Jurnal Ilmiah Teknik Elektro Komputer dan Informatika (JITEKI)
Vol. 11, No. 2, June 2025, pp. 225-237
ISSN: 2338-3070, DOI: 10.26555/jiteki.v11i2.30675

Optimizing Machine Learning-Based Network Intrusion Detection System with Oversampling, Feature Selection and Extraction

Rama Wijaya Shiddiq[1], Nyoman Karna[2], Indrarini Dyah Irawati[3]
[1]School of Electrical Engineering, Telkom University, Bandung 40257, Indonesia
[2]The University Center of Excellence for Intelligent Sensing-IoT, Telkom University, Indonesia
[3]School of Applied Science, Telkom University, Bandung 40257, Indonesia

ARTICLE INFO

Article history:

Received February 05, 2025
Revised March 17, 2025
Accepted May 03, 2025

Keywords:

Machine Learning;
Network Intrusion Detection System;
Imbalanced Dataset Handling in NIDS;
Feature Selection and Extraction;
Optuna

ABSTRACT

Network security is a global challenge that requires intelligent and efficient solutions. Machine Learning (ML)-based Network Intrusion Detection Systems (NIDS) have been proven to enhance accuracy in detecting cyberattacks. However, the main challenges in implementing ML-based IDS are dataset imbalance and large dataset size. This research addresses these challenges by applying oversampling techniques to balance the dataset, feature selection using random forest to identify the most relevant features, and feature extraction using Principal Component Analysis (PCA) to further reduce the selected important features. Additionally, K-fold cross-validation is used to test the features to minimize bias and ensure the model does not suffer from overfitting, while Optuna is implemented to automatically optimize model parameters for maximum accuracy. Since IDS performance deteriorates with high-dimensional features, the combination of methods used is evaluated based on feature selection applied to the model using datasets wtih 45 features selected from UNSW-NB15, 78 features from CIC-IDS-2017, and 80 features from CIC-IDS-2018 using various ML algorithms. The results demonstrate that the combination technique with feature selection, along with maximum optimization for each model significantly improves performance on large and imbalanced datasets reaching 99% accuracy compared to conventional methods in network traffic analysis.

Corresponding Author:

Nyoman Karna, The University Center of Excellence for Intelligent Sensing-IoT, Telkom University, Bandung 40257, Indonesia.
Email: aditya@telkomuniversity.ac.id

1. INTRODUCTION

The rapid development of information technology has made computer networks a very important infrastructure for various sectors, including organizations, companies, and individuals. However, as the complexity of networks increases, so do the challenges in maintaining their security. Cyber threats such as Denial-of-Service (DoS), malware attacks, and intrusions into networks are constantly evolving and becoming more sophisticated, demanding more adaptive and intelligent security solutions. To overcome this challenge, Network Intrusion Detection System (NIDS) is one of the widely used approaches to detect threats by analyzing network traffic and identifying suspicious activities in real-time.

Although NIDS have been widely implemented in network security systems, conventional signature-based or rule-based methods have limitations in detecting new undocumented threats, such as zero-day attacks and encrypted threats. These limitations lead to high false positive rates and decreased accuracy in identifying more complex attacks. Therefore, a more adaptive approach is needed to enhance the efficiency of NIDS [1]. The integration of ML significantly enhances IDS performance by reducing false positive rates, improving accuracy of threat detection, and dynamically adapting to evolving cybersecurity risks [5]. By leveraging

machine learning, security systems become more capable of safeguarding networks and sensitive information from malicious activities and unauthorized intrusions. In the modern cybersecurity landscape, developing robust models that effectively counteract cyber threats requires continuous advancements in training methodologies and data processing techniques. However, many contemporary ML-based IDS solutions face limitations due to their reliance on datasets that are often small, outdated, and artificially balanced. The use of such datasets, along with inconsistencies in data distribution, poses challenges in achieving high detection accuracy. The complexities associated with dataset preparation and algorithm selection play a crucial, with additional layer of difficulty to optimizing intrusion detection performance [20], [21]. Another more efficient approach was developed by Seth *et al.* [33], who applied the Hybrid Feature Selection (HFS) technique to minimize model complexity without sacrificing accuracy. With Light Gradient Boosting Machine (LightGBM) on the CIC-IDS2018 dataset, the model successfully accelerated the training time by 52.68% (to 17.94 seconds) and reduced the prediction latency by 44.52% (to 2.25 seconds), with an accuracy rate of 97.73%.

On the other hand, Talita *et al.* [32] used Naïve Bayes approach combined for feature selection on KDDCUP99 dataset. This dataset consists of more than 400,000 entries and 40 features, which were then reduced to 38 main features using PSO. This technique successfully improved computational efficiency and memory consumption, and resulted in a higher accuracy of 99.12%. In addition, research by Aghnia Fadhillah, Nyoman Karna, and Arif Irawan [33] evaluated the performance of an anomaly-based in the network traffic, which focused on DoS attack detection. The test results showed the developed IDS was able to detect 9,421 Torshammer packets, 10,618 Xerxes packets, and 6,115 LOIC packets on the FTP server, with an accuracy rate of 88.66%, although it still faces challenges in reducing the false positive rate of 63.17%. Employing this innova network activities, particularly DoS attacks was presented by Aghnia Fadhillah, Nyoman Karna, Arif Irawan [33]. The IDS successfully detected numerous attack packets, including 9,421 from Torshammer, 10,618 from Xerxes, and 6,115 from LOIC on the FTP server. On the Web server, it identified 299 packets from Torshammer, 530 from Xerxes, and 103 from LOIC. Overall, the IDS achieved an accuracy of 88.66%, a precision of 88.58%, and a false positive rate of 63.17%. Although various studies have proposed more advanced methods, there are still significant challenges in the optimization and efficiency of ML-based NIDS systems, especially in handling large-scale datasets, data imbalance, and feature optimization. Many developed models still have difficulties in handling complex network traffic, reducing detection errors, and improving computational efficiency in a real-time environment. Therefore, a new approach is needed that can overcome these limitations more effectively. This research contribution aims to develop a more efficient and adaptive Machine Learning-based NIDS framework by applying several combinations of techniques applied to each model. From the research results, the combination of techniques used is able to handle large and unbalanced datasets. In addition, the maximum level of accuracy with the addition of maximum optimization on each model parameter. Overall testing has been carried out by considering the parameters of accuracy, precision, recall, F1-score, ROC curve and traffic. Because if you do not consider other parameters, you will experience bias and overfitting in the results.

