Zero-Day Attack Detection in 6G Networks using Tree-Based Machine Learning Models

Maria Gabriela L. Damasceno[1], David M. Cavalcanti[1], Jamilson Dantas[1], Dalton C. G Valadares[2]
Andson M. Balieiro[1]

***Abstract*— The advent of Sixth Generation (6G) networks brings profound integration with Artificial Intelligence (AI). However, it also expands the attack surface, introducing critical cybersecurity threats such as Zero-Day attacks, which exploit previously unknown vulnerabilities. In this context, this paper evaluates the effectiveness of tree-based machine learning models, such as CatBoost, Decision Tree, and Random Forest, in detecting these threats, considering the relationship between detection accuracy and computational resource consumption. The results indicade that the CatBoost model performs best for Zero-Day attack detection, while the Decision Tree model is the most balanced in terms of attack detection and efficiency for 6G networks.**

Keywords— Performance Evaluation, Zero-Day Attack Detection, 6G Networks, Explainable Artificial Intelligence (XAI)

I. Introduction

The Sixth Generation (6G) of Mobile Networks is expected to provide Artificial Intelligence (AI)-native integration, enabling intelligent network management, automation, optimization, and security at all architectural layers [1]. However, this evolution also amplifies the cyber threat landscape, as advances in AI and processing capabilities empower attackers to design increasingly sophisticated attacks and exploit emerging vulnerabilities. Among these threats, zero-day attacks pose a considerable risk, as they can circumvent conventional defense mechanisms by exploiting previously unknown vulnerabilities and leveraging attacks that have not yet been identified by the network [2].

Detecting zero-day threats in real-time is a major challenge for 5G/6G networks. Despite the availability of AI/ML solutions, significant gaps persist. For example, some works [1], [3] only target known attacks, while others focusing zero-day scenarios [4], [5] frequently overlook the energy and computational costs, essential for IoT and sustainable 6G networks. Furthermore, relying on aggregate metrics [4], [6] can mask poor zero-day performance, and using non-mobile or single datasets limits generalization to real 5G/6G environments. Consequently, there is a need for comprehensive research that balances accuracy with efficiency and sustainability metrics, such as resource usage, energy consumption, and execution time, validated on 5G/6G datasets.

In this context, this paper focuses on tree-based models, specifically CatBoost, Random Forest, and Decision Trees, as an alternative to computationally intensive neural networks. These models are inherently interpretable, aligning with the principles of eXplainable Artificial Intelligence (XAI) by enabling transparent decision-making required for trustworthy AI [7] and 5G/6G architectures.

To evaluate these models, the study uses two datasets collected from real mobile networks [8], [9], encompassing both detection performance (accuracy, precision, recall, F1-score, and zero-day detection ratio) and computational efficiency (CPU usage, memory consumption, execution time, and energy consumption). This analysis highlights the trade-off between zero-day generalization and the practical feasibility of deploying such solutions in

[1]Maria Gabriela L. Damasceno, David M. Cavalcanti, Jamilson Dantas, and Andson M. Balieiro are with the Centro de Informática (CIn), Universidade Federal de Pernambuco (UFPE), Recife, PE, Brazil {mgld, djmc, jrd, amb4}@cin.ufpe.br

[2]Dalton Cézane Gomes Valadares is with the Universidade Federal da Paraíba (UFPB), Bananeiras, Brazil. daltoncezane@gmail.com

6G environments. Ultimately, the findings identify the model with superior detection performance despite its higher resource consumption, as well as the most balanced solution for real-world deployment.

This paper is structured as follows: Section II presents related works, and Section III establishes the background on models and metrics. Section IV details the methodology, including dataset selection, processing, and the experimental setup. Section V presents the results. Section VI concludes the paper and proposes some future works.

II. Related Work

AI-based security for 6G is an active field, yet gaps in zero-day detection and performance persist. For instance, Federated Learning (FL) models [1], [3] successfully detect DDoS in B5G/O-RAN contexts but remain limited to known threats and overlook resource consumption and response time. Regarding zero-day attacks, [10] achieves 97.78% detection using a sandbox-based SDN solution in Mininet, while [11] reaches 91.75% accuracy via deep transductive transfer learning on NSL-KDD and CIDD datasets. Additionally, the authors in [12] proposes a double Autoencoder (AE) trained on benign and known traffic to identify unprecedented malicious flows.

