# DoS Attack Detection in 5G Core Network using Machine Learning

Abdel Fadyl Chabi, João Vitor da Silva Campos,
Vivianne de Aquino Rodrigues and Matheus Fontinele de Aguiar
*Sidia Institute of Science and Technology*
Manaus-AM, Brazil

***Abstract*— Contrary to previous mobile networks, 5G architecture is service based and utilize Network Functions (NFs) to manage connectivity, security, data management, and other functionalities of the network. This kind of architecture allow 5G networks to be more scalable, flexible and eficient than the legacy networks, however, service based networks also introduces a series of vulnerabilities. As 5G networks evolve, security becomes as much of a concern as network performance. 5G networks allow the introduction of machine learning techniques, which can improve network performance, as well as ensure security through the implementation of machine learning models on both base station and User Equipment (UE) side. In this paper we use five machine learning algorithm, namely Random Forest, XGboost, Decision Tree, Multi-Layer Perceptron (MLP) and LightGBM to detect Denial of Service (DoS) attack in 5G core network. Friedmann and Nemenyi tests have been performed to compare the models performance and elect the best model among the five implemented. Random Forest, XGBoost and Decision Tree are elected the best model for Dos attack detection problem, when comparing the metrics and statistical results obtained.**

***Keywords*— 5G networks, Security, DoS attack, Machine learning.**

## I. Introduction

Ensuring security in 5G networks is essential in current scenario of massive connectivity and virtualized infrastructure. When compared to legacy mobile networks, 5G systems are more vulnerable to Denial of Service (DoS) attacks due to their architecture and the increased number of connected devices.

DoS attacks are difficult to mitigate and they can disrupt network functionality. During a DoS attack, the malicious user will tries to overwhelm the target network or service by sending a large amount of traffic [1]. In light of this, DoS attacks are considered one of the most damaging, because it prevents users from accessing network services. This can be especially concerning for 5G network deployments as several applications are expected to rely on 5G infrastructure [2].

In [3], the authors presented a model that exposes the vulnerability which enables a DoS attack during a network slicing, detecting an attack and further localizing. To demonstrate the effectiveness of proposed detection and localization model, experiments were conducted to indicate that the model can identify and restore normality once an attack has been initiated.

Aiming to explore concepts of deep learning (DL), [4] describes a 5G-V2X testbed that includes robot cars and has the ability of creating network slices on demand. The testbed includes a mobility server that acknowledges every mobility decision taken by the robot cars. In DoS attack scenarios with the mobility server targeted, such acknowledgments are not received, the robot cars cannot move correctly and may cause an accident. All DL models rely on convolutional neural networks (CNNs), supported by TensorFlow, Keras and OpenCV tools.

DoS attack can produce huge damage when Access and Mobility Management Function (AMF) is affected on 5G Core network. In [5], the authors addressed a specific type of DoS attack that can be initiated via both the data plane and control plane. In all scenarios showed, the AMF is the initial target, which subsequently impacts other Network Functions, leading to severe consequences for the 5GC.

Diverging from previous works, our proposal is to alsocreate a model for detecting and predicting DoS attacks in AMF on the 5G core, however, using both machine learning (ML) models and statistical analysis. ML algorithms are key solutions to optimize and secure 5G networks. The aim is to protect the 5G core from DoS attacks using different ML algorithms and compare the results, as well as performing statistics tests. The algorithms used were: Random Forest, XGBoost, Decision Tree, MLP and LightGBM. Also, the statistics tests Friedmann and Nemenyi were performed to guarantee the robustness of those solutions.

The paper is organized as follows: In Section II, the background about 5G networks, core and architecture, DoS attacks on 5G network and the machine leraning models used are presented. The experimental methodology used in model construction is outlined in Section III. The data collected in experiments and results obtained are presented in Section IV. Finally, the conclusion is presented.

## II. Background

### A. 5G NR Core Architecture

The global deployment of the 5G networks has led to support for increased connection density, with more functionalities, services, and use cases when compared to legacy networks. The central network component in 5G technology is the

Abdel Fadyl Chabi, Sidia Institute of Science and Technology, Manaus - Brazil, e-mail: abdel.chabi@sidia.com; João Vitor da Silva Campos, Sidia Institute of Science and Technology, Manaus - Brazil, e-mail: joao.vitor@sidia.com; Vivianne de Aquino Rodrigues, Sidia Institute of Science and Technology, Manaus - Brazil, e-mail: vivianne.aquino@sidia.com; Matheus Fontinele de Aguiar, Sidia Institute of Science and Technology, Manaus - Brazil, e-mail: matheus.fontinele@sidia.com.