2. METHODS

The methodology research proposed framework along with the data preprocessing methods applied, including oversampling techniques, feature selection, and feature extraction. Additionally, a concise overview of the machine learning model. The designed framework focuses on selecting and extracting relevant features to enhance intrusion detection performance, particularly when handling large and imbalanced datasets.

The machine learning model training process is divided into two training data and test data. The dataset undergoes pre-processing to balance the data, followed by feature engineering to extract influential features that impact accuracy. Next, classifier algorithms are applied to build various models, which are then trained and tested. The evaluation phase assesses performance using metrics such as accuracy and traffic parameters. The final results indicate how well the model predicts normal and anomalous traffic detection [9].

2.1. System Model

The proposed framework for intrusion detection involves several key steps to improve model performance at Fig. 2 [14]. Fig. 2 This research adopts three main stages in the modeling process, namely oversampling to handle data imbalance, feature selection using Random Forest, and hyperparameter optimization with Optuna. Oversampling is used to increase the amount of samples in the minority class to reduce model bias. The SMOTE to generate synthetic samples by interpolating existing data points, thus not only increasing the number of samples in the minority class, but also enriching the data, to prevent overfitting that can occur due to data duplication from ordinary random oversampling techniques.

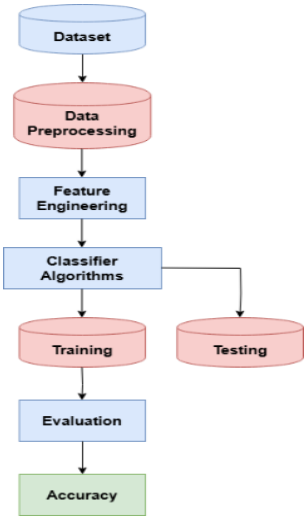


Fig. 1. Research Stages

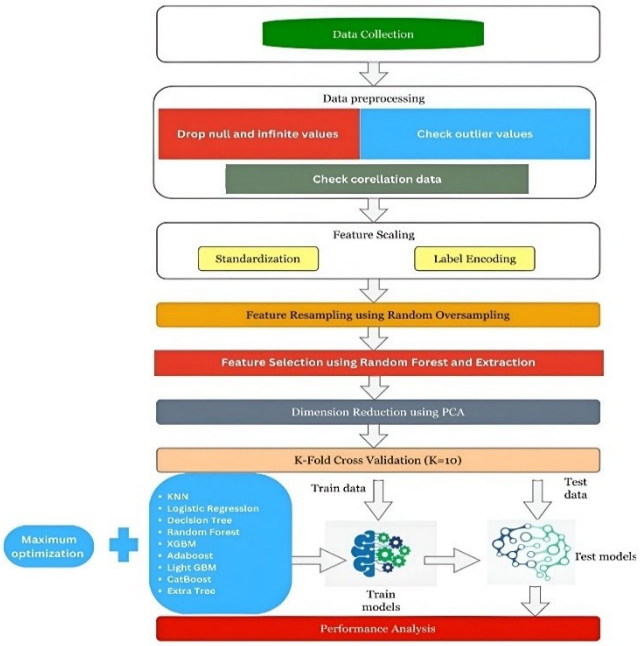


Fig. 2. Proposed System Architecture

After dealing with data imbalance, the next stage is feature selection using Random Forest, which aims to identify influential features in the model. Random Forest Importance Score is used to rank the features based on their contribution to the model prediction, and only features that score higher than a certain threshold are retained. In addition, a statistical significance test (P-value < 0.05) was applied to ensure that only statistically significant features were included in the model. Features with a P-value greater than 0.05 were considered to have no statistically significant influence and were removed to reduce noise in the data as well as improve model efficiency.

To reduce model complexity and improve computational efficiency, this research applies dimensionality reduction using PCA works by transforming the original feature space into components that capture the maximum variance in the data. Thus, PCA not only enhances data processing efficiency, but also reduces the risk of the curse of dimensionality, which can lead to overfitting when models handle datasets with a very large of features. Also, K-fold cross-validation is added, where the dataset is divided into K equal parts, then each part is used in turn as test data, while the other part is used as training data. This technique is used to avoid overfitting and provide a more accurate estimate of the model's performances compared to the conventional

train-test split method. In addition, optimization using Optuna was applied to optimize the parameter values in each model [26].

2.2. Dataset

For our research programs, we have used three benchmark big datasets: CIC-IDS2017 [12], UNSW-NB15 [14] and : CIC-IDS2018 [14] Both datasets have the most recent attack categories to identify attacks to the NIDS environments. Table 1 the dataset consists of 257,673 samples, of which 36.09% is normal traffic and 63.91% is malicious activity. The attacks with the highest number are Generic (22.85%), Exploits (17.28%), and Fuzzers (9.41%), which shows that automated exploits are often used in cyberattacks. The DoS (6.35%) and Reconnaissance (5.43%) categories are also quite significant, signaling reconnaissance and service disruption attempts. Meanwhile, lower numbers of attacks such as Backdoor (0.90%), Shellcode (0.59%), and Worms (0.07%) indicate more specific attack methods. Table 2 the dataset consists of 28,30,743 samples, of which 80.3% is BENIGN (normal) traffic and 19.7% is various attack categories. DoS Hulk (8.16%), PortScan (5.61%), and DDoS (4.52%) attacks dominate the threat categories, indicating that flooding and network scanning-based attacks are common. Meanwhile, other attacks such as DoS GoldenEye (0.36%), FTP-Patator (0.28%), and SSH-Patator (0.21%) have smaller numbers, but are still relevant in security analysis. Rare attacks such as Infiltration (0.01%) and Heartbleed (0.01%) reflect specific exploitation of system vulnerabilities. Table 3 this dataset contains 9,33,277 samples, with 70.55% Benign (normal) traffic and 29.45% consisting of various categories of cyberattacks. The dominant attacks are DDOS attacks-HOIC (7.35%), DDoS attacks-LOIC-HTTP (6.17%), and DoS Attacks-Hulk (4.95%), which reflect the main threats to network systems.

