Jurnal Riset Sistem Informasi Dan Teknik Informatika (JURASIK)
Volume 10, Nomor 1, Februari 2025, pp 410-421
Terakreditasi Nomor 204/E/KPT/2022 | ISSN: 2527-5771/EISSN: 2549-7839
https://tunasbangsa.ac.id/ejurnal/index.php/jurasik

Enhancing UAV Communication Security: Multi-Label Anomaly Detection Using Machine Learning in Imbalanced Data Environments

Gregorius Airlangga[1*] , Denny Jean Cross Sihombing[2], Oskar Ika Adi Nugroho[3]
[1,2]Information Systems Study Program, Universitas Katolik Indonesia Atma Jaya, Indonesia
[3]Electrical Engineering Department, National Chung Cheng University, Chiayi, Taiwan
Corresponding Author: gregorius.airlangga@atmajaya.ac.id

Abstract

Unmanned Aerial Vehicle (UAV) communication networks are increasingly vulnerable to cyber threats, including spoofing, jamming, malware, and distributed denial-of-service (DDoS) attacks. Effective anomaly detection is crucial to maintaining network integrity and operational security. This study evaluates multiple machines learning models, including Support Vector Machines, Logistic Regression, XGBoost, Gradient Boosting, and Random Forest, to detect anomalies in UAV communication networks. A real-world dataset containing 44,016 instances of network telemetry and security indicators was utilized, with each instance labeled for multiple potential anomalies. Experimental results reveal a significant class imbalance, where models achieve high accuracy (92%) but fail to detect minority class anomalies, yielding near-zero recall scores for critical cyber threats. The study highlights the limitations of traditional classifiers in imbalanced multi-label classification tasks and emphasizes the need for advanced techniques such as Synthetic Minority Over-sampling (SMOTE), cost-sensitive learning, and deep learning-based anomaly detection. The findings suggest that conventional machine learning approaches alone are insufficient for reliable anomaly detection in UAV networks, necessitating hybrid solutions that integrate multiple detection paradigms. Future work should explore adaptive ensemble learning methods and deep anomaly detection frameworks to improve recall and precision for rare cybersecurity threats.

Keywords: *UAV, Communication Security, Anomaly Detection, Machine Learning*

1. INTRODUCTION

Unmanned Aerial Vehicles (UAVs), commonly known as drones, have experienced significant advancements over the past decade, leading to widespread adoption across various sectors, including surveillance, logistics, disaster response, and military operations [1]–[3]. These applications rely on robust and secure communication networks to ensure real-time data transmission, situational awareness, and operational efficiency [4]–[6]. However, the increasing deployment of UAVs has also exposed them to a multitude of cyber threats, such as distributed denial-of-service (DDoS) attacks, spoofing, jamming, malware intrusions, and man-in-the-middle (MITM) attacks [7]. The integration of UAVs into modern communication infrastructures demands innovative solutions for detecting and mitigating network anomalies to enhance reliability, security, and resilience in UAV-based systems [1]. The UAV communication environment is characterized by unique challenges, including dynamic topologies, varying signal strengths, frequency interferences, and heterogeneous network architectures [8]–[10]. Unlike conventional terrestrial networks, UAV networks operate in highly unpredictable conditions where rapid mobility, constrained bandwidth, and fluctuating environmental factors significantly impact communication quality[11]–

[13]. Moreover, cyber threats targeting drone communication systems are becoming more sophisticated, exploiting vulnerabilities in encryption protocols, transmission power levels, and network traffic patterns [14]–[16]. These factors necessitate advanced anomaly detection mechanisms capable of identifying malicious activities while minimizing false positives in high-dimensional and multi-label datasets [17].

Machine learning (ML) and artificial intelligence (AI) techniques have emerged as powerful tools for network anomaly detection, offering adaptive, data-driven approaches to identifying suspicious behaviors in UAV communication networks [18]. Traditional rule-based intrusion detection systems struggle to cope with evolving attack vectors and complex patterns in network traffic data [19]. In contrast, ML models can analyze vast amounts of real-time telemetry data, extract meaningful features, and classify network anomalies with high precision .