5G Core (5GC) and it acts as the backbone for the connection of devices, applications, and services. 5GC operates with a Service-Based Architecture (SBA), which consists of Network Functions (NFs) and Service-Based Interfaces (SBIs). [6]

The connection between Radio Access Network (RAN) and the UE is established by AMF. In addition, the Session Management Function (SMF) manages mobile access on the 5G network through sessions. Policy Control Function (PCF) provides service rules for network functions, which include QoS parameters, and network access. Unified Data Management (UDM) manages registers and authorizes access to the network, as well as user profiles, signatures, and authentication. [7],[8]

### B. DoS Attacks in 5G Network

A DoS attack occurs on 5G Core and refers to any deliberate attempt to interrupt the normal functioning of a whole system or the system components by overwhelming them with excessive traffic, invalid requests, or by exploiting protocol vulnerabilities, making the systems and its services unavailable to legitimate users. Nevertheless, based on several researches, there is little or no research focused on DoS attacks targeting the core network[7]. From 5G perspective, DoS attacks can compromise the availability of 5G network services and hinder the ability of telecommunication service providers (TSPs) to deliver promised service level agreements (SLAs), which can lead to potential losses for both TSPs and their clients [9]. DoS is on of the most critical attack since it prevents users from accessing network services [10]. The Figure 1 show an example of DoS attack to the AMF function.

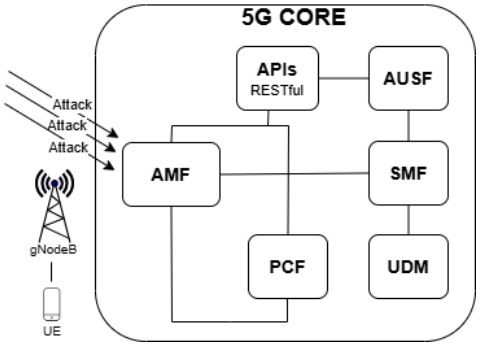


Fig. 1. DoS attack in 5G Core.

### C. ML Algorithms

*1) Random Forest:* The random forest algorithm uses bootstrap resampling as a basis for defining the decision model of sample sets. The algorithm separates the data into several sample sets randomly and from these new sets creates classification decision trees, that is, each tree is trained with a different data set combined with a random subset of attributes, and each tree is trained, independent and produce individual predictions. The final result is chosen from a majority vote, thus using the most common result obtained in each tree. [11]

*2) XGBoost:* The XGBoost is a supervised model that uses an algorithm implementation with emphasis on efficiency and scalability by optimizing gradient reinforcement, thus being more accurate and robust, for which a combination of predictive models is usually used as a decision tree. [12]

*3) Decision Tree:* Decision Tree is an algorithm for classification and regression of supervised learning, has a tree structure and can be interpreted has a tree structure and can be interpreted as a set of rules if-then. The rating of this model is its fast classification speed and easy interpretability of the results due to its hierarchical structure. [13]

*4) MLP:* The MLP is a key model for classification and regression problems. This model used interconnected neurons in different layers divided as input layer, hidden layer and output layer using a nonlinear activation function. The model is trained using the backpropagation algorithm where during training the model learns to adjust each of the weights associated with the input neurons. [14]

*5) LightGBM:* LightGBM uses decision tree-based learning algorithms to perform gradient boosting. It iteratively builds decision trees to minimize the loss function by adjusting the parameters of each tree, thereby optimizing the model's performance. LightGBM employs advanced algorithms that aid in the search for the gradient value, such as Gradient Boosting Decision Trees (GBDT) and Gradient-based One-Side Sampling. [15]

As in [16], this work aims to compare the performance of different ML algorithms.

## III. Methodology

### A. Data Pre-Processing

The data used came from the dataset for anomaly detection of 5G NF interactions [17], that proposes a specific method based on deep learning to detect anomalies in NF interactions within the 5GC network. In this work, we conduct a comparative study using statistical methods to evaluate the performance of traditional machine learning algorithms for anomaly detection. Therefore, the two studies have different methodological approaches, due to that a direct comparative analysis of the results will not be performed. The data includes abnormal and normal interactions between 5G Network Functions. Three types of abnormal interaction have been collected: Evil NF deletion, NF DoS and UE info extraction. These three types of attacks lead to a network service failure at some level.