Other works provide comparative analysis for zero-day scenarios [4], [6], [12], [5], but often omit resource consumption or real mobile network data. Recently, [13] presents a memory-efficient algorithm for IoT systems, but energy consumption analysis remains unexplored. The work [14] evaluates the performance and resource usage for zero-day detection in 6G networks, but focuses only on neural networks and did not analyze energy consumption.

Compared to existing solutions, this work evaluates three tree-based models using real 5G and Open RAN datasets [8], [9] with different attacks. Unlike prior research, this analysis combines general and zero-day attack detection with a detailed assessment of computational performance metrics and energy consumption, ensuring a comprehensive analysis of their effectiveness and feasibility for mobile networks.

III. Background

This section presents the tree-based models and the metrics used to evaluate their performance and resource consumption within 6G networks.

A. Tree-based models

Tree-based models, such as Decision Tree, Random Forest, and CatBoost, recursively partition data into subsets to create a hierarchical structures. Their interpretability and capacity for handling non-linear relationships make them ideal for anomaly detection and classification.

A Decision Tree is a non-parametric model characterized by a flexible structure that adapts to the data, partitioning the search through binary tests. One of the main advantage is its interpretability, as the logical path for each classification is explicitly stated [15]. Random Forest is an ensemble learning method that combines multiple independent decision trees using Bootstrap Aggregating [16]. This approach mitigates overfitting, a common limitation in individual decision trees, by reducing the variance of the final model. Finally, CatBoost (Categorical Boosting) uses the Gradient Boosting Decision Tree (GBDT) framework to process categorical variables without extensive manual preprocessing [17].

B. Attack Detection Metrics

In binary classification for attack detection, the concepts of True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) are fundamental. Correct classifications are defined as TP for positive instances and TN for negative ones. Conversely, an FP occurs when a negative instance is incorrectly flagged as positive. Finally, FN represents a failure to detect a positive instance. To evaluate model effectiveness, particularly against novel threats, the metrics summarized in Table I are employed.

C. Resource Consumption Metrics

To evaluate 6G feasibility, the following computational resource consumption metrics are defined: (i) CPU Usage as the average utilization percentage; (ii) Memory Consumption (MB) as the required RAM; (iii) Execution Time (s) for training or inference; and (iv) Energy (J), estimated

TABLE I
PERFORMANCE METRICS FOR ATTACK DETECTION

Metric	Description	Formula
Accuracy	Total correct rate [5]	$\frac{TP+TN}{TP+FP+TN+FN}$
Precision	Prediction reliability [4]	$\frac{TP}{TP+FP}$
Recall	Detection sensitivity [4]	$\frac{TP}{TP+FN}$
F1 Score	Balanced measure [5]	$\frac{2\times(Prec\times Rec)}{Prec+Rec}$
ZD_{DR}	Zero-day Accuracy [5]	$\frac{TP_{ZD}+TN_{ZD}}{Total_{ZD}}$

through a cubic model where consumption grows with frequency and load [18], as defined in Eq. 1:

$$E = z \cdot (C \cdot \theta \cdot f)^3 \cdot t \quad (1)$$

where C is the avg. CPU usage (%), f is the frequency (2.0 GHz), θ is the load factor (1.0), z is the normalization coefficient (1×10^{-27}), and t is the execution time (s).

IV. METHODOLOGY

This section describes the methodological aspects adopted, including dataset characteristics, pre-processing procedures, the zero-day attack evaluation strategy, and model configurations.

A. Datasets and Pre-processing

The efficacy of AI models in 5G/6G scenarios depends on high-quality data reflecting mobile network complexities, such as high-density traffic and specific signaling protocols. This work utilizes the 5G-NIDD [8] and NetsLab-5G Open RAN [9] datasets, which provide metrics from 5G infrastructures and physical Open RAN testbeds. The 5G-NIDD attacks, include HTTP, ICMP, and SYN Floods, alongside scanning techniques (SYN/UDP Scan) and Slowrate DoS. Complementarily, NetsLab-5G targets O-Cloud threats and application-layer vulnerabilities, including TCP ACK Flood, Slowloris, XSS, SQL Injection, and OS Fingerprinting.