This study presents an in-depth investigation of multiple ML models, including Random Forest (RF), Gradient Boosting (GB), XGBoost (XGB), Logistic Regression (LR), and Support Vector Machines (SVM), for multi-label classification of UAV network anomalies. The research leverages a high-dimensional dataset consisting of 26 input features and 8 multi-label target variables, capturing critical aspects such as signal strength, packet loss rate, round trip time, data rate, network traffic volume, encryption protocols, and behavioral patterns. The proposed methodology involves feature selection, label encoding, normalization, and manual undersampling to address class imbalance, followed by model training and performance evaluation using precision, recall, and F1-score metrics. A distinguishing aspect of this research is its focus on multi-output classification, where each data instance may exhibit multiple concurrent network anomalies. Unlike conventional binary or multi-class classification, multi-label classification presents unique computational and interpretative challenges, requiring specialized techniques such as MultiOutputClassifier wrappers and hierarchical classification strategies. This study systematically compares the effectiveness of different ML models, highlighting their strengths and limitations in real-world UAV communication security scenarios.

The key contributions of this research include a comprehensive feature engineering process that identifies and preprocesses critical communication parameters influencing network anomalies in UAV systems. Additionally, the study develops a machine learning pipeline tailored for detecting concurrent network threats using a large-scale, real-world dataset. Through comparative model performance analysis, the research evaluates state-of-the-art classification models to determine the most effective approach for UAV network anomaly detection. The implementation of manual undersampling addresses class imbalance, enhancing classification robustness. Furthermore, the study provides insights into the practical deployment of ML-based intrusion detection systems to enhance UAV communication security. The remainder of this paper is structured as follows: Section 2 details the proposed methodology, including dataset description, feature selection, preprocessing, and model implementation. Section 3 presents experimental results and comparative performance analysis. In addition, we also

discuss the implications of the findings and their relevance to UAV security. Finally, Section 4 concludes the paper and outlines potential directions for future research.

2. RESEARCH METHODOLOGY

2.1. Dataset Description

In this study, we utilize the Drone Communication and Network Anomaly Detection Dataset, a comprehensive collection of data designed to facilitate research in UAV communication networks and anomaly detection. The dataset can be downloaded from [20].This dataset encompasses multi-dimensional data collected from a simulated drone communication network over a period from November 1, 2019, to December 31, 2024, providing a rich temporal context for analysis. The dataset comprises a total of 44,016 samples, each corresponding to an hourly timestamp within the specified timeframe. Each sample is represented as a tuple (x_i, y_i), where $x_i \in R^{26}$ denotes the input feature vector, and $y_i \in \{0,1\}^8$ represents the multi-label target vector. The input feature vector x_i consists of 26 attributes that capture various aspects of drone communication and operation. These features include communication parameters such as signal strength, packet loss rate, round trip time, frequency band, data rate, and transmission power. GPS data, including latitude, longitude, and altitude, provide spatial positioning details of the drone. Network statistics such as network traffic volume, uplink-downlink quality, and base station load reflect the state and performance of the network infrastructure. Security indicators, including port scanning attempts, malware detection signals, anomaly in behavioral patterns, and intrusion detection flags, serve as potential indicators of security threats.

The target vector y_i is a binary vector of length 8, where each element corresponds to a specific type of network anomaly. The presence of an anomaly is indicated by a value of 1, while its absence is denoted by 0. The dataset supports multi-label classification, meaning each instance may exhibit multiple concurrent anomalies such as spoofing, man-in-the-middle attacks, DDoS, GPS spoofing, malware infections, jamming, and protocol exploits. To understand the dataset's characteristics, label cardinality and label density are computed. Label cardinality (LC) represents the average number of labels assigned to each instance, defined as (1).

$$LC = \frac{1}{N}\sum_{i=1}^{N}|y_i| \tag{1}$$

where N is the total number of instances, and $|y_i|$ denotes the number of active labels for the i -th instance. Label density (LD) normalizes label cardinality by the total number of possible labels L , given by (2).

