shown that models such as Random Forest (RF), LightGBM, and Categorical Boosting (CatBoost) are highly effective at detecting such attacks, frequently achieving strong performance.

1) Random forest

Random Forest (RF) is an ensemble learning technique that constructs multiple Decision Trees (DT) by randomly sampling data and features. It then combines the predictions of each tree to produce a final decision. RF's ability to reduce overfitting and underfitting enhances model performance [4, 5]. In addition, it has proven effective in handling imbalanced datasets. For example, pairing RF with the Synthetic Minority Oversampling Technique (SMOTE) achieved 92.4% accuracy for intrusion detection in wireless sensor networks [6]. Other studies have also highlighted its strong detection capabilities and low false alarm rates when evaluated on popular datasets such as NSL-KDD [7, 8].

2) LightGBM

LightGBM is a gradient boosting framework that employs a leaf-wise growth strategy, selecting nodes that achieve the largest error reduction. This approach enables deep tree construction, resulting in superior accuracy and faster training [9, 10]. These features make LightGBM highly suitable for large-scale and real-time intrusion detection in IoT environments. As demonstrated in Ref. [11], LightGBM achieves high performance while reducing prediction time by over 90% compared to other models. Other studies also rank LightGBM among the top-performing algorithms in IoT security [12].

3) CatBoost

CatBoost is a gradient boosting algorithm that combines the predictions of multiple simple models. It leverages techniques such as Ordered Boosting and SMOTE to minimize overfitting and underfitting while enhancing model interpretability. CatBoost has demonstrated strong performance on noisy, high-dimensional, and imbalanced datasets, making it suitable for IoT security applications [14, 15].

C. Generative Artificial Intelligence (GenAI)

GenAI is a type of artificial intelligence that have the ability to generate a new data that emulate patterns of existing datasets. Unlike traditional AI which their primary purpose is on prediction. GenAI is capable of generate different types of media such as text, images, or synthetic data. Its ability is powered by advanced deep learning architectures like transformers, GenAI has achieved remarkable success in various domains including natural language processing and computer vision. In cybersecurity, GenAI offers promising capabilities for enhancing attack detection and automating incident response.

D. IoT Network Attack Vectors

IoT networks are vulnerable to a wide range of security attacks due to their unique device types and deployment characteristics. These attacks can originate either outside or inside the network. External attacks include inbound attacks, which originate from outside and target IoT devices, and outbound attacks, which occur when compromised devices are used to launch attacks against other systems. Understanding these attack types and their sources is essential for developing effective detection and mitigation strategies. In this section, IoT network attacks are categorized into external which includes both inbound and outbound attacks and internal attacks.

1) External attacks

External attacks originate from outside the network or system. Remote adversaries who seek to exploit vulnerabilities to gain unauthorized access carry out these

attacks. External attacks in IoT environments are classified into two categories: inbound and outbound [1, 3].

- Inbound attacks target internal systems from external sources with the aim of disrupting services or compromising the network. These attacks include SYN floods, in which attackers exploit the Transmission Control Protocol (TCP) handshake process to overload a server, and UDP floods, which generate excessive traffic to exhaust system resources. Application-layer attacks often cause service failures by exploiting vulnerabilities in web servers or databases and are difficult to mitigate using traditional security measures. Additionally, ICMP Smurf and ICMP Direct attacks involve sending a large number of ICMP echo requests to flood a target network. In ICMP Smurf attacks, the attacker spoofs the victim's IP address, causing all devices on the network to respond to the victim and generate massive traffic. ICMP Direct attacks similarly send high volumes of ICMP echo requests, potentially exhausting system resources and resulting in a denial of service [1].
- Outbound attacks occur when an internal system is compromised and used to launch attacks against other systems or networks. DHCP starvation is a type of outbound attack in which the attacker sends a large number of DHCP requests to exhaust the server's available IP addresses, preventing legitimate devices from obtaining them. DHCP amplification is another outbound attack that exploits the DHCP protocol, amplifying the attacker's traffic and sending large volumes of malicious data to external targets, resulting in network congestion or service disruptions [16, 17].

2) Internal attacks

Internal attacks originate from within the network and are often carried out by malicious insiders or compromised devices with legitimate access. These attacks exploit trust relationships and local network protocols to disrupt communications, intercept data, or escalate privileges. A common example is ARP spoofing, in which the attacker sends falsified ARP messages to associate their MAC address with the IP address of another device on the same local network. This can lead to man-in-the-middle attacks, session hijacking, or denial of service, enabling the attacker to intercept, modify, or block data intended for the targeted device [18, 19].

II. Literature Review

This section reviews studies that employ ML to detect DDoS attacks in IoT networks, comparing attack types, ML models, dataset sizes, and traffic direction. The review highlights the strengths and limitations of existing approaches, emphasizing strategies that have proven most effective while identifying gaps that remain. Addressing these gaps is essential for enhancing IoT network security against multi-vector DDoS attacks.

Table I summarizes the ML techniques applied to different DDoS attack types, dataset sizes, evaluation metrics, traffic directions, and the limitations of each study.

A. External Attacks

1) Inbound attacks

Srivastava *et al.* [20] present an ensemble approach for detecting DDoS attacks in IoT networks by combining the strengths of XGBoost and Random Forest (RF). Ensemble techniques leverage multiple ML models for more reliable detection and system resilience. Trained on a dataset of 2800 samples, the model achieved 99% accuracy, demonstrating the effectiveness of ensemble methods in enhancing precision and robustness against adversarial attacks in IoT security. Kumar *et al.* [21] propose a detection system for DDoS and IoT botnet attacks using ML and deep learning models. The dataset contains 7999 samples, which were used to evaluate KNN, Logistic Regression, SVM, RF, CNN, and LSTM. The LSTM model achieved 99.80% accuracy, highlighting the potential of deep learning for IoT traffic analysis.

Sayed *et al.* [22] illustrates IoT vulnerability to DDoS attacks and proposes integrating ML with blockchain for enhanced security. By analyzing a dataset of 346,869 samples, DDoS attacks were detected using Decision Tree (DT) and RF models, achieving accuracies of approximately 99.60%. The most effective ML model was integrated with blockchain to improve detection and mitigation, enhancing overall IoT network security and resilience. Srivastava *et al.* [23] suggest using an LSTM model to detect DDoS attacks, which often exhibit temporal patterns in malicious traffic. Leveraging traces left by compromised IoT devices in network traffic, the model achieved high precision of 99.4%.

Srivastava *et al.* [24] discuss IoT security challenges, noting that device proliferation and diversity increase vulnerability to DoS and DDoS attacks. Due to the lack of a universal security protocol, continuous monitoring and adaptive strategies are necessary. They propose an ML-based approach for rapid detection and mitigation of malicious activity, using Linear and Non-Linear SVM. The Non-Linear SVM achieved higher accuracy (97.8%) compared to the Linear SVM (93%).

Overall, these studies differ in dataset size, attack coverage, and ML techniques. While small datasets (e.g., 2800 samples in [20, 23] achieve high accuracy (99–99.4%), generalizability is limited and precision is often unreported. Moderate datasets (7999 samples in [21]) and large datasets (346,869 samples in [22]) improve evaluation, yet many studies focus primarily on accuracy, neglecting precision, recall, and F1-Score. Most approaches also focus solely on inbound traffic, highlighting gaps in comprehensive multi-vector IoT DDoS detection.

2) Outbound attacks

Bhayo *et al.* [25] discuss the vulnerability of IoT to DDoS attacks and explores integrating Software Defined Networking (SDN) with IoT to enhance security. The study implements ML models Naive Bayes, Decision Tree, and SVM within an SDN-WISE IoT controller to detect DDoS attacks. The models achieve high accuracy: NB 97.4%, SVM 96.1%, and DT 98.1%. The paper does not mention any details about dataset, and it focuses only

on outbound traffic. In addition to only accuracy is reported which could lead to an inaccurate illustration of performance. These gaps show that a study required with comprehesive evaluation and detection of multidirectional traffic.