Data preparation involved: (i) removing null values for consistency, (ii) applying SMOTE[1] to mitigate training bias through class balancing, and (iii) partitioning data into 80% for training and 20% for testing. This process resulted in 243,178 samples for 5G-NIDD and 344,764 for NetsLab-5G for the evaluation of the tree-based models[2].

[1]https://thecontentfarm.net/smote-for-handling-imbalanced-data/

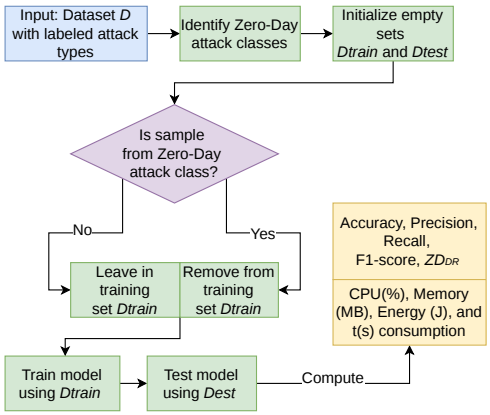

Fig. 1. Model Flow for Zero-day Attack Evaluation

B. Zero-Day Attack Evaluation Flow

The experimental procedure to evaluate the models' detection capability against unprecedented threats is shown in Figure 1. To simulate a Zero-Day scenario, an iterative Leave-One-Out approach was adopted. In each run, a specific attack class (e.g., HTTP Flood) is entirely removed from the training set (D_{train}) and reserved exclusively for the test set (D_{test}), ensuring the model has no prior exposure to that specific threat pattern.

D_{train} contains only benign traffic and known attacks, while D_{test} aggregates benign samples, known attacks, and the selected ZD samples. This strategy allows a rigorous assessment of the model's generalization capacity in 5G/6G environments. Following the training, the models are evaluated using detection metrics, including ZD_{DR}, and computational metrics such as CPU usage, memory, energy consumption, and execution time.

C. Model Configuration

The Decision Tree and Random Forest models utilize Python Scikit-learn libraries, while Catboost uses the CatBoostClassifier [19]. The Decision Tree

[2]https://github.com/gabrieladamasceno/Tree-Models/

is a supervised classifier configured with a maximum depth of 10 to prevent overfitting and a random seed of 42 to ensure consistency and reproducibility. Training uses the standard Recursive Partitioning method to maximize class separation. The Random Forest model combines multiple decision trees built from random subsets of samples and attributes. It was set with 100 independent decision trees, each with a maximum depth of 10, to enhance generalization. Finally, the Catboost model, a gradient boosting algorithm designed to efficiently handle categorical variables and reduce common problems such as overfitting and data leakage, is configured with 500 iterations and a 0.05 learning rate. To balance error correction and learning speed, the Catboost tree depth is limited to 6 levels, utilizing the CrossEntropy loss function for binary classification.

V. Results and Discussion

This section presents the results obtained. For each dataset and model, a multidimensional analysis compares zero-day attack detection effectiveness (Accuracy, Precision, Recall, F1-score, and ZD_{DR}) and computational efficiency (Execution Time, CPU, Memory, and Energy). This multidimensional analysis normalizes the results of all metrics, enabling a comparative analysis. The experiments were conducted using Google Colaboratory (Colab)[3], a cloud platform for developing and executing Jupyter notebooks[4]. This platform features a 2.0 GHz Intel® Xeon® CPU, 16 GB and provides native support for essential libraries, including NumPy, SciPy, Matplotlib, Pandas, and Seaborn.

Fig. 2 presents the 5G-NIDD accuracy results for the evaluated models considering different attack types as zero day ones. While ICMP Flood, SYN Scan, and Slow Rate DoS are identified with 99% accuracy by all models, the UDP Flood attack resulted in almost total detection failures. This suggests either high statistical similarity between malicious and benign traffic or significant divergence in patterns compared to other attack types. It is also important to note that only Catboost maintains performance above 50% for the TCP Connect Scan attack. In contrast, Decision Tree and Random Forest show significant recognition difficulties. This indicates that Catboost's structural complexity better supports the detection of subtle anomalous patterns.