Table 1. Attack categories of the UNSW-NB15 dataset

Attack Categories	Count	% (percentage)
Normal	93000	36.09
Generic	58871	22.85
Exploits	44525	17.28
Fuzzers	24246	9.41
DoS	16353	6.35
Reconnaissance	13987	5.43
Analysis	2677	1.04
Backdoor	2329	0.90
Shellcode	1511	0.59
Worms	174	0.07
Total	257673	100

Table 2. Attack categories of the CIC-IDS2017 dataset

Attack categories	Count	% (percentage)
BENIGN	22,73,097	80.3
DoS Hulk	2,31,073	8.16
PortScan	1,58,930	5.61
DDoS	1,28,027	4.52
DoS GoldenEye	10,293	0.36
FTP-Patator	7938	0.28
SSH-Patator	5897	0.21
DoS slowloris	5796	0.2
DoS Slowhttptest	5499	0.19
Web Attack	2180	0.08
Bot	1966	0.07
Infiltration	36	0.01
Heartbleed	11	0.01
Total	28,30,743	100

In addition, categories such as Bot (3.07%), FTP-BruteForce (2.07%), and SSH-Bruteforce (2.01%) show the importance of authentication-based attack detection. With the presence of specific attacks such as SQL Injection (0.001%) and Brute Force - XSS (0.002%), this dataset covers a wide range of threat vectors that can be usedo train and evaluate machine learning for cyberattack models. Therefore, this dataset is highly relevant for network security research in developing more accurate and adaptive intrusion detection systems.

Table 3. Attack categories of the CIC-IDS2018 dataset\

Attack Categories	Count	(%) Percentage
Benign	6,58,454	70.553
DDOS attack-HOIC	68,601	7.351
DDoS attacks-LOIC-HTTP	57,619	6.174
DoS attacks-Hulk	46,191	4.949
Bot	28,619	3.067
FTP-BruteForce	19,336	2.072
SSH-Bruteforce	18,759	2.01
Infiltration	16,193	1.735
DoS attacks-SlowHTTPTest	13,989	1.499
DoS attacks-GoldenEye	4,151	0.445
DoS attacks-Slowloris	1,099	0.118
DDOS attack-LOIC-UDP	173	0.019
Brute Force -Web	61	0.007
Brute Force -XSS	23	0.002
SQL Injection	9	0.001
Total	9,33,277	100

2.3. Testing Scenarios and Parameters

The initial phase of the process begins with data acquisition from the selected dataset. This data undergoes several preprocessing steps, including the removal of null and infinite values, detection and handling of outliers, analysis of correlations between variables, conversion of categorical variables into binary or numerical representations through label encoding, and standardization to address values that are excessively large or small, while the provision of 40 GB of storage space ensures sufficient capacity for managing and processing large datasets. The experiments are executed using Google Colab. The confusion matrix will represent all models as a structured table containing four possible prediction outcomes in comparison with actual values: True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN).

Table 4. Confusion Matrix

	Actual Positive	Actual Negative
Predicted Positive	TP	FP
Predicted Negative	FN	TN

2.3.1. Accuracy

For in the first step, the process to collection from the dataset. The data will process with steps such as removing null and infinite values, checking for outliers, checking for correlation between variables, converting categorical variables to binary or numeric (label encoding), and standardizing to deal with excessively large or small values.

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

2.3.2. Precision

Accuracy rate represents the ratio of classified instances to test samples. It is computed using the following:

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

The count of accurately predicted positive samples is referred to as True Positive (TP), while the number of negative samples that are incorrectly classified as positive is termed False Positive (FP)

2.3.3. Recall

The recall rate is the proportion of positive samples that are accurately identified.

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

On the other hand, the number of positive samples that are anticipated to be negative samples is known as false negatives (FN).

2.3.4. F1-Score

In classification problems, the F1-score represents the harmonic mean of precision and recall.

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

2.3.5. K-fold Cross-Validation (CV)

Cross validation is a model performance objectively. The objective is to get a reliable estimate of how the model to prevent overfitting, which occurs when the model fits the training and testing data but struggles to generalize to new data. During the cross-validation process, the dataset is split into two is the training set, which is used for training, and the validation or test set, which is used for testing the model. observed while training. One of the most crucial methods for testing and evaluating machine learning models is cross-validation. It is the process of evaluating a model to determine how well it generalizes to data that was not present during training [13].

2.3.6. ROC Curve

The Receiver Operating Characteristic (ROC) curve is a tool that illustrates the effectiveness of a diagnostic test is evaluated while the optimal cut-off point is used to classify conditions correctly. Numerous studies have employed empirical ROC curves to describe diagnostic accuracy rates, as this approach preserves the empirical distribution function's characteristics and remains independent of any theoretical distribution. However, when dealing with small sample sizes, the empirical ROC method may produce less reliable results due to variations in AUC calculations. Consequently, an alternative approach is necessary to improved the accuracy of ROC curves in such cases. To address this issue, the smoothed empirical ROC approach has been proposed as a solution, allowing for improved diagnostic performance evaluation, particularly in datasets with limited samples. By employing a smoothed empirical estimator, the sensitivity and 1-specificity values can be determined and represented graphically using the rotated ordinal graph method. Additionally, the trapezoidal rule is applied to calculate the area under the curve, ensuring more precise and reliable results. [12].

3. RESULTS AND DISCUSSION

Machine learning models across three datasets due to feature engineering and optimization techniques. The use of oversampling, focused feature selection and maximum optimization in each model is effective enough to improve accuracy especially when applied to large and unbalanced datasets. In this study deliberately tested various algorithms to see the performance of each model, the results of 9 models almost have an accuracy rate of 99% although some algorithms do not achieve this accuracy because they cannot capture non-linear and liner relationships on relevant features. High accuracy does not always reflect good performance, but in this study it has included testing other parameters, namely precision, recall, F1-Score and ROC Curve to see the level of false positives and false negatives. Because by considering these parameters, it can be seen that each model in addition to having high accuracy can also overcome overfitting and bias.

This research shows significant performance improvement compared to the previous approach that used stacking embedding with clustering and only applied four models without maximum optimization. The previous approach had difficulty in handling feature complexity and class imbalance and only testing led to lower accuracy. By utilizing maximum optimization techniques in each model parameter, the model in this study successfully overcomes these challenges and results in a significant increase in accuracy. This was proven in previous research [14], where better feature engineering and the use of optimization techniques resulted in a more accurate and reliable model.

3.1. Results of UNSW-NB15 Dataset

Table 5 shows the evaluation results, the most suitable models for anomaly detection in a Network Intrusion Detection System (NIDS) are CatBoost (99.16%), XGBM (99.15%), and the Voting Ensemble (99.15%). These models exhibit the highest accuracy and robust performance, making them ideal for detecting anomalies in network traffic. Their superiority lies in their ability to handle complex data distributions, manage noisy and imbalanced data, and leverage ensemble learning techniques to combine the strengths of multiple models. These features enable them to accurately differentiate between normal and malicious traffic, making them highly reliable for real-world deployment. Models like Random Forest (99.04%), LightGBM (98.90%), and Decision Tree (98.06%) also perform strongly and are suitable for NIDS tasks.