$$LD = \frac{1}{N}\sum_{i=1}^{N}\frac{|y_i|}{L} \tag{2}$$

where $L = 8$ in this dataset. These metrics provide insights into the multi-label nature of the dataset, indicating how many anomalies typically co-occur and the proportion of labels active per instance. Understanding the distribution of anomalies is crucial for developing effective detection models. Let n_j denote the number of instances where the j-th label is active. The label frequency for the j-th anomaly is computed as (3).

$$\text{Label Frequency}_j = \frac{n_j}{N} \tag{3}$$

Analyzing these frequencies helps identify class imbalances, which can significantly impact the performance of machine learning models. If certain anomalies are underrepresented, specialized techniques such as resampling or cost-sensitive learning may be employed to address the imbalance.

2.2. Feature Selection

Feature selection is a crucial step in machine learning pipelines, significantly affecting model performance, interpretability, and computational efficiency. In this study, a systematic approach is employed to identify and retain the most informative features from the dataset, ensuring optimal anomaly detection in UAV communication networks. The selection process integrates domain knowledge, statistical correlation analysis, feature importance ranking, and recursive elimination techniques to refine the feature set effectively.

Given the nature of UAV communication, certain features inherently carry more significance in detecting anomalies. Attributes related to signal strength, packet loss rate, round-trip time, transmission power, data rate, frequency band, and network traffic volume directly influence the quality of communication links. Features associated with encryption type, sequence number gaps, anomaly detection signals, and malware indicators provide valuable information regarding network security threats. Based on expert insights, these features are prioritized to ensure they encapsulate vital aspects of UAV network behavior. To further refine the feature space, correlation analysis is performed between each feature and the multi-label target variables. Pearson correlation is used for continuous variables, while point-biserial correlation is employed for categorical features. The Pearson correlation coefficient between a given feature and a target anomaly is computed as (4).

$$\rho_{X_i,Y_j} = \frac{\sum_{k=1}^{N}\left(X_{i,k} - \bar{X}_i\right)\left(Y_{j,k} - \bar{Y}_j\right)}{\sqrt{\sum_{k=1}^{N}\left(X_{i,k} - \bar{X}_i\right)^2}\sqrt{\sum_{k=1}^{N}\left(Y_{j,k} - \bar{Y}_j\right)^2}} \tag{4}$$

where $(X_{i,k})$ represents the value of feature (X_i) for sample (k), $(Y_{j,k})$ is the binary label for anomaly (j), $(\bar{X}_i)$ and $(\bar{Y}_j)$ denote the mean values of the respective feature and target variable, and (N) is the total number of samples. Features with correlation magnitudes exceeding a threshold are retained for further consideration, while highly collinear features are examined, and redundant ones are removed. To ensure that the most relevant features are selected, tree-

based models such as Random Forest and Gradient Boosting are employed to compute feature importance scores. These models provide an importance metric that quantifies the contribution of each feature to the model's predictive capabilities. In the case of Random Forest, feature importance is determined using the Gini importance, which is given by (5).

$$I(X_i) = \frac{1}{T}\sum_{t=1}^{T}\sum_{n\in\mathcal{N}_t}\Delta G_n \quad (5)$$

where (T) represents the number of trees in the ensemble, ($\mathcal{N}_t$) is the set of nodes in tree (t), and (ΔG_n) denotes the reduction in Gini impurity at node (n). The Gini impurity for a given node, defined as (6).

$$G = 1 - \sum_{j=1}^{C} p_j^2 \quad (6)$$

where (C) is the number of classes and (p_j) is the proportion of samples belonging to class (j), provides a measure of node purity. In Gradient Boosting models, feature importance is determined based on the total reduction in the loss function when a feature is used as a splitting criterion. To optimize the selection process further, Recursive Feature Elimination (RFE) is applied. RFE iteratively removes the least important features based on model performance, ensuring that only the most significant predictors are retained. The algorithm begins by training a model using all available features and computing their respective importance scores. The feature with the lowest importance score, denoted as (7).

$$i^* = \arg\min_i I\,(X_i) \quad (7)$$

is removed, and the model is retrained with the remaining features. This process continues until the desired number of features is achieved. The optimal feature subset is determined through cross-validation, evaluating model performance across multiple iterations. After applying these selection techniques, the final feature set is composed of attributes that capture critical aspects of communication quality, network traffic, security indicators, and behavioral patterns. These selected features ensure a balance between interpretability and predictive power, improving anomaly detection while reducing computational complexity.