Kumavat and Gomes [26] present a Multi-Layer DDoS Detection and Mitigation (MLDDM) protocol for IoT-enabled WSNs. Detection is performed using a trust-based network tree analysis, where each node's behavior across application (HTTP floods), transport (SYN Flood, TCP-based DDoS), internet (ICMP Flood, UDP Flood), and edge layers is continuously evaluated through audit trails and trust parameters. Nodes identified as suspicious are disconnected to prevent further attack propagation, covering primarily inbound traffic, with outbound mitigation applied at the edge layer.

TABLE I. LITERATURE REVIEW TABLE

Ref	Year	Attack Types	ML Technique	Size of Dataset	Evaluation	Direction of Network Traffic	Limitations
[26]	2025	SYN Flood, UDP Flood, Application Layer Attacks	Not Applied	-	-	Inbound	• Three types of DDoS attacks are covered.
[27]	2025	ARP Spoofing	SVM, Muli Layer Perceptron, Linear regression and RF	Not mentioned	Accuracy: • SVM: 95% •Muli Layer Perceptron: 97% • Linear Regression: 78% • RF: 88%	Internal	• Focuses on ARP spoofing as a DDoS vector • Only one type of attacks. • Dataset size is not mentioned.
[23]	2024	TCP flood	Long Short-Term Memory (LSTM)	2800 samples	• Accuracy: 99.4% • Precision: 99.4%	Inbound	• Small dataset • Only one DDoS attack type is covered
[20]	2024	TCP flood	Ensemble techniques: RF and XGBoost	2800 samples	Accuracy: 99%	Inbound	• Lack of details about dataset
[21]	2024	Not mentioned	LSTM	7999 samples	Accuracy: 99.8%	Inbound	• DDoS attack types are not mentioned
[22]	2024	DDoS slowloris, and DDoS Hulk	DT and RF	346,869 samples	DT: • Accuracy: 99.5% • Precision: 99.6% • Recall: 99.5% • F1-Score: 99.6% RF: • Accuracy: 99.50% • Precision: 99.66% • Recall: 99.61% • F1-Score: 99.63%	Inbound	• Two types of DDoS attacks are covered
[24]	2023	TCP flood and HTTP flood	Linear and non-linear Support Vector Machine (SVM)	Not mentioned	Accuracy: • linear: 93% • Non-linear: 97.8%	Inbound	• Lack of details about dataset • Two types of DDoS attacks are covered
[25]	2023	Zombie	NB, SVM, and DT	Not mentioned	Accuracy: • NB: 97.4% • SVM: 96.1% • SVM: 98.1%	Outbound	• Lack of details about dataset

B. Internal Attacks

Praveena and Sudha [27] detecting ARP spoofing attacks which could trigger denial-of-service conditions. The study addresses only one type of attack, dataset with four models Linear Regression, RF, Multi-layer perceptron, and SVM achieve accuracy rates of 78%, 88%, 97%, and 95%. The study addresses only one type of attack, no details about dataset evaluated, and only accuracy is reported which limits assessment of the models reliability.

C. Research Gap and Novelty

While numerous studies have proposed IoT DDoS detection methods, most focus on inbound or outbound traffic only and consider a limited set of attack types. Evaluations often emphasize accuracy rather than metrics like precision, recall, or F1-Score, which are critical to minimize false alarms in real-world deployments. These gaps underscore the need for a comprehensive model capable of detecting multiple attack types across inbound, outbound, and internal traffic, using practically relevant performance metrics. Our study addresses this gap by proposing a model that integrates eight attack types along with normal traffic, providing robust, multi-vector DDoS detection with high performance and generalizability across diverse IoT scenarios. Table I summarizes the research reviewed in this paper.

III. METHODOLOGY

This section explains how ML applied to detect attacks and distinguish between inbound, outbound, and internal DDoS attacks in IoT networks, supporting early attack detection and improved security. Fig. 1 is a visual representation of the methodology workflow.

A. Dataset

1) Dataset source and composition

This study applied a hybrid dataset that contains a combination of real-world IoT network traffic dataset with synthetically generated attack samples using GenAI. The real traffic data was obtained from the INDDDoS24 Kaggle dataset [2] which was collected between

January 1, 2019, and July 1, 2024 and contains 48,192 samples. The dataset captures hourly network traffic from diverse IoT devices under both normal conditions and multiple attack scenarios across different network layers which make it suitable for ML intrusion detection.

From the real-world INDDDoS24 dataset, a stratified subset of 8000 samples was selected from a dataset size of 48,000 samples to represent three types of attacks: SYN flood, UDP flood, application-layer attacks with normal traffic. This sampling step was performed to avoid bias and to balance the dataset with synthetic generated attack types, each of which consist each of them 8000 samples. The selected subset contains 23 features excluding the label.

2) Synthetic data generation using generative AI

Due to the lack of datasets covering inbound, outbound, and internal attack scenarios comprehensively, a GPT-based generative language model was employed to synthesize attack types that were underrepresented or missing in the INDDoS2024 dataset. This approach ensures that all relevant traffic characteristics are fully represented when combined with the INDDoS24 dataset [2] avoiding missing or inconsistent features between datasets.

Specifically, synthetic data was generated for the following attack types:

- Outbound attacks (Network layer): DHCP Starvation, DHCP Amplification
- Inbound attacks (Network layer): ICMP Direct, ICMP Smurf
- Inbound attacks (Transport layer): SYN Flood, UDP Flood
- Inbound attacks (Application layer): Application Layer attacks
- Internal attacks (Data Link layer): ARP Spoofing

Each synthetic attack type consists of 2000 samples, resulting in 10,000 synthetic samples. Combined with the original dataset 8000 samples, the final dataset comprises 18,000 samples with 23 features prior to preprocessing.

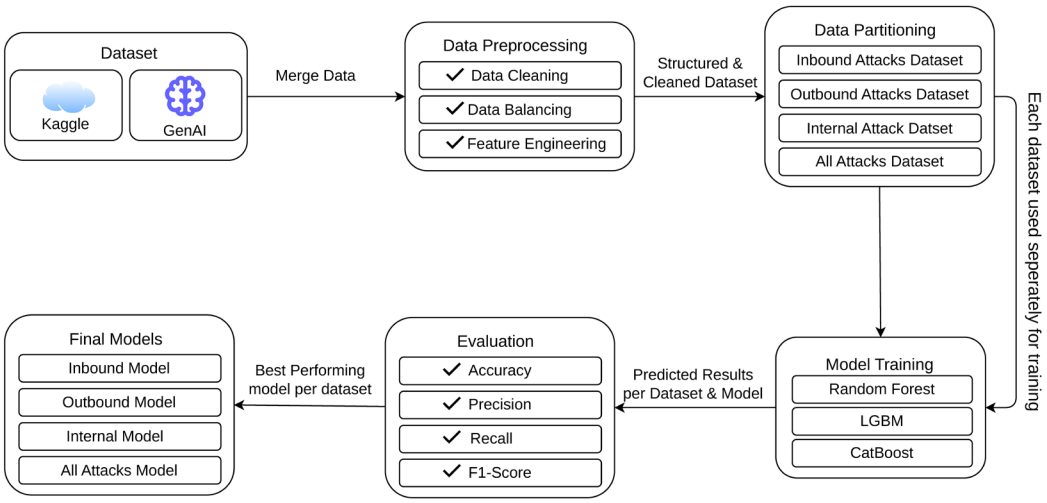

Fig. 1. Methodology process flow.

The generative model was provided with existing network traffic features and prompted with traffic-specific instructions to generate realistic instances of these attacks. The generated samples were validated using feature-wise statistics (mean, variance) Table II and Kolmogorov-Smirnov (KS) tests Table III. This approach enabled balanced representation across inbound, outbound, and internal traffic, improving the robustness of machine learning models for attack detection.