In this work we will focus on DoS attack, thereupon only DoS attack data will be used. Abnormal and normal traces about NF invocations are organized through JSON files. In order to posteriorly train our model the first data processing step is convert the JSON files in dataframe. Normal and abnormal JSON files have been converted to dataframe and the dataframes have been concatenated.

Since we will employ a supervised learning, we labeled abnormal interactions as ***"1"*** and normal interactions as ***"0"*** in the dataframe created to be used as target to predict. The dataset have 12 features that are directly related to the nodes of the network and interactions between

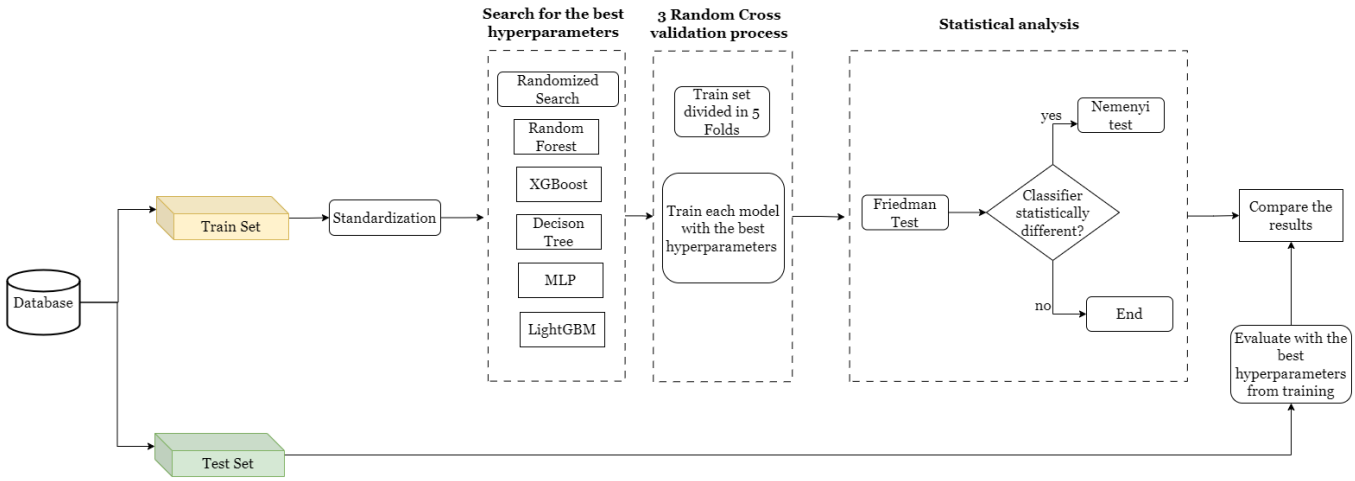


Fig. 2. Training and validation flow.

the NF, for the training,9 features were chosen, composed with quantitative and qualitative data. One of the most important feature is the ***operation_name***, namely the RESTful APIs used for the interactions. These APIs are available in URI format like (***nnrf-disc/v1/nf-instances?requester-nf-type=AMF&target-nf-type=AUSF***). Before training our model, we encoded the data of the feature ***operation_name*** using ***LabelEncoder*** function from ***sklearn*** library. The Table I, present the mapping of some URI involved in NF interaction.

TABLE I
URI TEMPLATES.

| Method | URI | Meaning |
|---|---|---|
| Get | apiRoot/namf-oam/v1/registered-ue-context | Register UE to web |
| Delete | apiRoot/nnrf-nfm/v1/nf-instances/nfInstanceID | Deregisters a given NF Instance |
| Put | apiRoot/nnrf-nfm/v1/nf-instances/nfInstanceID | Register a new NF Instance |

In addition to ***operation_name***, other features were selected, like: ***http_method***, which represents the method mentioned in table I; ***caller***, which is the function in charge of originating the network interaction; ***callee***: the function receiveing the interaction iniciated, have been encoded.

## B. Training

The Figure 2 shows how training and validation processes are structured. For training phase, firstly the dataset was divided and then the cross-validation was applied, as below:

*1) **Dataset Division**:* After the encoding process and features selection, the dataset have been divided in train set (75% of data) and test set (25%). The test set is saved and will be used at the final step as new data to validate the models generalization. The train set is first submitted to a standardization process. Following the standardization process, the training is initiated, searching for the best hyperparameters for Random Forest, XGBoost, Decision Tree, MLP and LightGBM respectively. For this purpose, Randomized Search have been used for each model.