2.3. Data Preprocessing

Data preprocessing is a crucial step in machine learning that ensures data quality, consistency, and suitability for training predictive models. Properly processed data enhances model generalization and robustness while mitigating the impact of noise, missing values, and imbalanced distributions. In this study, multiple preprocessing techniques are applied, including handling missing values, encoding categorical variables, normalizing numerical features, and addressing class imbalances. Handling missing values is necessary to ensure that no gaps in

Jurnal Riset Sistem Informasi Dan Teknik Informatika (JURASIK)
Volume 10, Nomor 1, Februari 2025, pp 410-421
Terakreditasi Nomor 204/E/KPT/2022 | ISSN: 2527-5771/EISSN: 2549-7839
https://tunasbangsa.ac.id/ejurnal/index.php/jurasik

the dataset degrade model performance. Given a feature matrix ($X \in R^{N \times d}$), where (N) is the number of samples and (d) is the number of features, missing values are imputed using appropriate statistical measures. If ($x_{i,j}$) represents the value of feature (j) in sample (i), missing values for continuous features are replaced by the mean of that feature across all available samples, computed as (8).

$$\widehat{x_{.,J}} = \frac{1}{N_j} \sum_{i \in S_j} x_{i,j} \quad (8)$$

where (S_j) denotes the set of samples with non-missing values in feature (j), and (N_j) represents the number of such samples. For categorical variables, missing values are imputed with the most frequent category in that feature, ensuring minimal bias in class distributions. Categorical features, such as communication protocols and encryption types, require transformation into numerical representations for compatibility with machine learning models. Label encoding is applied to categorical variables, mapping each unique category (C_k) in a feature (X_j) to an integer value ($L(C_k)$). Formally, the transformation function is defined as (9).

$$\mathrm{L(C_k)} = \mathrm{k}, \quad \text{for } \mathrm{k} \in \{0, 1, \ldots, \mathrm{K_j} - 1\} \quad (9)$$

where (K_j) is the number of unique categories in feature (X_j). This encoding method preserves ordinal relationships while allowing models to interpret categorical inputs as discrete numerical values. Feature scaling is applied to ensure that numerical attributes contribute equally to model training and do not introduce biases due to differing magnitudes. Standardization transforms a feature (X_j) into a distribution with zero mean and unit variance using the transformation as presented in (10).

$$X_j' = \frac{X_j - \mu_j}{\sigma_j} \quad (10)$$

where (μ_j) and (σ_j) denote the mean and standard deviation of feature (X_j), respectively. This transformation ensures that all features are scaled proportionally, reducing the risk of certain features dominating others in distance-based learning algorithms. Class imbalance is a significant challenge in anomaly detection, where certain types of anomalies occur far less frequently than normal instances. Let ($y_{i,k}$) represent the presence or absence of anomaly (k) in sample (i), and let the proportion of class (k) be denoted by (11).

$$P_k = \frac{1}{N} \sum_{i=1}^{N} y_{i,k} \quad (11)$$

where a small value of (P_k) indicates an underrepresented class. To address this imbalance, undersampling is employed by reducing the majority class size to match the smallest class. If ($N_{\min}$) represents the number of samples in the

minority class, a subset of size ($N_{\min}$) is randomly selected from the majority class, ensuring a balanced dataset. This approach prevents models from being biased toward frequently occurring labels while improving detection rates for rare anomalies. After applying these preprocessing steps, the dataset is structured into a refined feature matrix and target variable set, ensuring consistency and optimal representation of input attributes. These transformations enhance model performance by reducing data variability, eliminating inconsistencies, and addressing imbalances in class distributions. The preprocessed dataset is now suitable for model training and evaluation in multi-label anomaly detection tasks.