3) Validation of AI-generated data

The validation of AI-generated data ensures that synthetic samples resemble the real dataset closely enough to support model training. In this study, a total of 10,000 AI-generated samples were analyzed. As shown in Table II the generated data maintains similar mean values and overall trends for most features, which indicates that the generative model captures the core patterns of network traffic. Some differences in variance and extreme values exist for certain features, but these deviations are minor and do not significantly affect the overall distribution. The numerical discrete features, such as source and destination generated by AI, fall within realistic value ranges. Specifically, 75.5% of Source Ports and 74.8% of Destination Ports lie between 1024 and 49151, while 24.5% of Source Ports and 25.2% of Destination Ports lie between 49152 and 65535 with no values observed below 1024. This distribution of ports show that malicious traffic avoids well-known ports to limit detectability and targets less frequently monitored services. To further quantify the similarity between the AI-generated dataset and the original dataset, we performed a univariate KS, as summarized in Table III. As expected, given that the two datasets contain fundamentally different attack taxonomies, the target variable Attack_Type exhibits a very large discrepancy (KS = 0.800, $p < 10^{-4}$), confirming a substantial distributional shift in the primary label. Traffic- and time-related features, including Protocol, hour, bytes_to_packets_ratio, rate_packet_size, attack_duration_binned, and inter_arrival_time_binned, also show pronounced differences (KS > 0.2, $p < 10^{-4}$). This variation is consistent with the differing attack scenarios represented in each dataset, as distinct attack

types naturally induce different traffic patterns and temporal characteristics. In contrast, device- and network-level features such as network_zone, network_zone_score, and device_risk_factor have negligible KS values (≈ 0–0.006, $p > 0.99$), suggesting that their marginal distributions remain effectively unchanged across datasets despite the variation in attack types. Intermediate features, including Device_Type, Total_Unique_Count, and Operating_System, yield moderate KS statistics (0.08–0.14), indicating partial but not substantial variation. Overall, these KS test results suggest that the generative model preserves the distributions of most non-attack-related features, while allowing attack-related and traffic-related features to vary across datasets. This behavior can be beneficial for downstream model training and evaluation, as it maintains a consistent representation of the underlying device and network environment while exposing models to diverse and shifted attack patterns. Furthermore, these findings provide supporting evidence for the validity of the AI-generated data, as they demonstrate its ability to retain key statistical properties of the original dataset while introducing realistic variability in attack patterns.

4) Impact and limitations of synthetic data

Synthetic data played a crucial role in enhancing the training of ML models by providing a more balanced representation of underrepresented or missing attack types, including DHCP Starvation, DHCP Amplification, ARP Spoofing, ICMP Direct, and ICMP smurf attacks that were not present in the original INDDDoS24 dataset. By combining these AI-generated samples with the real dataset, the models were able to learn patterns across inbound, outbound, and internal traffic more effectively, thereby improving detection accuracy and robustness. Statistical validation techniques, such as the KS test performed on each feature, confirmed that most non-attack-related features exhibit distributions close to the real traffic data, while attack- and traffic-related features showed pronounced differences. These differences are primarily due to the presence of newly generated attacks, which introduce distinct traffic patterns and temporal behavior compared to the original attacks, resulting in observable distributional shifts in certain features. This analysis demonstrates that the generative model preserves the key statistical properties of the underlying dataset while exposing the models to realistic and diverse attack scenarios.

However, synthetic data has inherent limitations. Generative approaches, such as VagueGAN [28], which produce seemingly benign records with subtle adversarial noise, may introduce hidden patterns that can lead to bias or distributional shifts. Additionally, synthetic samples may not fully capture the complexity and variability of real-world network traffic, which could reduce the generalizability of models trained solely on the generated data. Therefore, relying exclusively on synthetic data without proper validation against real traffic could compromise detection performance, particularly for unseen or diverse attack types. Integrating the AI-generated and real INDDDoS24 datasets, along with validation techniques such as KS testing and adversarial detection, was essential to ensure reliable and robust model performance.

TABLE II. COMPARISON OF MEAN AND VARIANCE BETWEEN REAL AND AI-GENERATED FEATURES

Feature	Real Mean	AI Mean	Real Variance	AI Variance
Packet Size	767.9	779.5	1.77E+05	1.72E+05
Payload Length	749.4	709.2	1.80E+ 5	1.60E+05
Flow Duration	30.17	28.68	298.68	284.71
Bytes in Flow	50,686.90	39,628.10	8.29E+08	1.01E+09
Packets in Flow	496.1	44.2	8.37E+04	766.78
Average Packet Size	778.8	586.4	1.78E+05	1.08E+05
Inter-Arrival Time	2.51	4.43	2.08	7.88
Rate of Packets	49.99	15.74	806.04	570.29
Anomaly Score	0.506	0.539	0.0828	0.05
Attack Duration	3.1	6.88	113.61	42.52

TABLE III. KS TEST STATISTICS FOR FEATURES

Feature	KS Statistic	*p*-value
Attack Type	0.8	0.00E+00
Protocol	0.374	0.00E−02
hour	0.353	0.00E−02
Bytes to packets ratio	0.31	0.00
Rate packet size	0.287	0.00E−02
Attack duration binned	0.256	8.74E−257
Inter arrival time binned	0.2	1.26E−155
Packet flow size	0.17	3.25E−112
Device Type	0.137	1.30E−72
Total Unique Count	0.13	9.28E−66
Is weekend	0.11	1.71E−47
Operating System	0.085	2.01E−28
Anomaly Score	0.079	2.55E−24
Target Device	0.077	1.72E−23
Payload/ Packet Size Ratio	0.069	1.38E−18
Log Payload Length	0.035	3.94E−05
Source Port	0.023	1.53E−02
Destination Port	0.023	1.92E−02
Network zone	0.006	9.99E−01
Network zone score	0.006	9.99E−01
Device risk factor	0	1.00E+00

B. Data Preprocessing

After the dataset have been organized and ready for analysis, data preprocessing is vital step in process flow. It ensures that the data is properly formatted and encoded so it can be effectively used by ML algorithms. Well-executed preprocessing plays a key role in improving model performance and reliability, particularly in accurately identifying various types of network attacks.

1) Fill null values using K-nearest neighbor

Handling missing data is essential for the integrity of the dataset. In this research, missing values are filled using either the KNN imputation which replaces missing values by averaging the values of the nearest neighbors, or the mean value which provides a simpler method by filling in the average of the values for a given feature. Both help in ensure that the dataset remains complete without introducing significant bias.

2) Feature engineering

Feature engineering is a crucial step in the data preprocessing phase that involves creating new features from the existing features which improve the performance of ML models, helps in enhancing the models ability to detect patterns and make accurate predictions by providing

it with the most relevant and important information. In the context of network traffic analysis such as packet size, flow duration, and rate of packets.

Table AI is all the features after applying feature engineering on them.

3) Label encoder

ML algorithms often require input data to be numerical not categorical. The label encoder is applied to convert categorical variables such as device types or operating systems into numeric format by assigning a unique integer to each category. This transformation allows the model to process non-numeric data effectively while preserving the inherent categorical relationships among the values. Table IV shows the encoder for each attack type used.

TABLE IV. LABEL ENCODING FOR ATTACK TYPES

Label	Attack Type
0	ARP Spoofing
1	DHCP Amplification
2	Application Layer Attack
3	DHCP Starvation
4	ICMP Direct
5	ICMP Smurf
6	Normal
7	SYN Flood
8	UDP Flood

4) Class balancing

To prevent bias in the ML models, the dataset was balanced so that each attack type contains 2000 samples and normal traffic also contains 2000 samples. This results in a total of 18,000 samples for the entire dataset, ensuring uniform representation across all classes and improving the reliability of the model's predictions.

Table AII shows the features present in the dataset before preprocessing, and Table AI lists the features after preprocessing.

C. Attack Distribution Analysis

The analysis of attack distribution shows distinct patterns depending on time of day, device type, target device, and operating system Table V.

From a temporal perspective, inbound attacks show a clear peak at night (65.4%), suggesting that attackers typically take advantage of times when visibility and monitoring are lower. On the other hand, outbound attacks are more common in the morning (44%) and evening (43.7%), which may indicate attempts at data exfiltration or botnet-related activity taking place during periods of high network usage. Although they are less common overall, internal attacks are more common in the morning (14.7%), which may be due to insider activity or automated attacks that are initiated during business hours.