*2) **Croos-Validation**:* The motivation behind using croos-validation is to reduce the risk of overfitting or underfitting, making the performance evaluation more realistic. To guarantee a better robustness, we used stratified cross validation. During stratified cross validation, the training set is divided randomly in 5 subset (folds) and each fold represent faithfully the whole set. The hyper-parameters from Randomized Search are tested and validated during the croos-validation and the process is repeated three times. The best hyper-parameters are saved and used to evaluate each model during the training process. The models were evaluated using accuracy, precision and F-measure metrics.

## C. Statistical Analysis

Once the models have been evaluated, selecting the one with the best performance just comparing the metrics can be inefficient, specifically when it comes to a sensitive problem like 5G network security. In this regard, a statistical validation was performed to compare the results. Statistical evaluation of experimental results has been considered an essential part of validation of new machine learning methods for quite some time [18]. The first step consists in performing Friedman test.

*1) **Friedman Test**:* Friedman test is a global statistical test used to compare at least 3 classifiers on multiple datasets. For each data set, the test assigns a ranking to the classifier. The best classifier receives rank 1, the second best, rank 2, and so on. In cases of a tie, the average of the tied rankings is used. Two hypothesis $H_0$ and $H_1$ are used as assumptions to identify if there are significant differences between the classifiers.

- $H_0$: There is no differences between the classifiers;
- $H_1$: The classifiers are different.

The output of Friedman test provide the ***p_value***, which is described in equation 1 and Friedman statistic value ***Q***. If $p\_value < 0.05$, $H_0$ is rejected and another test, namely Nemenyi test need to be performed. On the other hand, if

*p_value* $\geq 0.05$, $H_0$ is true and as conclusion there is no differences between the classifiers.

$$p_{value} = 1 - F_{Q^2_{(k-1)}}(Q^2) \tag{1}$$

where $Q$ is Friedman statistic value and $F_{Q^2_{(k-1)}}$ is the *Cumulative Distribution Function* of Q with $k-1$ degrees of freedom.

*2) **Nemenyi Test**:* The next step after detecting differences between the classifiers through Friedman test is to perform Nemenyi test to analyse with more details the differences found. This analysis is conducted comparing the average rank of each classifier. The average rank of each classifier is computed and if the difference between two average ranks is greater than the Critical Difference (CD) we conclude that the classifier are statistically different as pointed out by Friedman test. Otherwise the classifiers are equal and the best performance corresponds to the model with the lowest average rank.

$CD$ is computed using the equation 2:

$$CD = q_\alpha \sqrt{\frac{k(k+1)}{6n}} \tag{2}$$

where $k$ is the number of classifiers, $n$ is the number of metrics and $q_\alpha$ is a parametric value obtained according to the number of classifiers and a certain significance level. For 5 classifiers and $\alpha = 0.05$, $q_\alpha = 2.728$.

After the statistical analysis, the test set that was separated before the training process is evaluated under accuracy, precision and F-measure metric and the performance for all models were compared.

## IV. Results

During cross-validation, each folds were evaluated separately under accuracy, precision and f-measure metrics. The metric values are presented as a single point estimate which is the mean value along with the confidence interval (CI). CI is computed for each metric for the respective model using the equation 3:

$$CI = \bar{x} \pm z\frac{s}{\sqrt{n}} \tag{3}$$

where $\bar{x}$ is the mean value, $z$ is the confidence level value, $s$ is the standard deviation and $n$ is the sample size. The confidence level used is $95\%$ which correspond to $z = 1.96$.

The tables II, III and IV bring out the mean value and CI for each models for accuracy, precision and F-Measure respectively.

TABLE II
Train Accuracy.

| Model | Mean | CI |
|---|---|---|
| Random Forest | 0.7980 | [0.7951, 0.8008] |
| XGBoost | 0.8018 | [0.7985, 0.8051] |
| Decision Tree | 0.7819 | [0.7785, 0.7853] |
| MLP | 0.7086 | [0.7027, 0.7146] |
| LightGBM | 0.8154 | [0.8121, 0.8186] |

TABLE III
Train Precision.

| Model | Mean | CI |
|---|---|---|
| Random Forest | 0.7994 | [0.7969, 0.8020] |
| XGBoost | 0.8041 | [0.8012, 0.8070] |
| Decision Tree | 0.7823 | [0.7790, 0.7857] |
| MLP | 0.7194 | [0.7095, 0.7293] |
| LightGBM | 0.8167 | [0.8136, 0.8197] |

TABLE IV
Train F-Measure.

| Model | Mean | CI |
|---|---|---|
| Random Forest | 0.7977 | [0.7948, 0.8006] |
| XGBoost | 0.8013 | [0.7980, 0.8047] |
| Decision Tree | 0.7818 | [0.7784, 0.7852] |
| MLP | 0.7051 | [0.6999, 0.7103] |
| LightGBM | 0.8151 | [0.8118, 0.8184] |

After Friedman test, hypothesis $H_1$ have been confirmed showing the need of performing Nemenyi test. The Figure 3 present the result of Nemenyi test for accuracy.