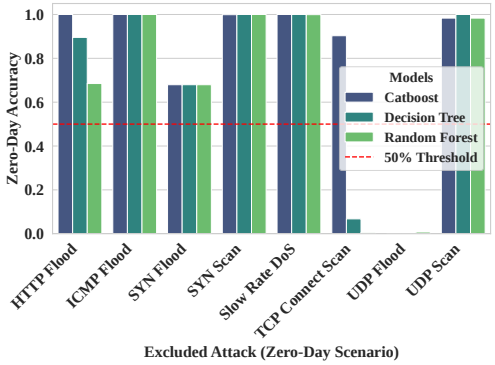

Fig. 2. Zero-Day Accuracy by Attack Type and Model - 5G-NIDD

To compare metrics with different units, such as accuracy (%), energy consumption (J), and memory usage (MB), the Min-Max Scaling method was applied to normalize all values into a common 0.0 to 1.0 range. In this context, the maximum value observed for each metric across all models serves as the 1.0 upper limit. Consequently, performance dimensions such as Accuracy, Precision, Recall, F1-score, and ZD_{DR} indicate higher effectiveness as they approach 1.0. Conversely, for the resource consumption dimensions (CPU, Memory, Execution Time, and Energy), 1.0 represents the highest operational impact observed during the experiment.

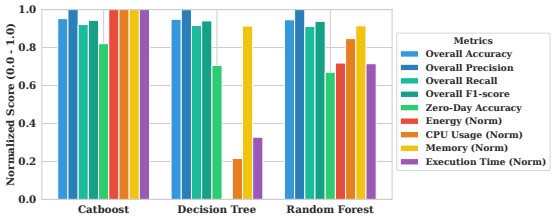

Fig. 3. Multidimensional Analysis: Performance vs. Resource Consumption - 5G-NIDD

Fig. 3 presents the multidimensional analysis of the 5G-NIDD dataset and reveals a clear contrast between detection performance and computational

[3]https://colab.research.google.com/
[4]https://jupyter.org/

efficiency. While Catboost, Decision Tree, and Random Forest models achieve excellent results in attack detection, their computational resource consumption differs significantly. Catboost stands out as the most computationally expensive model, reaching peak levels in CPU, memory, execution time, and energy consumption. In contrast, Decision Tree is the most sustainable and efficient model for resource-constrained environments, exhibiting the lowest energy and processing impact without significantly compromising overall predictive metrics.

In the NetsLab-5G dataset, the models face significant challenges in detecting application-layer and brute-force attacks as zero-day threats (ZD_{DR}), as shown in Fig. 4. Unlike flood attacks such as SYN and HTTP Flood, which were detected with almost 99% accuracy by all models, categories such as SQL Injection, SSH, and XSS showed poor performance, falling well below the 50% threshold. Furthermore, performance varies between the models for attacks such as UDP and UDP Port Scan. Random Forest demonstrated superior detection capacity compared to Decision Tree, suggesting that using multiple trees helps identify more dispersed anomalous patterns. The second attack case showed that this model did not perform well, with CatBoost performing better.

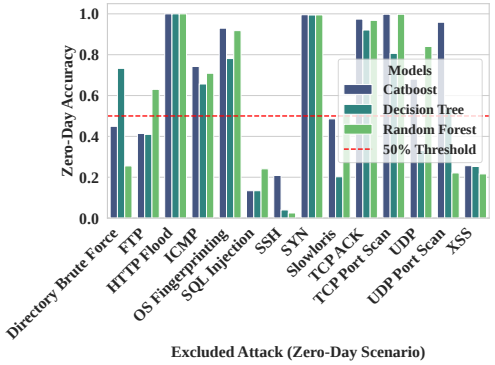

Fig. 4. Zero-Day Accuracy by Attack Type and Model - NetsLab-5G

Fig. 5 presents the multidimensional analysis for the NetsLab-5G dataset. While detection metrics (Accuracy, Precision, Recall, and F1-score) remain high and consistent across all models, their resource footprints differ. Catboost is the most resource-intensive, operating at peak limits for Energy, CPU, Memory, and Execution Time. Conversely, Decision Tree is the most efficient alternative, presenting significantly lower energy consumption due to its minimal execution time in both training and testing.