Sensors and cameras are the most targeted devices in all traffic directions due to their exposure and vital importance in IoT systems. They are equally affected by inbound and outbound attacks, but internal attacks shows a slightly greater emphasis on sensors (35.7%), perhaps because of their susceptibility to insider manipulation. Smart hubs are rarely targeted internally (0%) because they are difficult to access from within the network, while smart appliances are the main targets of internal attacks (41.9%) because of weaker internal defences.

TABLE V. DISTRIBUTION OF ATTACKS ACROSS TIME, DEVICE TYPE, TARGET DEVICE AND OPERATING SYSTEM

Attack Direction	Time of Day (%)	Device Type (%)	Target Device (%)	Operating System (%)
Inbound	• Morning: 41.5 • Evening: 41.7 • Night: 65.4	• Camera: 31.7 • Sensor: 34.4 • Smart appliance: 26.3 • Smart hub: 7.6	• Camera: 23.4 • Sensor: 27.1 • Smart appliance: 27.5 • Smart hub: 22	• Linux: 38.8 • RTOS: 25 • Windows: 36.2
Outbound	• Morning: 44 • Evening: 43.7 • Night: 26	• Camera: 33.2 • Sensor: 33.1 • Smart appliance: 22 • Smart hub: 9.7	• Camera: 22.9 • Sensor: 25.3 • Smart appliance: 29.1 • Smart hub: 22.9	• Linux: 36.4 • RTOS: 25 • Windows: 39
Internal	• Morning: 14.7 • Evening: 14.6 • Night: 8.6	• Camera: 33.2 • Sensor: 35.7 • Smart appliance: 31.2 • Smart hub: 0	• Camera: 16.5 • Sensor: 17.9 • Smart appliance: 41.9 • Smart hub: 25.8	• Linux: 34.3 • RTOS: 30.4 • Windows: 35.4

In terms of operating systems, Linux devices are the most attacked devices because of their extensive use in IoT the environment. Windows devices are most vulnerable to outbound attacks (39%), while RTOS devices are particularly vulnerable to internal attacks (30.4%).

D. Experimental Setup

In all experiments, the dataset was split into 80% training and 20% testing, with stratification by attack type to preserve the class distribution. A fixed random seed of 42 was used to ensure reproducibility. Additionally, 5-fold stratified cross-validation was applied to the training set to enable a robust evaluation and reduce variability in performance.

Library Versions were Python 3.10 with pandas 1.5.3, numpy 1.23.5, and scikit-learn 1.2.1. Three ML models were trained using the following hyperparameters:

1) Random forest

- Number of trees (n_estimators): 100.
- Minimum samples per split: 2.

2) CatBoost

- Objective: multi-class softmax.
- Learning rate: 0.1.
- Maximum depth: 6.
- Number of estimators: 100.

3) LightGBM

- Learning rate: 0.01.

- Number of estimators: 100.
- Number of leaves: 31.

E. Training on Different Models

The paper evaluate the performance using RF, LightGBM, and CatBoost to identify which model perform the best in DDoS detection. The evaluation is performed in two phases. In the first phase, the models were trained and tested separately in three scenarios: outbound attacks with normal, inbound attacks with normal, and internal attacks with normal to enable a detailed performance analysis for each type individually. In the second phase, a combined dataset that integrates all attacks with normal data. This approach enables the evaluation of model performace to be trained on traffic direction attacks versus a comprehensive dataset that includes all attack types.

1) RF classifier

- Outbound attacks only with normal:
  The dataset that contains outbound attacks with normal trained on RF which achieved accuracy, precision, recall, and F1-Score of 97.7%.
- Inbound attacks only with normal:
  The dataset that contains inbound attacks with normal have been trained on RF and achieved accuracy, precision, recall, and F1-Score between 93.6–94%.
- Internal attack only with normal:
  The dataset that contains internal attack with normal trained on RF which achieved accuracy, precision, recall, and F1-Score of 99.1%.
- Combined all attacks with normal:
  The dataset that contains all attacks with normal have been trained on RF and acheived accuracy, precision, recall, and F1-Score around 99.9%.

2) LightTGBM classifier

- Outbound attacks only with normal:
  The dataset that contains outbound attacks with normal has been trained on LightGBM and achieved accuracy, precision, recall, and F1-Score of 100%.
- Inbound attacks only with normal:
  The dataset that contains inbound attacks with normal has been trained on LightGBM and achieved accuracy, precision, recall, and F1-Score of 99.5%.
- Internal attack only with normal:
  The dataset that contains internal attack with normal has been trained on LightGBM and achieved accuracy, precision, recall, and F1-Score of 100%.
- Combined all attacks with normal:
  The dataset that contains all attacks with normal on LightGBM and achieved accuracy, precision, recall, and F1-Score of 99.8%.

3) Catboost classifier

- Inbound attacks only with normal:
  The dataset that contains inbound attacks with normal has been trained on CatBoost and achieved accuracy, precision, recall, and F1-Score around 83%.
- Outbound attacks only with normal:
  The dataset that contains outbound attacks with normal has been trained on CatBoost and achieved accuracy, precision, recall, and F1-Score 100%.
- Internal attack only with normal:
  The dataset that contains outbound attacks with normal has been trained on CatBoost and achieved accuracy, precision, recall, and F1-Score 100%.
- Combined all attacks with normal:
  The dataset that contains all attacks with normal have been trained on CatBoost and achieved accuracy, precision, recall, and F1-Score of 99.8%.

IV. Evaluation

A. Standard Metrics Results

The performance of three models RF, LightGBM, and CatBoost was evaluated in two phases. In the first phase, the models were tested separately on three distinct datasets representing different traffic types: Outbound, Inbound, and Internal.

For outbound traffic, all three models achieved perfect classification performance, obtaining 100% accuracy, precision, recall, and F1-Score. This indicates that outbound traffic has a distinguishable pattern and is detected by all models.

For inbound traffic, CatBoost outperformed other models with 90.79% accuracy, 90.57% precision, 90.79% recall, and an F1-Score of 90.63%. LightGBM followed closely with 90.29% accuracy, 90.05% precision, 90.29% recall, and an F1-Score of 90.08%, while RF performed lower results with 89.96% accuracy, 89.69% precision, 89.96% recall, and an F1-Score of 89.74%.

For internal traffic, all models again achieved perfect results with 100% across all evaluation metrics, indicating that internal traffic has a distinguishable pattern and is detected by all models.

In the second phase, all datasets from the first phase (inbound, outbound, and internal) were combined into a single comprehensive dataset (All Data) to assess the overall robustness of the models. LightGBM achieved the best overall performance with 94.08% accuracy, 93.94% precision, 94.08% recall, and an F1-Score of 93.90%. CatBoost performed comparably with 93.97% accuracy, 93.78% precision, 93.97% recall, and an F1-Score of 93.79%. RF recorded slightly lower but still strong performance, achieving 93.58% accuracy, 93.39% precision, 93.85% recall, and an F1-Score of 93.29%.

The near-perfect classification results observed for outbound and internal traffic are primarily attributed to the clearly distinguishable patterns present in these datasets. The data was carefully split into 70% training and 30% testing sets and 5-fold cross-validation was applied to ensure robust evaluation and to minimize the risk of data leakage. Feature selection (Table VI) was conducted using attributes that reflect real network behavior without directly encoding attack labels, which strengthens the validity and reliability of the learning process. Statistical validation, including the KS test, showed only minor distributional differences between the real and GenAI-generated datasets due to variations in attack types; however, these differences did not negatively affect the model's ability to distinguish traffic patterns.

TABLE VI. MODELS PERFORMANCE ACROSS FOUR DATASETS

Type	Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
Outbound	RF	100	100	100	100
Outbound	LightGBM	100	100	100	100
Outbound	CatBoost	100	100	100	100
Inbound	RF	89.96	89.69	89.96	89.74
Inbound	LightGBM	90.29	90.05	90.29	90.08
Inbound	CatBoost	90.79	90.57	90.79	90.63
Internal	RF	100	100	100	100
Internal	LightGBM	100	100	100	100
Internal	CatBoost	100	100	100	100
All	RF	93.58	93.39	93.85	93.29
All	LightGBM	94.08	93.94	94.08	93.90
All	CatBoost	93.97	93.78	93.97	93.79

Overall, these findings indicate that the dataset is highly consistent and well-aligned with the characteristics of the applied machine learning algorithms and their tuned parameters, demonstrating that the classifier model used was highly effective and well-suited to the structure of the data, resulting in strong and stable predictive performance. Finally, the combined dataset (inbound, outbound, and internal traffic) introduces greater complexity and variability, which is reflected in slightly reduced but still robust performance across all evaluated models.