2.4. Model Implementation

Model implementation is a critical phase in machine learning, encompassing the selection of appropriate algorithms, training strategies, and performance evaluation techniques. In this study, a multi-label classification approach is employed to detect network anomalies in UAV communication systems. Given that each instance in the dataset can be associated with multiple anomalies, the learning process requires specialized methods capable of handling multi-output predictions efficiently. A fundamental aspect of the implementation involves transforming the multi-label classification problem into a set of independent binary classification tasks. Let ($X \in R^{N \times d}$) represent the feature matrix, where (N) is the total number of samples and (d) is the number of selected features. Let ($Y \in \{0,1\}^{N \times L}$) denote the corresponding label matrix, where (L) is the number of anomaly classes. Each row (y_i) of (Y) consists of binary values, where ($y_{i,j} = 1$) indicates the presence of anomaly (j) in sample (i), and ($y_{i,j} = 0$) signifies its absence. To address the multi-label nature of the dataset, the binary relevance (BR) approach is applied. In binary relevance, each label (Y_j) is treated as an independent binary classification problem, effectively training (L) separate classifiers. Given a classifier function ($h_j: R^d \rightarrow \{0,1\}$), the predicted label matrix is obtained by computing (12).

$$\hat{Y} = [h_1(X), h_2(X), \dots, h_L(X)] \quad (12)$$

where ($h_j(X)$) represents the output of the classifier trained for label (j). This transformation allows the use of traditional binary classifiers while maintaining the ability to predict multiple labels per instance. Several machine learning algorithms are utilized to model the classification task, including Random Forest (RF), Gradient Boosting (GB), XGBoost (XGB), Logistic Regression (LR), and Support Vector Machines (SVM). Each classifier is wrapped in a multi-output framework to enable simultaneous prediction of multiple anomalies.

The Random Forest classifier, an ensemble method based on decision trees, constructs multiple decision trees during training and outputs the majority vote prediction. Given a training set ((X, Y)), a set of (T) decision trees ($\{h_t\}_{t=1}^{T}$) is trained on bootstrapped samples of (X), where each tree contributes to the final prediction via majority voting as presented as (13).

Jurnal Riset Sistem Informasi Dan Teknik Informatika (JURASIK)
Volume 10, Nomor 1, Februari 2025, pp 410-421
Terakreditasi Nomor 204/E/KPT/2022 | ISSN: 2527-5771/EISSN: 2549-7839
https://tunasbangsa.ac.id/ejurnal/index.php/jurasik

$$\widehat{y_{i,J}} = \arg\max_{c \in \{0,1\}} \sum_{t=1}^{T} I(h_t(x_i) = c) \quad (13)$$

where $(I(\cdot))$ is the indicator function. The model's strength lies in its ability to capture non-linear relationships and reduce variance through averaging multiple predictions. Gradient Boosting models, including XGBoost, optimize classification by sequentially constructing decision trees that correct the errors of their predecessors. The model minimizes a differentiable loss function (L) by iteratively updating tree parameters. At each iteration (t), the model computes a residual term $(r_i^{(t)})$ for sample (i) as (14).

$$r_i^{(t)} = -\frac{\partial L\left(y_i, \widehat{y_i^{(t-1)}}\right)}{\partial \widehat{y_i^{(t-1)}}} \quad (14)$$

where $(\widehat{y_i^{(t-1)}})$ represents the model's prediction at the previous iteration. The new tree $(h^{(t)})$ is fitted to these residuals, and the final prediction is computed as the sum of previous iterations as presented as (15).

$$\widehat{y_i^{(t)}} = \widehat{y_i^{(t-1)}} + \eta h^{(t)}(x_i) \quad (15)$$

where (η) is the learning rate controlling the contribution of each tree. This iterative refinement process leads to improved generalization and robustness. Logistic Regression models the probability of a class label using the logistic function. Given a sample (x_i), the probability of anomaly presence for label (j) is expressed as (16).