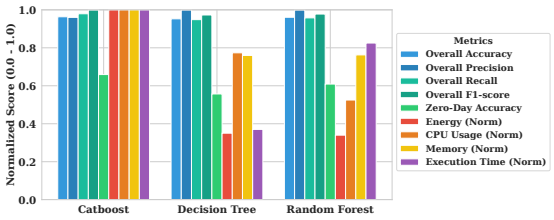

Fig. 5. Multidimensional Analysis: Performance vs. Resource Consumption - NetsLab-5G

Table II demonstrates that Catboost is the most demanding model, requiring up to 1,342.02 J and 99.16% CPU usage during training, significantly exceeding its counterparts (NetsLab-5G dataset). In contrast, Random Forest presents a balanced profile with intermediate energy costs, while Decision Tree is the most lightweight option, completing training with minimal power and time. During the testing phase, all three models exhibit high operational efficiency, with energy consumption dropping to the milliJoule (mJ) scale and inference times staying under 3.46 seconds.

TABLE II
AVERAGE RESOURCE CONSUMPTION

Model	Phase	CPU (%)	Mem (MB)	Energy	Time (s)
Dataset: 5G-NIDD					
DT	Train	49.95	3,220.07	8.82 J	8.59
	Test	1.17	3,257.35	0.03 mJ	1.14
RF	Train	49.88	3,262.39	125.71 J	126.64
	Test	5.54	3,237.56	12.17 mJ	2.47
CB	Train	99.00	3,805.23	711.36 J	93.11
	Test	5.56	3,570.80	11.74 mJ	3.46
Dataset: NetsLab-5G					
DT	Train	49.99	3,127.95	14.13 J	14.13
	Test	2.72	3,129.73	1.96 mJ	1.13
RF	Train	49.95	3,136.86	314.79 J	315.68
	Test	1.84	3,141.52	1.90 mJ	2.52
CB	Train	99.16	4,090.29	1,342.02 J	172.12
	Test	3.51	4,119.51	5.61 mJ	3.05

Across both datasets, Catboost consistently maintains the largest memory footprint, peaking at approximately 4.1 GB. In contrast, Decision Tree and Random Forest operate within more con-

strained memory limits of around 3.2 GB. These findings highlight the need to consider the trade-off between performance and resource consumption. Although CatBoost demonstrated higher performance in terms of accuracy, Decision Tree showed lower computational resource consumption, potentially making it more suitable for operation on mobile network devices.

VI. Conclusion and Future Works

This paper addressed Zero-Day attack detection in 5G/6G networks using tree-based models. A data processing methodology was defined to ensure that novel attacks were excluded from the training phase, enabling an evaluation of the models on both known and previously unseen samples. The work assessed the generalization capability and operational efficiency of Random Forest, Decision Tree, and CatBoost models, focusing on computational resource consumption and sustainability, which are essential for 6G networks. Results show that CatBoost and Random Forest achieve superior detection performance. At the same time, Decision Tree offers a competitive but slightly lower performance. In terms of computational efficiency, the Decision Tree is the fastest and most efficient model, offering the best trade-off for resource-constrained environments. Future work will explore optimization techniques to reduce computational and memory consumption, investigate generalization capability to new attack families, and evaluate alternative approaches to improve Zero-Day attack detection.

References

[1] S. Kianpisheh and T. Taleb, "Collaborative federated learning for 6G with a deep reinforcement learning based controlling mechanism: A DDoS attack detection scenario," *IEEE Trans. on Network and Service Management*, 2024.

[2] A. J. G. De Azambuja, C. Plesker, K. Schützer, R. Anderl, B. Schleich, and V. R. Almeida, "Artificial intelligence-based cyber security in the context of industry 4.0—a survey," *Electronics*, vol. 12, no. 8, p. 1920, 2023.

[3] C. Benzaïd, F. M. Hossain, T. Taleb, P. M. Gómez, and M. Dieudonne, "A federated continual learning framework for sustainable network anomaly detection in O-RAN," in *IEEE Wireless Comm. and Net. Conf. (WCNC)*, 2024.