B. Confusion Matrix Analysis

The performance of the three ML models was evaluated using confusion matrices across four datasets: Outbound, Inbound, Internal, and All traffic. For Outbound traffic, all three models achieved perfect classification across all classes (DHCP Amplification, DHCP Starvation, and Normal traffic), with True Positives (TP) and True Negatives (TN) equal to the total number of samples per class, and False Positives (FP) and False Negatives (FN) equal to zero. This indicates that outbound attacks are highly distinguishable and easily detectable Table VII.

TABLE VII. CONFUSION MATRIX FOR OUTBOUND DATA

Class	Model	TP	TN	FP	FN
DHCP Amplification	RF	400	800	0	0
DHCP Amplification	LGBM	400	800	0	0
DHCP Amplification	CatBoost	400	800	0	0
DHCP Starvation	RF	400	800	0	0
DHCP Starvation	LGBM	400	800	0	0
DHCP Starvation	CatBoost	400	800	0	0
Normal	RF	400	800	0	0
Normal	LGBM	400	800	0	0
Normal	CatBoost	400	800	0	0

For Inbound traffic, the models exhibited varying detection performance across attack types Table VIII. ICMP Direct and ICMP Smurf attacks were perfectly classified by all models, with no misclassifications. Application Layer Attacks showed minor variations, with CatBoost achieving the highest true positives (TP = 385) compared to RF (TP = 388) and LightGBM (TP = 378). SYN Flood and UDP Flood attacks exhibited more misclassifications, reflected by lower TP and higher FP and FN across models. Normal traffic was also classified accurately, with slight differences among the models. Overall, CatBoost generally demonstrated the best balance between TP, FP, and FN, highlighting its marginal advantage in detecting diverse inbound attacks, while all models performed reliably across most classes.

TABLE VIII. CONFUSION MATRIX FOR INBOUND DATA

Class	Model	TP	TN	FP	FN
Application Layer Attack	RF	388	1947	53	12
Application Layer Attack	LGBM	378	1969	31	22
Application Layer Attack	CatBoost	385	1982	18	15
ICMP Direct	RF	400	2000	0	0
ICMP Direct	LGBM	400	2000	0	0
ICMP Direct	CatBoost	400	2000	0	0
ICMP Smurf	RF	400	2000	0	0
ICMP Smurf	LGBM	400	2000	0	0
ICMP Smurf	CatBoost	400	2000	0	0
SYN Flood	RF	268	1912	88	132
SYN Flood	LGBM	269	1922	78	131
SYN Flood	CatBoost	277	1912	88	123
UDP Flood	RF	317	1922	78	83
UDP Flood	LGBM	331	1920	80	69
UDP Flood	CatBoost	326	1929	71	74
Normal	RF	386	1978	22	14
Normal	LGBM	389	1956	44	11
Normal	CatBoost	391	1956	44	9

For Internal traffic, which contains only two classes, all models achieved perfect classification, with TP and TN equal to the number of samples and FP and FN equal to zero, showing that internal attack patterns are highly distinguishable Table IX.

For the all traffic dataset, which combines outbound, inbound, and internal traffic, the confusion matrices indicate that the models maintain strong performance across different traffic types. While most classes achieve near-perfect classification, classes such as Normal, SYN Flood, UDP Flood, and Application Layer Attacks show minor misclassifications, reflecting the inherent complexity of these patterns when analyzed in a combined dataset Table X.

TABLE IX. CONFUSION MATRIX FOR INTERNAL DATA

Class	Model	TP	TN	FP	FN
ARP Spoofing	RF	400	400	0	0
ARP Spoofing	LGBM	400	400	0	0
ARP Spoofing	CatBoost	400	400	0	0
Normal	RF	400	400	0	0
Normal	LGBM	400	400	0	0
Normal	CatBoost	400	400	0	0

Overall, these results demonstrate that RF, LightGBM, and CatBoost are robust in detecting and classifying outbound, inbound, and internal network attacks, with

LightGBM and CatBoost achieving slightly higher TP and lower FP and FN values compared to RF.

TABLE X. CONFUSION MATRIX FOR ALL DATA

Class	Model	TP	TN	FP	FN
ARP Spoofing	RF	400	3200	0	0
ARP Spoofing	LGBM	400	3200	0	0
ARP Spoofing	CatBoost	400	3200	0	0
DHCP Amplification	RF	400	3200	0	0
DHCP Amplification	LGBM	400	3200	0	0
DHCP Amplification	CatBoost	400	3200	0	0
Application Layer Attack	RF	396	3140	60	4
Application Layer Attack	LGBM	390	3157	43	10
Application Layer Attack	CatBoost	393	3159	41	7
DHCP Starvation	RF	400	3199	1	0
DHCP Starvation	LGBM	400	3200	0	0
DHCP Starvation	CatBoost	400	3200	0	0
ICMP Direct	RF	400	3200	0	0
ICMP Direct	LGBM	400	3200	0	0
ICMP Direct	CatBoost	400	3200	0	0
ICMP Smurf	RF	400	3198	2	0
ICMP Smurf	LGBM	400	3200	0	0
ICMP Smurf	CatBoost	400	3200	0	0
Normal	RF	246	3140	60	154
Normal	LGBM	269	3140	60	131
Normal	CatBoost	268	3131	69	132
SYN Flood	RF	333	3118	82	67
SYN Flood	LGBM	335	3122	78	65
SYN Flood	CatBoost	335	3131	69	65
UDP Flood	RF	394	3174	26	6
UDP Flood	LGBM	393	3168	32	7
UDP Flood	CatBoost	387	3162	38	13

C. ROC Curve

The ROC curves were evaluated for all traffic directions and models. For outbound traffic Fig. 2 all models achieved an AUC of 1, indicating that a perfect discrimination between classes. In the inbound traffic dataset Fig. 3, the models performed exceptionally well, with RF, LGBM, and CatBoost achieving AUCs of 0.99, 0.99, and 0.99, respectively, demonstrating their strong ability to identify inbound attacks accurately. On the internal traffic dataset Fig. 4, all models achieved an AUC of 1, reflecting flawless classification performance. When considering the combined dataset encompassing all traffic Fig. 5 the models maintained high performance with AUCs of 0.99 for RF and 0.99 for both LGBM and CatBoost, underscoring their robustness and reliability across diverse network traffic scenarios.

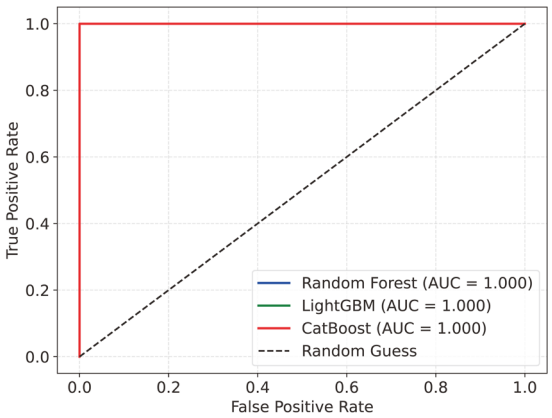

Fig. 2. ROC curves for outbound data.

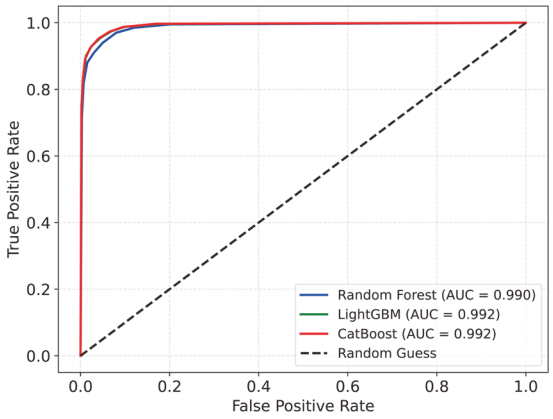

Fig. 3. ROC curves for inbound data.