Table 5. Performance Analysis of all model on UNSW-NB15 Dataset

Model	Accuracy Scores	Precision
KNN	83.38%	85%
Logistic Regression	75.12%	79%
Decision Tree	98.06%	98%
Random Forest	99.04%	99%
XGBoost	99.16%	99%
Adaboost	94.70%	95%
Light GBM	98.90%	99%
CatBoost	99.16%	99%
Extra Tree	78.98%	99%

3.2. Results of CIC-IDS2017 Dataset

Table 6 shows the evaluation results, the most suitable models for anomaly detection are XGBM (99.43%), Random Forest (99.37%), and Extra Tree (99.30%). These models demonstrate the highest accuracy and robust performance, making them well-suited for detecting anomalies in network traffic. Their effectiveness lies in their ensemble-based techniques, which combine multiple decision trees to handle complex data patterns and imbalanced distributions. This allows them to effectively differentiate between normal and malicious in the network traffic, ensuring high reliability for real-world applications. CatBoost (99.21%) and LightGBM (99.40%) also perform strongly, leveraging boosting techniques to achieve high accuracy while maintaining computational efficiency.

Table 6. Performance Analysis of all model on CIC-IDS2017 Dataset

Model	Accuracy Scores	Precision	Recall	F1-Score
KNN	83.38%	85%	83%	83%
Logistic Regression	75.12%	79%	78%	77%
Decision Tree	98.06%	98%	98%	98%
Random Forest	99.04%	99%	99%	99%
XGBoost	99.16%	99%	99%	99%
Adaboost	94.70%	95%	95%	95%
Light GBM	98.90%	99%	99%	99%
CatBoost	99.16%	99%	99%	99%
Extra Tree	78.98%	99%	99%	99%

On the other hand, models like Adaboost (59.62%) and Logistic Regression (63.75%) show significantly lower performance. Adaboost's reliance on weak classifiers limits its ability to manage complex datasets effectively, while Logistic Regression's linear decision boundaries struggle to capture patterns in network traffic data. Models like KNN (98.49%) perform moderately well but are computationally expensive in high-dimensional data, making them less ideal for large- scale NIDS tasks.

3.3. Results of CIC-IDS2018 Dataset

Table 7 shows the evaluation results, the most suitable models to detect anomalies in a Network Intrusion Detection System (NIDS) are Extra Trees (99.98%), and Random Forest (99.97%). These models exhibit the highest accuracy and robust performance, making them ideal for detecting anomalies in network traffic. Their superiority lies in their ability to handle complex data distributions, manage noisy and imbalanced data, and leverage ensemble learning techniques to combine the strengths of multiple models. These features enable them to accurately differentiate between normal and malicious traffic, making them highly reliable for real-world deployment. Models like LightGBM (99.96%), CatBoost (99.95%), and Decision Tree (99.97%) also perform strongly and are suitable for NIDS tasks. Especially in the case of LightGBM. However, their performance slightly lags behind the top models, indicating room for further optimization.

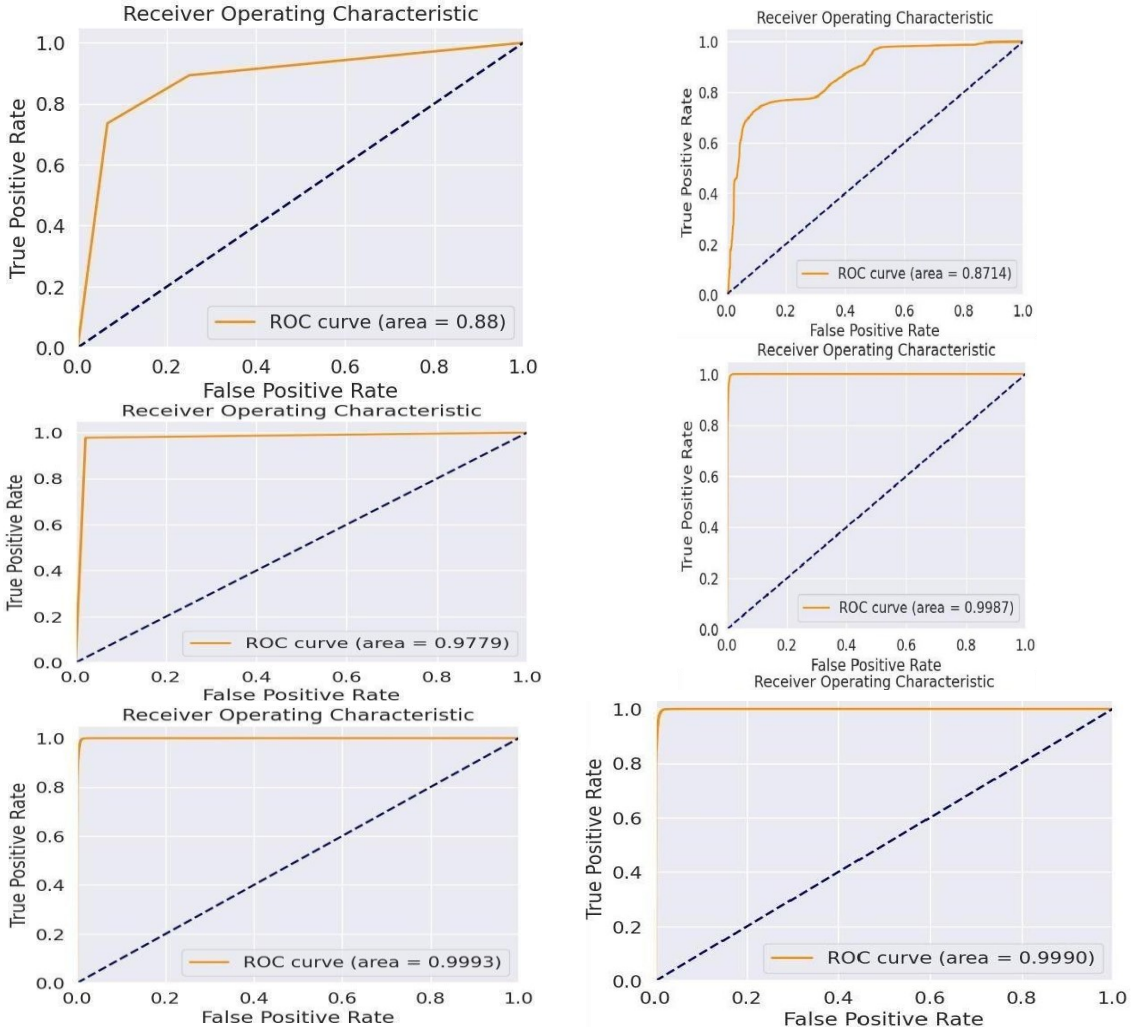
In the results of this study, models with 100% precision (such as KNN, Decision Tree, Random Forest, LightGBM, CatBoost, and Extra Trees) showed no false positives, which means that all attack predictions made by the models were real attacks. However, in real-world applications, it is necessary to ensure that these results remain consistent on new, more complex and dynamic data. A high false negative rate in an IDS can result in attacks that escape detection, potentially leading to data theft, system exploitation, or even overall security system failure. In this study, models with 100% recall showed that they did not miss any attacks, meaning there were no false negatives. However, in models with recall below 100% (such as Logistic Regression and XGBoost), there is a possibility of undetected attacks. Overall, this study was able to produce good false positive and false negative rates in each model.