$$P\left(y_{i,j} = 1 \mid x_i\right) = \frac{1}{1 + e^{-\left(w_j^{T} x_i + b_j\right)}} \quad (16)$$

where (w_j) is the coefficient vector for label (j), and (b_j) is the bias term. The model parameters are estimated by maximizing the log-likelihood function as presented as (17).

$$\mathrm{l}(W, b) = \sum_{i=1}^{N} \sum_{j=1}^{L} y_{i,j} \log P\left(y_{i,j} = 1\right) + \left(1 - y_{i,j}\right) \log\left(1 - P\left(y_{i,j} = 1\right)\right) \quad (17)$$

which ensures optimal separation between anomaly and non-anomaly classes. Support Vector Machines (SVM) classify instances by finding the optimal decision boundary that maximizes the margin between different classes. For each label (j), a hyperplane is defined as (18).

$$w_j^{T} x_i + b_j = 0 \quad (18)$$

where (w_j) and (b_j) are model parameters. The classification rule assigns sample (i) to class $(y_{i,j} = 1)$ if $w_j^{T} x_i + b_j \geq 0$ and to class $(y_{i,j} = 0)$ otherwise. The

optimal hyperplane is obtained by solving the quadratic optimization problem as (19).

$$\min_{w_j, b_j} \frac{1}{2}|w_j|^2 \tag{19}$$

subject to the constraints as presented as (20).

$$y_{i,j}\left(w_j^{\top x_i} + b_j\right) \geq 1, \quad \forall i. \tag{20}$$

Each model is trained using the preprocessed dataset, with hyperparameters tuned through cross-validation. Performance evaluation is conducted using multi-label classification metrics, including Hamming loss, precision, recall, and F1-score. The Hamming loss measures the fraction of incorrect labels relative to the total number of labels as presented as (21).

$$H_{\text{loss}} = \frac{1}{NL}\sum_{i=1}^{N}\sum_{j=1}^{L} I\left(\widehat{y_{i,j}} \neq y_{i,j}\right) \tag{21}$$

Precision, recall, and F1-score are computed for each label and averaged to assess overall model effectiveness. These metrics provide a robust framework for evaluating the ability of machine learning algorithms to detect multiple anomalies in UAV communication networks. The trained models are deployed in a multi-output classification setting, enabling real-time anomaly detection. Predictions from multiple classifiers are aggregated to construct a comprehensive anomaly detection framework, providing actionable insights for securing UAV communication channels.

3. RESULTS AND DISCUSSION

The results obtained from evaluating various machine learning models for UAV network anomaly detection highlight significant challenges in detecting minority class anomalies as presented in the table 1. Five models were assessed: Support Vector Machine (SVM), Logistic Regression, XGBoost, Gradient Boosting, and Random Forest. Performance was measured in terms of precision, recall, and F1-score for both normal communication instances and network anomalies. The overall accuracy for all models remained consistent at approximately 92%, with minor variations. However, closer analysis of the classification reports reveals a pronounced imbalance in model performance across different classes. The precision for detecting normal communication (class 0) was consistently high, reaching 0.92 across all models. Similarly, recall for this class was close to 1.00, indicating that normal instances were correctly classified with near-perfect accuracy. Conversely, the models failed to detect anomalous instances (class 1), with recall values of 0.00 for all models, except for Random Forest, which achieved a recall of 0.00 but a slightly higher precision of 0.03.

Macro-averaged precision, recall, and F1-score provide further insight into the imbalance issue. The macro-averaged precision across all models remained at

0.46, with recall fluctuating around 0.50. The F1-score, which accounts for both precision and recall, remained at 0.48, indicating suboptimal performance in detecting anomalies. The weighted average scores, which reflect the influence of class distribution, were more favorable, with F1-scores around 0.88. However, these values were primarily driven by the model's proficiency in classifying normal communication rather than detecting anomalies.