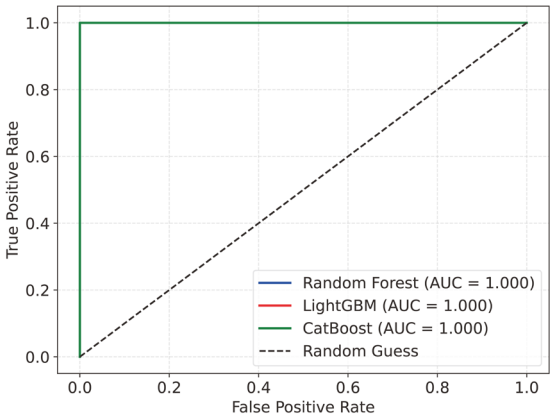

Fig. 4. ROC curves for internal data.

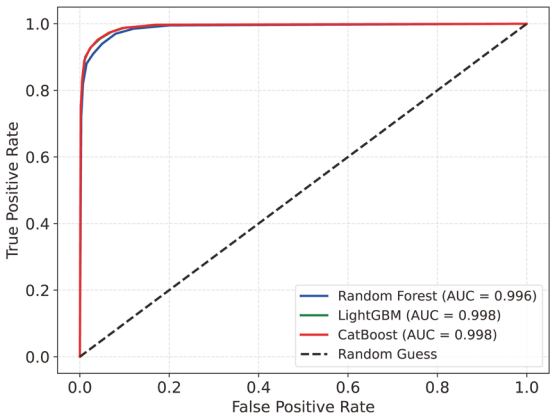

Fig. 5. ROC curves for all data.

V. DISCUSSION AND COMPARATIVE ANALYSIS

Table XI presents a comparative analysis between the proposed model and recent state-of-the-art studies on DDoS and IoT attack detection. This comparison highlights several critical research gaps in existing works and clearly positions the contribution of the proposed approach.

No existing study in the literature employs the INDDoS24 [2] dataset, making this work the first to evaluate and report results on this dataset. Prior studies predominantly rely on well-known but repeatedly used datasets such as CICIoT2023, CIC-IDS2017, CIC-DDoS-2019, and application-specific datasets. While

these datasets have been instrumental in earlier research, their extensive reuse raises concerns regarding dataset aging and reduced representativeness of contemporary attack behaviors, particularly in modern IoT environments.

Several recent papers report very high performance metrics, but most focus exclusively on inbound traffic attacks. This narrow scope limits their applicability in real-world IoT networks, where attacks may originate internally or be launched as outbound attacks, such as amplification or DHCP-based attacks. The proposed model explicitly addresses this limitation by supporting inbound, outbound, and internal traffic analysis, providing a more realistic and comprehensive security framework.

A limitation in multiple comparative works is that attack types are either partially specified or not mentioned at all, as observed in studies using CIC-IDS2017 and UNSWIoT2 datasets. The proposed model evaluates a diverse set of attack types including ARP spoofing, DHCP amplification, application-layer attacks, DHCP starvation, ICMP-based attacks, SYN floods, and UDP flood. Some prior works show the perfect accuracy but these results depend on only single direction traffic and narrow set of attack types, and are limited to one or two attacks only. The proposed model illustrates that detection of all attacks with strong performance of high accuracy, precision, recall, F1-Score. This confirms its ability on generalization and effectiveness rather than simply optimizing individual metrics.

In summary, existing studies rely on lagacy datasets with incomplete attack specifications and inbound only. The proposed approach introduces a novel dataset with multi-directional traffic and explicit attack coverage, Consequently, a stronger solid for efficient IoT network security is established.

TABLE XI. COMPARATIVE PERFORMANCE ACROSS DATASETS, ATTACK TYPES, AND TRAFFIC DIRECTIONS

Paper	Dataset	Attack Type	Traffic Direction	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
[29]	CICIoT2023	Slow Loris, HTTP, ICMP, SYN, UDP, TCP, RSTFIN, PSHACK Floods, ACK, UDP Fragmentation, Synonymous IP Flood	Inbound	99.95	99.95	99.95	99.95
[30]	CIC2023	TCP, SYN, SlowLoris, HTTP, UDP floods	Inbound	99.90	99.90	99.90	99.90
[31]	CIC-DDoS-2019	SYN Flood, UDP Flood, UDP-Lag, MSSQL attack, CharGen attack	Inbound	94	94.39	94.22	94.17
[32]	Application-Layer DDoS Dataset	DoS Slowloris, DDoS Hulk	Inbound	99.60	99.66	99.61	99.63
[33]	CIC-IDS2017	Not mentioned	-	99.80	99.84	99.78	99.54
[34]	UNSWIoT2	Not mentioned	-	100	100	100	100
Proposed Model	INDDoS24	ARP Spoofing, DHCP Amplification, Application Layer Attack, DHCP Starvation, ICMP Direct, ICMP Smurf, SYN Flood, UDP Flood	Outbound /Inbound /Internal	Outbound: 100% Inbound: 90.79% Internal: 100% All: 94.08%	Outbound: 100% Inbound: 90.57% Internal: 100% All: 93.94%	Outbound: 100% Inbound: 90.79% Internal: 100% All: 94.08%	Outbound: 100% Inbound: 90.63% Internal: 100% All: 93.90%

VI. RECOMMENDATIONS

Based on the performance of ML in detection. The following recommendations are proposed:

A. Model Selection for Deployment

1) Random forest

Random Forest exhibits excellent performance across all traffic types, achieving perfect scores on outbound and internal traffic with 100% accuracy, precision, recall, and F1-Score. For inbound traffic, it achieved 89.96% accuracy, 89.69% precision, 89.96% recall, and 89.74% F1-Score. On the full dataset, RF maintained strong overall performance with 93.58% accuracy, 93.39% precision, 93.85% recall, and 93.29% F1-Score, demonstrating its effectiveness in detecting diverse DDoS attacks in IoT networks.

2) LightGBM

In general-purpose deployment, LightGBM has highest overall performance on full dataset with 94.08% accuracy, 93.94% precision, 94.08% recall, and 93.90% F1-Score. Its strong performance is consistent across internal and outbound traffic as well, making it a reliable choice for detecting diverse IoT network attacks.

3) CatBoost

CatBoost demonstrates strong performance across all traffic types, achieving perfect scores on outbound and internal traffic with 100% accuracy, precision, recall, and F1-Score. For inbound traffic, it achieved 90.79% accuracy, 90.57% precision, 90.79% recall, and 90.63% F1-Score. On the full dataset, CatBoost maintained high overall performance with 93.97% accuracy, 93.78% precision, 93.97% recall, and 93.79% F1-Score, making it a reliable model for comprehensive DDoS attack detection in IoT networks.

B. Traffic Type-Specific Model Deployment

1) Outbound traffic (application layer), attack vectors:

(a) DHCP Starvation;
(b) DHCP Amplification.

All models showed 100% accuracy detecting these, so simpler models like RF are recommended for efficient real-time detection.

2) Inbound traffic (network, transport, and application layers), attack vectors include:

(a) SYN Flood and UDP Flood (Transport Layer);
(b) Application Layer Attacks (Application Layer);
(c) ICMP Smurf and ICMP Direct (Network Layer).

Due to the complexity, Catboost is recommended for superior detection performance.

3) Internal traffic (data link layer), attack vector includes:

ARP Spoofing: The three models achieved 100% detection accuracy, making any model suitable for internal traffic attack detection.

4) All attacks

LightGBM achieved the highest detection performance.

C. Integrated or Hybrid Deployment

A hybrid system can be designed in which traffic is routed through the most suitable model for each category:
(1) Outbound (Application Layer) → RF, LightGBM, CatBoost;
(2) Inbound (Network, Transport, Application Layers) → LightGBM;
(3) Internal (Data Link Layer) → RF, LightGBM, CatBoost.