Table 7. Performance Analysis of all model on CIC-IDS2018 Dataset

Model	Accuracy Scores	Precision	Recall	F1-Score
KNN	99.93%	100%	100%	100%
Logistic Regression	63.53%	79%	64%	58%
Decision Tree	99.98%	100%	100%	100%
Random Forest	99.98%	100%	100%	100%
XGBoost	99.16%	99%	99%	99%
Adaboost	99.44%	99%	99%	99%
Light GBM	99.97%	100%	100%	100%
CatBoost	99.96%	100%	100%	100
Extra Trees	99.99%	100%	100%	100%

3.4. ROC Curve Result in Each Model

Fig. 3 illustrates The ROC Curve results in the figure show very high AUC (Area Under the Curve) values, with some models approaching 1.0, indicating near-perfect classification performance. Models with high AUC indicate that they have a very low false positive rate (FPR) and high true positive rate (TPR), so they can detect attacks with high accuracy without much error in classifying normal activity as a threat (false positive). However, in models with lower AUC such as 0.88 or 0.8714, there is a higher probability of false negatives (FN), which means some attacks may not be detected. In real applications such as Intrusion Detection System (IDS), high false negatives can be a big risk as threats can slip through undetected, while high false positives can lead to too many false alarms that burden the system and security operators. Therefore, models with high AUC above 0.99 are more recommended for IDS implementation as they have an optimal balance between attack detection and minimal false alarms.



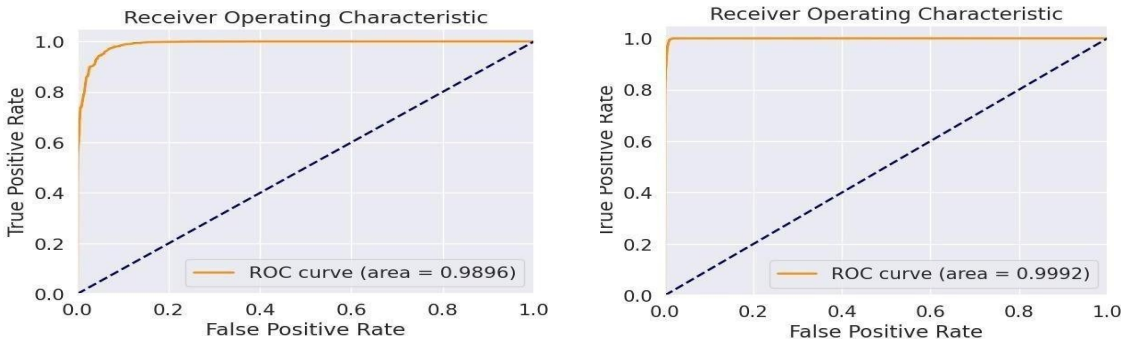


Fig. 3. The ROC Curve

3.5. Results of NIDS Traffic Parameters

Fig. 4 presents an analysis of NIDS traffic parameters in the UNSW-NB15 dataset, highlighting that the most influential feature for anomaly detection is attack_cat, which categorizes various attack types and plays a crucial role in distinguishing normal from anomalous traffic. The second most influential feature is id, which facilitates session or packet identification, aiding dataset structuring and supporting machine learning models. Features like ct_dst_sport_ltm (long-term count of destination port occurrences) and also essential for detecting suspicious activities like port scanning and brute force attacks. Additionally, the Source Load (sload) metric indicates system load, often associated with high-traffic DDoS attacks.Preprocessing techniques, including data normalization and encoding, enhance model efficiency. The findings suggest that feature selection in intrusion detection models plays a vital role classification accuracy. By applying oversampling techniques, issues related to data imbalance in features like attack_cat can be addressed. Furthermore, feature extraction methods help create more relevant attributes, reducing complexity and optimizing detection performance.

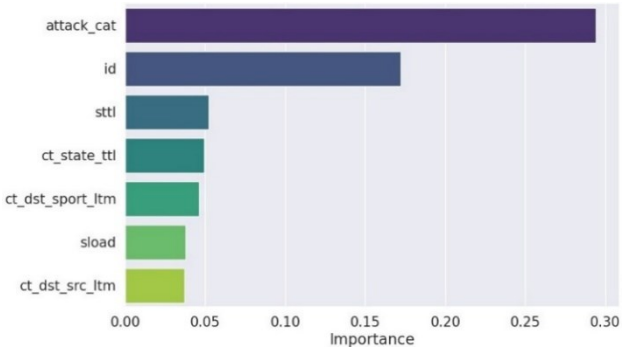


Fig. 4. Results of Parameters Traffic UNSW-NB15 Dataset

Fig. 5 illustrates the analysis of NIDS traffic parameters using the CIC-IDS-2017 dataset, revealing that the Init_Win_bytes_backward feature has the greatest impact on anomaly detection. This feature signifies the initial bytes received in the backward, which plays a role in differentiating between normal and anomalous traffic patterns. The second most influential feature is Init_Win_bytes_forward, highlighting the importance of the initial window bytes in the forward direction for anomaly detection. Other parameters, including Min Bwd Packet Length exhibit a moderate influence, while min_seg_size_forward and Avg Fwd Segment Size have a minor yet still relevant contribution in detecting specific traffic patterns. To optimize the NIDS model, priority should be given to key features, while less impactful features can be re- evaluated to simplify the model. Additionally, oversampling techniques can be employed to address data imbalance, and feature extraction method.

Fig. 6 presents the analysis of traffic parameters from the CIC-IDS-2018 dataset for machine learning-driven NIDS. The results highlight that the Total Length of Bwd Packets is the most influential parameter in identifying anomalies. This feature represents the cumulative size of packets transmitted in the backward direction, which frequently serves as a crucial indicator for detecting abnormal network. Additionally, other significant features include Bwd Packet Length Mean, Min Packet Length, and Flow Duration, which contribute to anomaly detection by preserving the most relevant information.