The failure of all models to correctly classify the minority class can be attributed to the severe class imbalance in the dataset, where normal communication instances vastly outnumber anomalies. The models prioritize minimizing the classification error for the majority class, leading to biased decision boundaries that favor normal instances. Consequently, recall for anomalies remains at 0.00, demonstrating that none of the models successfully identified any anomaly in the test set. This suggests that additional techniques, such as resampling methods, cost-sensitive learning, or anomaly detection-specific models, are necessary to improve classification performance for rare events. Among the evaluated models, Random Forest exhibited slightly better anomaly detection capabilities, achieving a nonzero precision for class 1. This may be due to the model's ability to leverage multiple decision trees to learn more granular decision boundaries. However, the improvement was marginal, and the recall value remained extremely low. Ensemble-based methods such as XGBoost and Gradient Boosting failed to enhance anomaly classification, as they continued to prioritize the majority class due to the imbalance in the dataset. These results highlight the need for addressing class imbalance through advanced techniques. Potential solutions include Synthetic Minority Over-sampling Technique (SMOTE), which artificially generates synthetic samples for the minority class to balance the dataset. Alternatively, cost-sensitive learning, where misclassification penalties for anomalies are increased, can force models to focus on rare instances. Another viable approach is the use of anomaly detection algorithms, such as Isolation Forests or Autoencoders, which may be more effective in identifying rare network anomalies.

Table 1. Machine Learning Performance

Method	Accuracy	Precision	Recall	F1-Score
Support Vector Machine	0.92	0.46	0.50	0.48
Logistic Regression	0.92	0.46	0.50	0.48
XGBoost	0.92	0.46	0.50	0.48
Gradient Boosting	0.92	0.46	0.50	0.48
Random Forest	0.91	0.48	0.50	0.48

4. CONCLUSION

The findings of this study highlight significant challenges in detecting network anomalies within UAV communication systems using conventional machine learning models. Despite achieving high overall accuracy, all evaluated models, including Support Vector Machine, Logistic Regression, XGBoost, Gradient Boosting, and Random Forest, exhibited poor performance in detecting minority class anomalies. The severe class imbalance in the dataset led to models that effectively classified normal communication but failed to identify anomalies, as

evidenced by the recall values of zero for the minority class across all models. The results indicate that supervised learning approaches alone are insufficient for handling such an imbalanced classification problem. The inability to detect anomalies suggests the need for alternative techniques, such as data resampling methods, cost-sensitive learning strategies, or anomaly detection-specific models. Methods such as Synthetic Minority Over-sampling Technique (SMOTE) could help balance class distributions, while cost-sensitive learning could adjust misclassification penalties to emphasize the detection of anomalies. Furthermore, unsupervised learning techniques like Isolation Forests or Autoencoders may offer a more effective solution by focusing on the detection of deviations from normal communication patterns.

Future research should explore hybrid models that integrate both supervised and unsupervised learning techniques to enhance anomaly detection capabilities. Incorporating domain-specific knowledge into feature engineering and selection may also contribute to improving classification performance. Addressing these challenges is essential for developing more reliable and robust UAV communication security systems capable of identifying and mitigating network threats in real time. By implementing advanced techniques tailored to imbalanced datasets, future studies can improve the ability of machine learning models to detect and respond to network anomalies, thereby enhancing UAV cybersecurity and operational reliability.

DAFTAR PUSTAKA

[1] A. Khan, S. Gupta, and S. K. Gupta, "Emerging UAV technology for disaster detection, mitigation, response, and preparedness," J. F. Robot., vol. 39, no. 6, pp. 905–955, 2022.

[2] S. A. H. Mohsan, N. Q. H. Othman, Y. Li, M. H. Alsharif, and M. A. Khan, "Unmanned aerial vehicles (UAVs): Practical aspects, applications, open challenges, security issues, and future trends," Intell. Serv. Robot., vol. 16, no. 1, pp. 109–137, 2023.

[3] S. A. H. Mohsan, M. A. Khan, F. Noor, I. Ullah, and M. H. Alsharif, "Towards the unmanned aerial vehicles (UAVs): A comprehensive review," Drones, vol. 6, no. 6, p. 147, 2022.