This strategy making the best usage of computational resources with high detection accuracy.

VII. Future Directions

While this study illustrates the strong performance of ML in detecting multi-vector IoT DDoS attacks across inbound, outbound, and internal traffic, there is advanced research directions can increase scientific contribution and practical application in the real world.

A. Evaluation with Diverse and Heterogeneous Datasets

Future work should focus on extending the evaluation using diverse and heterogeneous datasets that include varying traffic patterns, attack distributions, and behavioral characteristics. This will enhance the assessment of model generalization across different data conditions and reduce dependency on specific dataset properties. Such an extension is essential to improve the robustness of detection models and ensure reliable performance under realistic and dynamic environments.

B. Real-Time Deployment and Online Learning

Future work should focus on real-time detection framework in IoT an environment where the traffic is continuous and the attack pattern are rapidly evolving. Integrating online and incremental learning will enable the models to dynamically adapt without being retrained and ensure accuracy.

Downtime and retraining costs are prohibitive in large-scale IoT networks; therefore, assessing latency, throughput, and resource utilization under live network conditions will be essential to confirm effectiveness and operational viability of the framework.

C. Temporal Deep Learning for Multi-Stage Attack Detection

Future work should move beyond tree-based models by exploring temporal deep learning architectures such as Gated Recurrent Units (GRU) and Temporal Convolutional Networks (TCN). Current models show strong performance, but they remain limited in capturing how network traffic evolves over time. Temporal models are better suited to learning sequential behavior and long-term traffic patterns, which are essential for detecting slow-rate, stealthy, and multi-stage DDoS attacks. By integrating time-aware features, future systems can enable earlier attack detection and improve robustness in complex and continuously evolving IoT traffic.

D. Federated and Privacy-Preserving Learning

Future work should explore federated learning mechanisms that enable collaborative model training across multiple IoT domains without requiring the sharing of raw network traffic. It will allow more scalable, general and private data set while building a robust multi-domain detection system. Aggregation and differential privacy can be adopted to protect sensitive information, reduce the risk of data leakage during training, and enforce compliance with privacy laws in practice.

E. Adversarial Robustness and Evasion Resistance

Recently, attackers target ML detection systems. Future researech should evaluate framework of the ML against adversarial attacks, including evasion, poisoning, and mimicry techniques.

Implementing models against adversarial, and model hardening strategies can enhance resilience against manipulated traffic designed to bypass detection.

Furthermore, systematically investigating how attacks transfer between models can provide valuable insights for building more comprehensive and robust defenses in IoT DDoS detection.

F. Cross-Domain and Cross-Dataset Generalization

Future research should explore the generalizability of the proposed framework across diverse IoT environments, such as smart cities, industrial IoT, healthcare IoT, and the Internet of Vehicles (IoV). Validating models across multiple datasets that include varied network topologies and device types will help assess robustness beyond controlled experimental settings. In addition, employing domain adaptation techniques can improve performance when transferring models across different IoT ecosystems, ensuring reliable multi-vector DDoS detection in large-scale, real-world deployments.

G. Autonomous Mitigation and Response Integration

Beyond detection, future research should focus on integrating the framework with automated mitigation and response mechanisms. Combining detection with approaches such as Software-Defined Networking (SDN), network slicing, or adaptive access control can enable rapid containment of attacks. Developing closed-loop systems that seamlessly link detection, decision-making, and response will move IoT security

toward fully autonomous, self-healing architectures, reducing the need for human intervention and enhancing resilience against multi-vector DDoS attacks.

H. Federated Learning Security and Poisoning Mitigation

Future research should consider the security challenges specific to Federated Learning in IoT environments. Since FL training relies on an "available but not visible" mechanism, malicious clients can inject poisoned updates, potentially corrupting the model's accuracy or achieving adversarial objectives. Recent studies highlight VagueGAN [28] as a poisoning attack model that leverages GANs to generate seemingly benign records containing carefully crafted adversarial noise, demonstrating high degradation effectiveness with minimal detectability. To counter such threats, future work can explore combined detection mechanisms, such as integrating Shapley Additive Explanations (SHAP) with Support Vector Machines (SVMs), to distinguish malicious clients from benign ones, thereby ensuring robust and secure federated model training.

VIII. CONCLUSION

The study proposed a comprehensive ML framework capable of detecting eight types of DDoS attacks across different layers of IoT networks. By integrating the INDDoS24 dataset with the GenAI dataset, the framework achieves extensive coverage of attack types across the application, transport, network, and data link layers. Experimental results demonstrate that the proposed model is both robust and generalizable, with LightGBM achieving the highest overall performance on the combined dataset.

Compared to existing studies, this methodology effectively handles multi-directional traffic, a wide variety of attack types, and provides interpretable results suitable for practical deployment.

Overall, the study delivers a reliable and comprehensive framework for securing IoT networks against diverse DDoS attacks, underscoring the value of multi-layered, multi-directional detection strategies.

APPENDIX A: FINAL FEATURES AND DESCRIPTION

TABLE AI. FINAL FEATURES WITH ITS DESCRIPTION

Feature	Description
Source Port	Port number used by the source device for communication.
Destination Port	Port number used by the destination device for communication.
Protocol	Communication protocol used (e.g., TCP, UDP, ICMP).
Anomaly Score	A numerical score indicating the likelihood of the traffic being anomalous.
Device Type	Type of device generating the network traffic (e.g., camera, sensor).
Operating System	Operating system running on the device (e.g., Linux, Windows).
Attack Type	Type of attack detected during network traffic analysis, if applicable.
Target Device	Device targeted by the attack, if applicable.
Hour	Hour extracted from the timestamp to represent time of day.
is_weekend	Binary indicator showing whether the timestamp falls on a weekend.
Total Unique Count	Sum of unique source and destination IPs in the flow.
packet_flow_size	Total size of the network flow in bytes.
rate_packet_size	Ratio of bytes transferred per second in the flow.
Payload/Packet Size Ratio	Ratio between payload length and total packet size.
bytes_to_packets_ratio	Ratio of total bytes to the number of packets in the flow.
Log Payload Length	Logarithmic transformation of payload length.
attack_dur_binned	Binned representation of attack duration into intervals.
int_arr_time_binned	Binned representation of packet inter-arrival time.
Device Risk Factor	Risk level assigned based on device type and operating system vulnerabilities.
Network Zone	Classification of network segment (e.g., internal, DMZ, external).
Network Zone Score	Score representing the security posture of a network zone.

TABLE AII. FEATURES DESCRIPTION

Feature	Description
Timestamp	Time when the network traffic flow was recorded.
Source IP	IP address of the device sending the data.
Destination IP	IP address of the device receiving the data.
Source Port	Port number used by the source device.
Destination Port	Port number used by the destination device.
Protocol	Communication protocol used (e.g., TCP, UDP, ICMP).
Packet Size	Size of each transmitted data packet in bytes.
Payload Length	Length of actual transmitted data excluding headers.
Flow Duration	Total duration of the network flow from start to end.
Bytes in Flow	Total number of bytes transferred in the flow.
Packets in Flow	Total number of packets exchanged in the flow.
Average Packet Size	Average size of packets within the flow.
Inter-Arrival Time	Time interval between consecutive packets in the flow.
Rate of Packets	Rate of packet transmission within the flow.
Unique Source Count	Number of unique source IPs involved in the flow.
Unique Destination Count	Number of unique destination IPs involved in the flow.
Anomaly Score	Score indicating likelihood of anomalous traffic.
Device Type	Type of device generating the traffic (e.g., camera, sensor).
Operating System	Operating system of the device (e.g., Linux, Windows).
Firmware Version	Firmware version installed on the generating device.
Attack Type	Type of attack detected during traffic analysis, if any.
Attack Duration	Duration of the attack if present in the flow.
Target Device	Device targeted by the attack, if applicable.

CONFLICT OF INTEREST

The authors declare no conflict of interest.