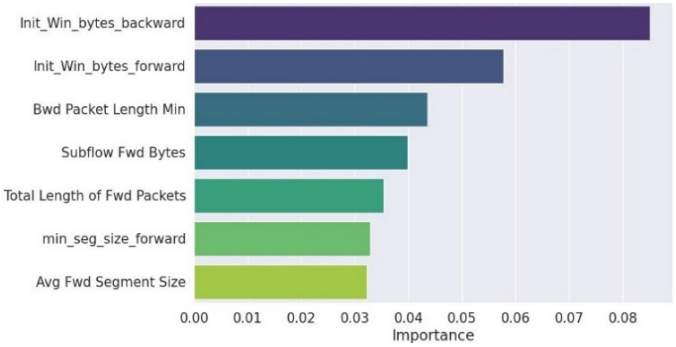
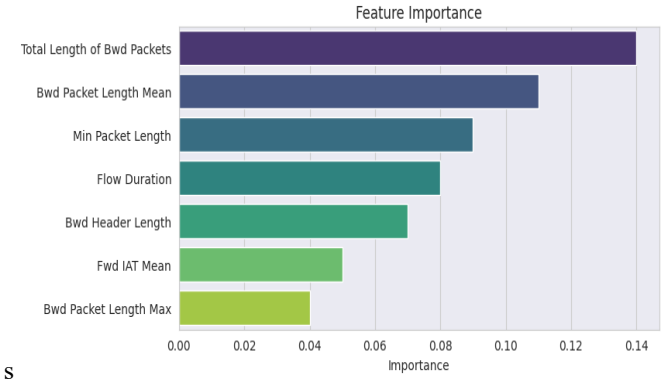


Fig. 5. Results of Parameters Traffic CIC-IDS2017 Dataset



s

Fig. 6. Results of Parameters Traffic CIC-IDS2018 Dataset

4. CONCLUSION

This study presents an optimized NIDS through a combination of data preprocessing, feature selection, oversampling, and hyperparameter optimization. The experimental results indicate that ensemble models, particularly Extra Trees (99.98%) and Random Forest (99.97%), excel in anomaly detection with high accuracy, robustness, and efficiency. LightGBM (99.96%) and CatBoost (99.95%) also demonstrate strong performance, whereas Logistic Regression (63.53%) and KNN (99.92%) struggle with large and imbalanced datasets.

The key contributions of this research include oversampling using SMOTE to address class imbalance and feature selection through Random Forest, PCA, and K-fold validation to reduce redundant features and select the most relevant ones. Additionally, maximum optimization using Optuna was employed to fine-tune model parameters. The validation conducted on three benchmark datasets confirms the effectiveness of this approach in various network attack scenarios. From a theoretical perspective, this research provides insights into the application of machine learning techniques for improving anomaly detection. However, certain limitations exist, such as the lack of real-time network testing and challenges related to computational efficiency and model interpretability. For future research, exploring hybrid models that integrate deep learning, enhancing interpretability using explainable AI, testing in real-world network environments, and implementing NIDS on edge computing are recommended.

Acknowledgments

This research was partially supported by Telkom University. The authors extend their sincere gratitude to the Faculty of Electrical Engineering, Telkom University, Indonesia, for their valuable support. Their endorsement of the final manuscript for publication is deeply appreciated.

REFERENCES

[1] M. Sarhan, S. Layeghy, N. Moustafa, M. Gallagher, and M. Portmann, "Feature extraction for machine learning-based intrusion detection in IoT networks," *Digital Communications and Networks*, vol. 10, no. 1, pp. 205–216, Feb. 2024. https://doi.org/10.1016/j.dcan.2022.08.012.

[2] Y. Kim and J. Kim, "CNN-LSTM: Hybrid Deep Neural Network for Network Intrusion Detection System," in *IEEE Transactions on Network and Service Management*, vol. 19, no. 4, pp. 5125–5138, Dec. 2023, https://doi.org/