[4] A. Munir, A. Aved, and E. Blasch, "Situational awareness: techniques, challenges, and prospects," AI, vol. 3, no. 1, pp. 55–77, 2022.

[5] M. N. Nafees, N. Saxena, A. Cardenas, S. Grijalva, and P. Burnap, "Smart grid cyber-physical situational awareness of complex operational technology attacks: A review," ACM Comput. Surv., vol. 55, no. 10, pp. 1–36, 2023.

[6] L. Ge, Y. Li, Y. Li, J. Yan, and Y. Sun, "Smart distribution network situation awareness for high-quality operation and maintenance: a brief review," Energies, vol. 15, no. 3, p. 828, 2022.

[7] Z. Yu, Z. Wang, J. Yu, D. Liu, H. H. Song, and Z. Li, "Cybersecurity of unmanned aerial vehicles: A survey," IEEE Aerosp. Electron. Syst. Mag., vol. 39, no. 9, pp. 182–215, 2023.

[8] S. I. Han, "Survey on UAV deployment and trajectory in wireless communication networks: Applications and challenges," Information, vol. 13, no. 8, p. 389, 2022.

[9] A. Baltaci, E. Dinc, M. Ozger, A. Alabbasi, C. Cavdar, and D. Schupke, "A survey of wireless networks for future aerial communications (FACOM)," IEEE Commun. Surv. \& Tutorials, vol. 23, no. 4, pp. 2833–2884, 2021.
[10] M. A. K. et al., "Swarm of UAVs for Network Management in 6G: A Technical Review," IEEE Trans. Netw. Serv. Manag., vol. 20, no. 1, pp. 741–761, 2023.
[11] Y. Bai, H. Zhao, X. Zhang, Z. Chang, R. Jäntti, and K. Yang, "Towards Autonomous Multi-UAV Wireless Network: A Survey of Reinforcement Learning-Based Approaches," IEEE Commun. Surv. \& Tutorials, 2023.
[12] A. Rovira-Sugranes, A. Razi, F. Afghah, and J. Chakareski, "A review of AI-enabled routing protocols for UAV networks: Trends, challenges, and future outlook," Ad Hoc Networks, vol. 130, p. 102790, 2022.
[13] N. Nomikos, P. K. Gkonis, P. S. Bithas, and P. Trakadas, "A survey on UAV-aided maritime communications: Deployment considerations, applications, and future challenges," IEEE Open J. Commun. Soc., vol. 4, pp. 56–78, 2022.
[14] G. E. M. Abro, S. A. B. M. Zulkifli, R. J. Masood, V. S. Asirvadam, and A. Laouiti, "Comprehensive review of UAV detection, security, and communication advancements to prevent threats," Drones, vol. 6, no. 10, p. 284, 2022.
[15] R. K. Mahmood et al., "Optimizing network security with machine learning and multi-factor authentication for enhanced intrusion detection," J. Robot. Control, vol. 5, no. 5, pp. 1502–1524, 2024.
[16] H. Kurunathan, H. Huang, K. Li, W. Ni, and E. Hossain, "Machine learning-aided operations and communications of unmanned aerial vehicles: A contemporary survey," IEEE Commun. Surv. \& Tutorials, 2023.
[17] Z. Azam, M. M. Islam, and M. N. Huda, "Comparative analysis of intrusion detection systems and machine learning based model analysis through decision tree," IEEE Access, 2023.
[18] E. El-Shafeiy, M. Alsabaan, M. I. Ibrahem, and H. Elwahsh, "Real-time anomaly detection for water quality sensor monitoring based on multivariate deep learning technique," Sensors, vol. 23, no. 20, p. 8613, 2023.
[19] M. Y. Alzahrani, "Enhancing Drone Security Through Multi-Sensor Anomaly Detection and Machine Learning," SN Comput. Sci., vol. 5, no. 5, p. 651, 2024.
[20] D. Engineer, "Drone Communication and Network Anomaly Detection Dataset." 2024.