[4] S. Ali, S. U. Rehman, A. Imran, G. Adeem, Z. Iqbal, and K.-I. Kim, "Comparative evaluation of AI-based techniques for Zero-Day attacks detection," *Electronics*, vol. 11, no. 23, p. 3934, 2022.

[5] M. Sarhan, S. Layeghy, M. Gallagher, and M. Portmann, "From zero-shot machine learning to Zero-Day attack detection," *Intl. Journal of Information Security*, 2023.

[6] K. Hamid, M. W. Iqbal, M. Aqeel, X. Liu, and M. Arif, "Analysis of techniques for detection and removal of Zero-Day attacks (ZDA)," in *International Conference on Ubiquitous Security*. Springer, 2022, pp. 248–262.

[7] E. Commission, "Ethics guidelines for trustworthy ai," European Commission - High Level Expert Group on AI, Tech. Rep., 2018.

[8] S. Samarakoon, Y. Siriwardhana, P. Porambage, M. Liyanage, S.-Y. Chang, J. Kim, J. Kim, and M. Ylianttila, "5G-NIDD: A comprehensive network intrusion detection dataset generated over 5G wireless network," *arXiv preprint arXiv:2212.01298*, 2022.

[9] A. Civciss, V. Ravihansa, F. A. Zadeh, C. Sandeepa, and M. Liyanage, "Silent signals, loud threats: Using dApps for radio signal intelligence-based intrusion detection in 5G O-RAN."

[10] H. Al-Rushdan, M. Shurman, S. H. Alnabelsi, and Q. Althebyan, "Zero-Day attack detection and prevention in software-defined networks," in *International Arab Conference on Information Technology (ACIT)*, 2019.

[11] N. Sameera and M. Shashi, "Deep transductive transfer learning framework for Zero-Day attack detection," *ICT Express*, vol. 6, no. 4, pp. 361–367, 2020.

[12] C. Redino, D. Nandakumar, R. Schiller, K. Choi, A. Rahman, E. Bowen, A. Shaha, J. Nehila, and M. Weeks, "Zero day threat detection using graph and flow based security telemetry," in *Intl. Conf. on Computing, Communication, and Intelligent Systems (ICCCIS)*. IEEE, 2022.

[13] P. R. Agbedanu, S. J. Yang, R. Musabe, I. Gatare, and J. Rwigema, "A scalable approach to internet of things and industrial internet of things security: Evaluating adaptive self-adjusting memory K-Nearest neighbor for Zero-Day attack detection," *Sensors*, vol. 25, no. 1, p. 216, 2025.

[14] M. G. L. Damasceno, C. B. B. de Souza, A. M. Balieiro *et al.*, "Evaluating MLP and Autoencoder models for Zero-Day attack detection in 6G networks," in *Simpósio Brasileiro de Segurança da Informação e de Sistemas Computacionais (SBSeg)*. SBC, 2025, pp. 384–400.

[15] P. Yoshioka, M. Damasceno, A. Balieiro, S. N. Swain, and E. Alves, "A decision-tree solution for the outdated CQI feedback problem in 5G networks," in *Brazilian Symp. on Computing Systems Engineering (SBESC)*. IEEE, 2024.

[16] A. Gupta and R. Simon, "Enhancing security in cloud computing with anomaly detection using Random Forest," in *Intl. Conference on Reliability, Infocom Technologies and Optimization (ICRITO)*. IEEE, 2024.

[17] L. Prokhorenkova, G. Gusev, A. Vorobev, A. V. Dorogush, and A. Gulin, "Catboost: unbiased boosting with categorical features," in *Proc. of the 32nd Intl. Conf. on Neural Information Processing Systems*, ser. NIPS'18. Red Hook, NY, USA: Curran Associates Inc., 2018, p. 6639–6649.

[18] W. Zhang, Y. Wen, D. O. Wu *et al.*, "Collaborative task execution in mobile cloud computing under a stochastic wireless channel," *IEEE Transactions on Wireless Communications*, vol. 14, no. 1, pp. 81–93, 2014.

[19] M. Saleem, M. Azam, Z. Mubeen, and G. Mumtaz, "Machine learning for improved threat detection: LightGBM vs. CatBoost," *Journal of Computing & Biomedical Informatics*, vol. 7, no. 01, pp. 571–580, 2024.