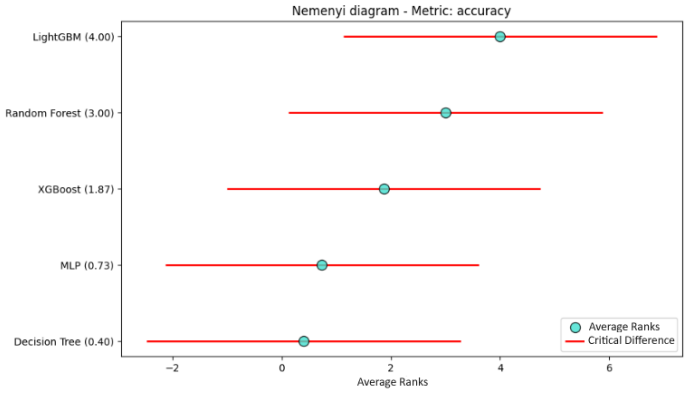


Fig. 3. Nemenyi Diagram

In the diagram of Figure 3, the red line represent the CD and the blue points the average ranks. If the distance between the average ranks of two models are bigger than CD, these models are statistically different, otherwise the models are not different and the model with the lower average ranks is the one with the better performance. Derived from this we can observe that there are three statistical group according to CD overlapping. The CD of MLP and Decision tree are superposed, thereupon the difference between these models is not statistically significant and Decision tree have the better performance since its average rank is lower. LightGBM and Random Forest's CD coincide and does not overlap with the others, that is, both models are statistically equivalent and different from the others. XGBoost's CD don't overlap neither with MLP and Decision tree's CD nor with LightGBM and Random Forest's, hence XGBoost is also statistically different from all the other models.

Following the statistical tests, the evaluation of the test set

have been performed. The table V, presents accuracy, precision and F-Measure values respectively for each model.

TABLE V
TEST METRICS.

| Model | Accuracy | Precision | F-Measure |
|---|---|---|---|
| Random Forest | 0.7937 | 0.7955 | 0.7934 |
| XGBoost | 0.7973 | 0.7995 | 0.7969 |
| Decision Tree | 0.7795 | 0.7800 | 0.7793 |
| MLP | 0.6866 | 0.6867 | 0.6865 |
| LightGBM | 0.8125 | 0.8134 | 0.8123 |

It apparent that all the models presented a good performance on test set, which means that there is neither overfitting nor underfitting. The test metric values are close to train value for all the model trained. For instance the accuracy on training set for Random Forest is 0.7980 while is 0.7977 on test set. Additionally we could classify the models in three statistical group perfoming Nemenyi test. LighGBM and Random Forest form the first group and Random Forest is better due to lower average ranks criteria, XGBoost form the second group and both MLP and Decision Tree form the third group with Decision Tree having the best performance. Based on this analysis Random Forest, XGBoost and Decision Tree are elected the best model for DoS attack detection problem.

## V. CONCLUSIONS

This paper assesses security aspects in 5G networks involve DoS attacks, and through the study carried out we can highlight the main strengths achieved:

- Using Machine Learning to predict attacks on the 5G network is very effective;
- The importance of statistical evaluation based on the Friedman and Nemenyi test for defining ML models in the context of network attacks;
- This study has a significant degree of contribution to the scientific community, as a way to bring new directions about security in 5G, and indicate possible solutions to mitigate attacks on core network.

By the use of Friedman and Nemenyi statistical tests, based on the accuracy metric, it was possible to observe 3 statistical groups according to CD overlap. According to both tests, the models Random Forest, XGBoost and Decision Tree presented the best performance on DoS attack classification, when comparing the metrics and statistical results obtained. The accuracy of test set is 0.7937, 0.7973 and 0.7795 respectively for these models and they are the more consistent according to statistical tests performed.

This work emphasized the importance of implementing machine learning techniques for DoS attacks by evaluating the results using statistical tests. For future work, it is possible to use unsupervised models so the model itself decides which parameters in the signaling are responsible for defining a normal or abnormal pattern. Another alternative is using deep learning models to analyze attacks in other network functions with a more complex structure and build models that can make predictions with encrypted data.