AUTHOR CONTRIBUTIONS

Rania A. Al-Ali conducted the research, developed the methodology, implemented the models, performed the experiments and data analysis, and prepared the original manuscript draft. Mohammad Alnabhan contributed to the methodology, technical guidance, and critical review of the manuscript. Qasem Abu Al-Haija conceived and supervised the study, provided scientific guidance, validated the results, revised the manuscript, and served as the corresponding author. All authors read and approved the final manuscript.

REFERENCES

[1] C. Kolias, G. Kambourakis, A. Stavrou *et al.,* "DDoS in the IoT: Mirai and other botnets," *Computer*, vol. 50, no. 7, pp. 80–84, 2017.

[2] Dataset Engineer. (2024). Inddos24 dataset: IoT network traffic for ddos attack detection. [Online]. Available: https://www.kaggle.com/datasets/ datasetengineer/inddos24-dataset

[3] M. Miettinen, S. Marchal, I. Hafeez *et al.*, "IoT sentinel: Automated device-type identification for security enforcement in IoT," in *Proc. 2017 IEEE 37th International Conf. on Distributed Computing Systems (ICDCS),* 2017, pp. 2177–2184.

[4] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[5] Wikipedia. (2025). Random forest. [Online]. Available: https://en.wikipedia.org/wiki/ Random_forest

[6] X. Tan, S. Su, Z. Huang *et al.,* "Wireless sensor networks intrusion detection based on SMOTE and the random forest algorithm," *Sensors*, vol. 19, no. 1, 203, 2019.

[7] R. Devi and M. Abualkibash, "Intrusion detection system classification using different machine learning algorithms on KDD-99 and NSL-KDD datasets—A review paper," *International Journal of Computer Science and Information Technology*, vol. 11, no. 3, pp. 65–80, 2019.

[8] N. Farnaaz and M. A. Jabbar, "Random forest modeling for network intrusion detection system," *Procedia Computer Science*, vol. 89, pp. 213–217, 2016. doi: 10.1016/j.procs.2016.06.047

[9] G. Ke, Q. Meng, T. Finley *et al.*, "LightGBM: A highly efficient gradient boosting decision tree," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[10] Wikipedia (2025). LightGBM. [Online]. Available: https://en.wikipedia.org/wiki/LightGBM

[11] S. Seth, G. Singh, and K. K. Chahal, "A novel time-efficient learning-based approach for a smart intrusion detection system," *Journal of Big Data*, vol. 8, no. 111, 2021. doi: 10.1186/s40537-021-00498-8

[12] J. Vitorino, R. Andrade, I. Praça *et al.*, "A comparative analysis of machine learning techniques for IoT intrusion detection," in *Proc. International Symposium on Foundations and Practice of Security*, 2022, pp. 191–207. doi: 10.1007/978-3-031-08147-7_13

[13] Wikipedia (2025). CatBoost. [Online]. Available: https://en.wikipedia.org/wiki/CatBoost

[14] L. Prokhorenkova, G. Gusev, A. Vorobev *et al.*, "CatBoost: Unbiased boosting with categorical features," *Advances in Neural Information Processing Systems*, vol. 31, 2018. doi: 10.5555/3327757.3327770

[15] A. V. Dorogush, V. Ershov, and A. Gulin, "CatBoost: gradient boosting with categorical features support," arXiv preprint arXiv: 1810.11363, 2018.

[16] H. Mukhtar, K. Salah, and Y. Iraqi, "Mitigation of DHCP starvation attack," *Computers & Electrical Engineering*, vol. 38, no. 5, pp. 1115–1128, 2012. doi: 10.1016/j.compeleceng.2012.06.005

[17] S. Tripathi and M. M. A. Hubballi, "Exploiting DHCP server-side IP address conflict detection," in *Proc. IEEE Int. Conf. Adv. Netw. Telecommun. Syst. (ANTS)*, 2015. doi: 10.1109/ANTS.2015.7413661

[18] N. Trabelsi and M. El-Hajj, "On investigating ARP spoofing security solutions," *Int. J. Inf. Privacy, Secur. Integrity*, 2010. doi: 10.1504/IJIPT.2010.032618

[19] S. Y. Nam, S. Jurayev, S.-S. Kim *et al.*, "Mitigating ARP poisoning-based man-in-the-middle attacks in wired or wireless LAN," *EURASIP Journal on Wireless Communications and Networking*, vol. 2012, no. 89, pp. 1–10, 2012.

[20] A. Srivastava, S. Tiwari, D. Kumar *et al.*, "Finding of DDoS attack in IoT-based networks using ensemble technique," in *Proc. Int. Conf. Intelligent Systems for Cybersecurity (ISCS)*, 2024, pp. 1–4.

[21] I. Kumar, M. Bohra, N. Mohd *et al.,* "DDoS and IoT botnet malware identification using machine learning and deep learning models," in *Proc. 2nd Int. Conf. Advances in Information Technology (ICAIT)*, 2024, pp. 1–6.

[22] A. I. El Sayed, M. Abdelaziz, M. Hussein *et al.,* "DDoS mitigation in IoT using machine learning and blockchain integration," *IEEE Networking Letters*, vol. 6, no. 2, pp. 152–155, 2024.

[23] A. Srivastava, V. Sawan, K. Jugnu *et al.,* "IoT attack detection using LSTM model," in *Proc. 2nd Int. Conf. Intelligent Data Communication Technologies and Internet of Things (IDCIoT)*, 2024, pp. 7–10.

[24] A. Srivastava, S. Tiwari, P. K. Saini *et al.,* "Attack detection and mitigation in IoT using SVM," in *Proc. 2nd Int. Conf. Augmented Intelligence and Sustainable Systems (ICAISS)*, 2023, pp. 1547–1551.

[25] J. Bhayo, S. A. Shah, S. Hameed *et al.,* "Towards a machine learning-based framework for DDoS attack detection in software-defined IoT networks," *Engineering Applications of Artificial Intelligence*, vol. 123, 106432, 2023.

[26] K. S. Kumavat and J. Gomes, "Multi-layer DDoS attacks detection with mitigation in IoT-enabled sensor network," *Cybersecurity*, vol. 8, no. 1, 72, 2025.

[27] S. J. Praveena and S. Sudha, "Arp spoofing attack detection to prevent ddos attack," in *Proc. 5th Int. Conf. Trends in Material Science and Inventive Materials (ICTMIM)*, 2025, pp. 1054–1059.

[28] E. Nowroozi, Y. Habibi, M. S. Naveed *et al.,* "Dual perspectives on GAN-based data poisoning in federated learning: Vaguegan attacks and data poisoning detection," in *Adversarial Example Detection and Mitigation Using Machine Learning*, Springer Nature Switzerland, 2026, pp. 249–260

[29] R. B. Mallela, O. Srinivas, E. A. Goud *et al.,* "Optimizing IoT DDoS detection with hybrid feature selection and ensemble learning," in *Proc. 2025 3rd International Conf. on Sustainable Computing and Data Communication Systems (ICSCDS)*, 2025, pp. 1004–1008.

[30] A. Berqia, H. Bouijij, A. Merimi *et al.,* "Detecting DDoS attacks using machine learning in IoT environment," in *Proc. 2024 International Conference on Intelligent Systems and Computer Vision (ISCV)*, 2024, pp. 1–8.

[31] A. Rashid, S. M. K. -U. -R. Raazi, and S. Ali, "Detection and mitigation of DDoS attacks on IoT networks using machine learning," in *Proc. 2024 26th International Multi-Topic Conf. (INMIC)*, 2024, pp. 1–6.

[32] A. El Sayed, M. Abdelaziz, M. Hussein *et al.,* "DDoS mitigation in IoT using machine learning and blockchain integration," *IEEE Networking Letters*, vol. 6, no. 2, pp. 152–155, 2024.

[33] Md. Asad and R. K. Tiwari, "Artificial neural network technique for DDoS attack prediction in cyber-IoT system," in *Proc. 2023 3rd International Conf. on Technological Advancements in Computational Sciences (ICTACS)*, 2023, pp. 497–501.

[34] A. S. Sa'Idi and A. M. Jamil, "IoT DDoS attack detection system using machine learning classification techniques," in *Proc. 2023 IEEE 21st Student Conference on Research and Development (SCOReD)*, 2023, pp. 234–239.