10.1109/9889698.
[3] R. Kumar and A. K. Singh, "A Deep Learning Approach to Network Intrusion Detection," in *IEEE Access*, vol. 30, no. 3, pp. 7856–7868, May 2021, https://doi.org/10.1109/8264962.
[4] S. K. Sharma, V. S. Kushwaha, and T. H. Kim, "IoT Intrusion Detection System Using Deep Learning and Enhanced Transient Search Optimization," in *IEEE Internet of Things Journal*, vol. 9, no. 5, pp. 11248–11260, Sept. 2022, https://doi.org/10.1109/9525369.
[5] A. R. Rashed and W. A. Rizk, "Machine Learning-Powered Encrypted Network Traffic Analysis: A Comprehensive Survey," *in IEEE Transactions on Information Forensics and Security*, vol. 18, no. 2, pp. 312–329, Feb. 2024, https://doi.org/10.1109/9896143.
[6] M. W. Oh, P. S. Kim, and J. K. Noh, "Ensemble Learning Approach for Network Intrusion Detection using Hybrid Feature Extraction," *IEEE Access*, vol. 10, pp. 157395-157406, Apr. 2022, https://doi.org/10.1109/ACCESS.2022.3195505.
[7] H. Zhao, X. Sun, and Y. Liu, "A Deep Reinforcement Learning Approach for Anomaly Network Intrusion Detection System," in *Proceedings of the IEEE International Conference on Communications (ICC),* pp. 102–108, Jan. 2023, https://doi.org/10.1109/9335796.
[8] R. A. Disha and S. Waheed, "Performance analysis of machine learning models for intrusion detection system using Gini Impurity-based Weighted Random Forest (GIWRF) feature selection technique" Spring Open Access Journal, vol. 5, no. 1, pp. 1-22, 2022. https://doi.org/10.1186/s42400-021-00103-8.
[9] M. S. El-Masri, E. E. El-Alfy, and A. A. M. Sayad, "A Hybrid Machine Learning Framework for Intrusion Detection in IoT Systems," in *Proceedings of the IEEE International Conference on Industrial Informatics (INDIN*), pp. 12-17, 2022, https://doi.org/10.1109/INDIN49073.2022.9622198.
[10] Y. Zhang, C. Zhang, and X. Wang, "A Review of Machine Learning Methods for Network Intrusion Detection Systems*," IEEE Transactions on Neural Networks and Learning Systems*, vol. 34, no. 2, pp. 1183-1195, Feb. 2023, https://doi.org/10.1109/TNNLS.2022.3204521.
[11] M. A. M. Hossain, M. R. S. S. Sayeed, and M. M. Rahman, "Deep Learning-based Intrusion Detection System for Secure IoT Networks," in *IEEE International Conference on Smart Computing (SMARTCOMP),* pp. 208-214, 2022, https://doi.org/10.1109/SMARTCOMP53562.2022.9782321.
[12] F. Hussain *et al.,* "Machine learning in iot security: current solutions and future challenges," *IEEE Commun Surv Tutor*, vol. 22, no. 3,pp. 1686–1721, 2020, https://doi.org/10.1109/COMST.2020.2986444.
[13] M. Talukder *et al.,* "Machine learning-based network intrusion detection for big and imbalanced data using oversampling, stacking feature embedding and feature extraction," *Journal of Big Data,* vol. 11, 2024, https://doi.org/10.1186/s40537-024-00886-w.
[14] P. L. S. Jayalaxmi, R. Saha, G. Kumar, M. Conti, T. H. Kim, "Machine and Deep Learning Solutions for Intrusion Detection and Prevention in IoTs: A Survey," *IEEE Access,* vol. 10, pp. 121173-121192*,* 2022, https://doi.org/10.1109/ACCESS.2022.3220622.
[15] S. Khan, F. R. Khan, and N. Anwar, "Machine Learning-based Intrusion Detection System for IoT: A Survey and Comparative Study," in *IEEE Access*, vol. 9, pp. 95073-95087, Jul. 2021. https://doi.org/10.1109/ACCESS.2021.3099298.
[16] L. Zhang, Y. Wang, and S. Liu, "A Deep Learning Approach for Network Intrusion Detection System using Recurrent Neural Networks," *IEEE Access*, vol. 9, pp. 154693-154702, Oct. 2021. https://doi.org/10.1109/ACCESS.2021.3115683.
[17] W. Liu, M. H. Anwar, and H. T. H. Nguyen, "Real-time Network Intrusion Detection using Convolutional Neural Networks," in *IEEE Transactions on Network and Service Management*, vol. 18, no. 5, pp. 4629-4639, May 2022. https://doi.org/10.1109/TNSM.2022.3153501.
[18] R. K. Gupta, S. D. Bhatti, and K. S. Bawa, "A Comprehensive Survey on Feature Selection for Intrusion Detection Systems," in *IEEE Transactions on Information Forensics and Security*, vol. 17, no. 3, pp. 196-208, Mar. 2022. https://doi.org/10.1109/TIFS.2021.3071781.
[19] N. K. Meena, M. S. Soni, and S. A. Rathore, "Deep Neural Networks-based Network Intrusion Detection System for High-Dimensional Traffic Data," *IEEE Access,* vol. 9, pp. 105872-105881, Jun. 2021.
[20] A. Zarei, A. Mozaffari, and M. S. M. Sajadi, "Deep Learning Methods for Network Traffic Analysis and Intrusion Detection Systems," *IEEE Transactions on Cybernetics*, vol. 53, no. 8, pp. 3921-3931, Aug. 2023. https://doi.org/10.1109/TCYB.2022.3205680.
[21] Z. H. U. Guowei *et al.,* "Research on network intrusion detection method of power system based on random forest algorithm." *13th international conference on Measuriing Technology and Mechatronics Automation (ICMTMA)*. p. 374-379, 2021, https://doi.org/10.1109/ICMTMA52658.2021.00087.
[22] S. W. Kim, H. S. Kim, and B. D. Lee, "An Effective Intrusion Detection System Using Ensemble Learning for Network Traffic Classification," *in Proceedings of the IEEE International Conference on Communications (ICC),* pp. 1031-1036, 2022, https://doi.org/10.1109/ICC45636.2022.9836852.
[23] K. R. H. S. Gummadi, Y. K. S. Reddy, and R. H. Raj, "Machine Learning-Based Classification of Malicious Network Traffic," in *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 50, no. 5, pp. 1701-1709, May 2022. https://doi.org/10.1109/TSMC.2022.3208749.
[24] A. S. L. Xie, T. C. Y. Chan, and Z. S. Lin, "Comparing Deep Learning Approaches for NIDS on Cloud Networks," in *Proceedings of the IEEE International Symposium on Cloud Computing and Big Data*, pp. 121-126, 2021, https://doi.org/10.1109/ISCCBD53030.2021.9745523.