## ACKNOWLEDGMENT

This paper is a result of the Research, Development & Innovation Project (AMAN) performed at Sidia Institute of Science and Technology sponsored by Samsung Eletrônica da Amazônia Ltda., using resources under terms of Federal Law No. 8.387/1991, by having its disclosure and publicity in accordance with art. 39 of Decree No. 10.521/2020.

## REFERENCES

[1] D. Sattar and A. Matrawy, "Towards secure slicing: Using slice isolation to mitigate ddos attacks on 5g core network slices," in *2019 IEEE Conference on Communications and Network Security (CNS)*, 2019, pp. 82–90.

[2] A. Pagadala and G. Ahmed, "Analysis of ddos attacks in 5g networks," in *2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, 2023, pp. 1–6.

[3] H. Bisht, M. Patra, and S. Kumar, "Detection and localization of ddos attack during inter-slice handover in 5g network slicing," in *2023 IEEE 20th Consumer Communications Networking Conference (CCNC)*, 2023, pp. 798–803.

[4] B. Bousalem, V. F. Silva, S.-B. Bakhouche, R. Langar, and S. Cherrier, "Detecting and mitigating ddos attacks in 5g-v2x networks: A deep learning-based approach," in *2025 Global Information Infrastructure and Networking Symposium (GIIS)*, 2025, pp. 1–2.

[5] M. R. Dey, P. Nithiyasri, and M. Patra, "Early detection of dos attacks in 5g core networks," in *2024 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS)*, 2024, pp. 1–6.

[6] M. I. T. Training, "5G System Architecture," Mpirical, Reference Document, 05 2020, fist puplished by Mpirical Limited in 2020.

[7] S. Park, B. Cho, D. Kim, and I. You, "Machine learning based signaling ddos detection system for 5g stand alone core network," *Applied Sciences*, vol. 12, no. 23, p. 12456, 2022.

[8] S. Jing and H. Wang, "Design and implementation of a 5g network architecture based on software defined network," in *2023 IEEE 3rd International Conference on Electronic Technology, Communication and Information (ICETCI)*, 2023, pp. 1444–1449.

[9] A.-A. Maiga, E. Ataro, and S. Githinji, "Xgboost and deep learning based-federated learning for ddos attack detection in 5g core network vnfs," in *2024 6th International Conference on Computer Communication and the Internet (ICCCI)*, 2024, pp. 128–133.

[10] A. Pagadala and G. Ahmed, "Analysis of ddos attacks in 5g networks," in *2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, 2023, pp. 1–6.

[11] L. Wei, "Genetic algorithm optimization of concrete frame structure based on improved random forest," in *2023 International Conference on Electronics and Devices, Computational Science (ICEDCS)*, 2023, pp. 249–253.

[12] J. Deng, L. Cheng, H. Yuan, K. Zheng, X. Li, and Q. Li, "An online detection system for ldos attack based on xgboost," in *2023 IEEE Intl Conf on Parallel Distributed Processing with Applications, Big Data Cloud Computing, Sustainable Computing Communications, Social Computing Networking (ISPA/BDCloud/SocialCom/SustainCom)*, 2023, pp. 1083–1088.

[13] K. Liang and L. Zhang, "Application of decision tree random forest algorithm in the evolution of dual innovation in smes," in *2025 IEEE International Conference on Electronics, Energy Systems and Power Engineering (EESPE)*, 2025, pp. 895–899.

[14] J. Naskath, G. Sivakamasundari, and A. A. S. Begum, "A study on different deep learning algorithms used in deep neural nets: Mlp som and dbn," *Wireless personal communications*, vol. 128, no. 4, pp. 2913–2936, 2023.

[15] R. Liu, "Research on lightgbm algorithm for subway gate fault detection," in *2024 International Conference on Control, Electronic Engineering and Machine Learning (CEEML)*, 2024, pp. 116–120.

[16] Y.-E. Kim, Y.-S. Kim, and H. Kim, "Effective feature selection methods to detect iot ddos attack in 5g core network," *Sensors*, vol. 22, no. 10, p. 3819, 2022.

[17] (2025) Dataset-for-anomaly-detection-of-5g-nf-interactions. [Online]. Available: https://github.com/tywofxd/Dataset-for-anomaly-detection-of-5G-NF-interactions

[18] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *Journal of Machine Learning Research*, vol. 7, pp. 1–30, 2006.