[25] S. Moualla, K. Khorzo, A. Jafar, "Improving the performance of machine learning-based network intrusion detection systems on the UNSW-NB15 dataset," *Comput Intel Neurosci*. pp. 1–13, 2021, https://doi.org/10.1155/2021/5557577.
[26] S. M. Kasongo, Y. Sun, "Performance analysis of intrusion detection systems using a feature selection method on the unsw-nb15 dataset," *J Big Data*. vol. 7, no. 1, pp. 1–20, 2020, https://doi.org/10.1186/s40537-020-00379-6.
[27] P. Nimbalkar, D. Kshirsagar, "Feature selection for intrusion detection system in internet-of-things (IOT)," *ICT Express*, vol. 7, no. 2, pp. 177–181, 2021, https://doi.org/10.1016/j.icte.2021.04.012.
[28] P. S. Hwang, Y. D. Lee, and T. W. Choi, "Evaluating Machine Learning Algorithms for Real-Time Intrusion Detection," in *Proceedings of the IEEE International Conference on Computational Intelligence (ICCI),* pp. 1158-1163, 2021, https://doi.org/10.1109/ICCI53462.2021.00022.
[29] M. Ahmad *et al.,* "Intrusion detection in internet of things using supervised machine learning based on application and transport layer features using unsw-nb15 data-set," *J Wirel Commun Netw.* pp. 1-23, 2021, https://doi.org/10.1186/s13638-021-01893-8.
[30] D. Kshirsagar, S. Kumar, "An efficient feature reduction method for the detection of DoS attack," *ICT Express*. vol. 7, no. 3, pp. 371–375, 2021, https://doi.org/10.1016/j.icte.2020.12.006.
[31] E. Mugabo *et al.,* "Intrusion detection method based on mapreduce for evolutionary feature selection in mobile cloud computing," *Int J Netw Secur,* vol. 23, no. 1, pp. 106–115, 2021, https://doi.org/10.6633/IJNS.202101_23(1).13.
[32] A. Talita, O. Nataza, Z. Rustam, "Naïve bayes classifier and particle swarm optimization feature selection method for classifying intrusion detection system dataset," In: *Journal of Physics: Conference Series*, p 012021, 2021, https://doi.org/10.1088/1742-6596/1752/1/012021.
[33] A. Fadhlillah, N. Karna, A. Irawan, "IDS Performance Analysis using Anomaly-based Detection Method for DOS Attack," *IEEE International Conference on Internet of Things and Intelligence Systems (IoTaIS)),* pp. 18-22, 2021, https://doi.org/10.1109/IoTaIS50849.2021.9359719.
[34] T. Rahmawati, R. W. Shiddiq, M. Sumpena, S. Setiawan, N. Karna, and S. Hertiana, "Web Application Firewall Using Proxy and Security Information and Event Management for OWASP Cyber Attack Detection*," IEEE International Conference on Internet of Things and Intelligence Systems (IoTaIS)),* pp. 280–285, Nov. 2023, https://doi.org/10.1109/IoTaIS60147.2023.10346051.
[35] H. Haugerud, H. N. Tran, N. Aitsaadi, and A. Yazidi, "A dynamic and scalable parallel Network Intrusion Detection System using intelligent rule ordering and Network Function Virtualization," *Future Generation Computer System*s, vol. 124, pp. 254–267, Nov. 2021, https://doi.org/10.1016/j.future.2021.05.037.
[36] T. Bajtoš, P. Sokol, and F. Kurimský, "Processing of IDS alerts in multi-step attacks[Formula presented]," *Software Impacts,* vol. 19, Mar. 2024, https://doi.org/10.1016/j.simpa.2024.100622.
[37] Z. Chiba, N. Abghour, K. Moussaid, O. Lifandali, and R. Kinta, "A Deep Study of Novel Intrusion Detection Systems and Intrusion Prevention Systems for Internet of Things Networks," in *Procedia Computer Science*, pp. 94–103, 2022, https://doi.org/10.1016/j.procs.2022.10.124.
[38] T. S. Pooja and P. Shrinivasacharya, "Evaluating neural networks using Bi-Directional LSTM for network IDS (intrusion detection systems) in cyber security*," Global Transitions Proceedings*, vol. 2, no. 2, pp. 448–454, Nov. 2021, https://doi.org/10.1016/j.gltp.2021.08.017.
[39] A. Adu-Kyere, E. Nigussie, and J. Isoaho, "Analyzing the effectiveness of IDS/IPS in real-time with a custom in-vehicle design," in *Procedia Computer Science*, pp. 175–183, 2024, https://doi.org/10.1016/j.procs.2024.06.013.
[40] M. A. Talukder *et al.,* "Machine learning-based lung and colon cancer detection using deep feature extraction and ensemble learning," *Expert Syst Appl*. 2022;, vol. 205(, no. 117):, p. 695-, 2022, https://doi.org/10.1016/j.eswa.2022.117695.
[41] A. R. Gad, A. A. Nashat, and T. M. Barkat, "Intrusion detection system using machine learning for vehicular ad hoc networks based on ToN-IoT dataset," *IEEE Access,* vol. 9, pp. 142206–142217, 2021, https://doi.org/10.1109/ACCESS.2021.3120626.
[42] G. Guo, "An intrusion detection system for the internet of things using machine learning models," in *3rd International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE)*, pp. 332–335, 2022, https://doi.org/10.1109/ICBAIE56435.2022.9985800.
[43] L. Qi, Y. Yang, X. Zhou, W. Rafque, and J. Ma, "Fast anomaly identification based on multiaspect data streams for intelligent intrusion detection toward secure Industry 4.0," *IEEE Transactions on Industrial Informatics,* vol. 18, no. 9, pp. 6503–6511, 2022, https://doi.org/10.1109/TII.2021.3139363.
[44] M. Shafq, Z. Tian, A. K. Bashir, X. Du, and M. Guizani, "CorrAUC: A malicious Bot-IoT traffic detection method in IoT network using machine-learning techniques," *IEEE Internet of Things Journal*, vol. 8, no. 5, pp. 3242–3254, 2021, https://doi.org/10.1109/JIOT.2020.3002255.
[45] S. I. Popoola, B. Adebisi, M. Hammoudeh, G. Gui, and H. Gacanin, "Hybrid deep learning for botnet attack detection in the internet-of-things networks," *IEEE Internet of Things Journal*, vol. 8, no. 6, pp. 4944–4956, 2021, https://doi.org/10.1109/JIOT.2020.3034156.
[46] T.-N. Dao and H. Lee, "Stacked autoencoder-based probabilistic feature extraction for on-device network intrusion detection," *IEEE Internet of Things Journal,* vol. 9, no. 16, pp. 14438–14451, 2022, https://doi.org/10.1109/JIOT.2021.3078292.
[47] A. R. Gad, A. A. Nashat, and T. M. Barkat, "Intrusion detection system using machine learning for vehicular ad hoc

networks based on ToN-IoT dataset," *IEEE Access,* vol. 9, pp. 142206–142217, 2021, https://doi.org/10.1109/ACCESS.2021.3120626.
[48] A. Fatani, A. Dahou, M. A. A. Al-Qaness, S. Lu, and M. A. Abd Elaziz, "Advanced feature extraction and selection approach using deep learning and Aquila optimizer for IoT intrusion detection system," *Sensors*, vol. 22, no. 1, p. 140, 2021, https://doi.org/10.3390/s22010140.
[49] J. Liu, D. Yang, M. Lian, and M. Li, "Research on intrusion detection based on particle swarm optimization in IoT," *IEEE Access,* vol. 9, pp.s 38254–38268, 2021, https://doi.org/10.1109/ACCESS.2021.3063671.

BIOGRAPHY OF AUTHORS

Rama Wijaya Shiddiq, received her Bachelor's degree in Telecommunication Engineering from Telkom University, Indonesia, in 2023, after earning a Diploma in Telecommunication Technology from the same institution in 2021. He is currently pursuing his Master's degree at the School of Electrical Engineering, Telkom University, Indonesia. He has been working as a IT Security Specialist at Astra Otoparts Tbk. His research about all cybersecurity, Email: ramawijayashi@student.telkomuniversity.ac.id.

Nyoman Bogi Aditya Karna, received the Ph.D. degree in electrical engineering and computer science from Bandung Institute of Technology, West Java, Indonesia, in 2018. He has been a full-time Lecturer with the School of Electrical Engineering, Telkom Higher School of Technology (now Telkom University), West Java, since 1999. His research interests include the intelligent IoT, cybersecurity, and the Internet of Drone Things. Email: aditya@telkomuniversity.ac.id.

Indrarini Dyah Irawati received the Ph.D. degree from the School of Electrical and Information Engineering, Institut Teknologi Bandung. She was with Telkom University as an Instructor, from 2007 to 2014, an Assistant Professor, from 2014 to 2019, and has been an Associate Professor, since 2019, and a Professor, since 2023. She has published 12 papers in reputed international journals, nine papers in Scopus international proceedings, ten intellectual property rights, and one book. Her main research interests include compressive sensing, watermarking, signal processing, artificial intelligence, the Internet of Things, and computer networks. Email: indrarini@telkomuniversity.ac.id.