SCIENCE AND INFORMATION TECHNOLOGY
SINTECH JOURNAL

SINTECH JOURNAL | ISSN 2598-7305 | E-ISSN 2598-9642
Vol. 8 No 2 – Agustus 2025 | https://s.id/sintechjournal
DOI: https://doi.org/10.31598
Publishing: Prahasta Publisher

Optimizing Ddos Attack Detection Performance Through Feature Selection In Machine Learning

Purnama Shiddiq M[1], Ferry Norman S[2], Luhur Bayu A[3]

[1,2,3]Magister IlmuKomputer, Fakultas Teknologi Informasi, Universitas Budi Luhur
Jl. Ciledug Raya, DKI Jakarta, Indonesia

e-mail: pshiddiq@gmail.com[1], ferry_ns@yahoo.com[2], luhur.bayuadi@budiluhur.ac.id[3]

Received : July, 2025	Accepted : August, 2025	Published : August, 2025

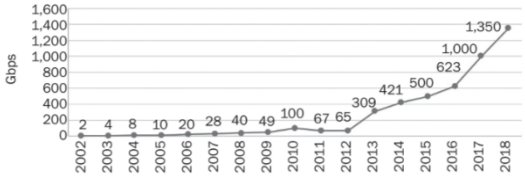
Abstract

Distributed Denial of Service (DDoS) attacks continue to pose significant challenges to cybersecurity infrastructure by overwhelming servers with massive traffic, rendering them inaccessible. Machine learning (ML) has become a critical tool for detecting such attacks efficiently. This study aims to enhance DDoS detection by applying and comparing three feature selection methods—Recursive Feature Elimination (RFE), Principal Component Analysis (PCA), and Information Gain (IG)—in conjunction with four ensemble-based classification algorithms: Random Forest (RF), LightGBM, XGBoost, and AdaBoost. The CIC-DDoS2019 dataset is utilized due to its diversity and representation of modern DDoS scenarios. The proposed approach evaluates each combination of feature selection and classification models based on accuracy, precision, recall, and F1-score. Furthermore, we incorporate k-fold cross-validation to ensure model robustness and assess computational efficiency during training and inference stages. The experimental results demonstrate that the combination of RFE with LightGBM yields superior performance across all evaluation metrics while maintaining low resource utilization. The novelty of this work lies in its systematic comparison of feature selection methods under hardware-aware constraints and its contribution to guiding efficient ML-based DDoS mitigation strategies. This study bridges the gap between detection accuracy and system efficiency, making it suitable for deployment in constrained environments such as edge devices or cloud-based intrusion detection systems.

Keywords: *DDoS, Cyber Attack, Machine Learning, Feature Selection, Computational Efficiency*

1. INTRODUCTION

The advancement of modern computing technology and global internet infrastructure has transformed human activities—social, business, and work—into increasingly online interactions. However, this progress has also given rise to various cyber threats, notably Distributed Denial of Service (DDoS) attacks. DDoS as one of the most severe threats, aiming to disrupt services by overwhelming targeted systems [1].



Picture 1. DDoS Traffic Intensity from 2002 to 2018 [2].

Picture 1, data from Akamai Inc. shows a steady increase in DDoS traffic intensity from 2002 to 2018, peaking at 1,350 Gbps. These attacks are often launched using botnets controlled by

malware, making real-time detection and mitigation highly challenging [3].

Marvi et al. (2021) demonstrated that reducing features by 77% using an Integrated Feature Selection (IFS) method improved classification performance by approximately 20% using LightGBM [8]. Omuya et al. (2021) applied PCA and Information Gain to reduce training time and enhance accuracy [9], while Upadhyay et al. (2021) found that combining Recursive Feature Elimination with XGBoost (RFE-XGBoost) led to higher accuracy and reduced misclassification [10].

Prior studies have employed a variety of ML algorithms for DDoS detection, including decision trees, support vector machines (SVM), and ensemble methods such as Random Forest (RF) and AdaBoost. Moreover, recent advances have explored deep learning architectures such as Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), and transformer-based models, which offer high detection accuracy by capturing temporal and spatial patterns in traffic data. Federated learning and stream-based intrusion detection systems (IDS) have also emerged, aiming to address privacy concerns and real-time processing requirements, respectively.

Despite these advancements, there is still a practical need to optimize detection performance while minimizing computational resource consumption, especially for deployment in resource-constrained environments such as edge computing or IoT networks. Deep learning models, while accurate, are often computationally intensive and less interpretable, making them less feasible for real-time or low-power scenarios.

To address these challenges, this study investigates the impact of feature selection techniques on the performance of ensemble-based machine learning classifiers for DDoS detection. Feature selection reduces data dimensionality by identifying the most relevant features, thereby improving model efficiency and interpretability. We evaluate three widely used feature selection techniques: Recursive Feature Elimination (RFE), Principal Component Analysis (PCA), and Information Gain (IG). These are combined with four ensemble learning models—Random Forest (RF), XGBoost, AdaBoost, and LightGBM—to assess their effectiveness in identifying DDoS attacks.

While existing literature has explored ML-based detection using individual feature selection techniques or classifiers, few studies have provided a comprehensive comparative analysis across multiple combinations under consistent evaluation settings. Furthermore, prior works rarely emphasize hardware-aware model efficiency, which is critical for deployment in constrained environments.[9].

2. RESEARCH METHOD

Various studies have explored the application of machine learning techniques for DDoS attack detection, emphasizing both classification accuracy and computational efficiency. Traditional models such as Decision Trees, Naïve Bayes, and Support Vector Machines (SVM) have demonstrated reasonable detection capabilities but often struggle with scalability and performance in high-dimensional data scenarios [1][2].

Ensemble learning algorithms, including Random Forest (RF), AdaBoost, and XGBoost, have gained popularity due to their ability to combine multiple weak learners to achieve higher accuracy and robustness. For instance, Roy et al. [3] applied Random Forest on the NSL-KDD dataset and reported improved performance over single classifiers. Similarly, Ahmed et al. [4] used XGBoost for classifying attack traffic and observed its superiority in handling imbalanced datasets.

Recent research has focused on integrating feature selection techniques with classification models to enhance performance. Recursive Feature Elimination (RFE) was used by Kumar et al. [5] to eliminate redundant features and improve model interpretability. Principal Component Analysis (PCA), a dimensionality reduction method, has been widely employed to reduce feature space and mitigate overfitting [6]. Information Gain (IG) has also been used to rank features based on their relevance to the target class [7].

However, most prior works investigate a single feature selection method in isolation, without conducting comprehensive comparisons across multiple selection strategies. Moreover, the

interaction effects between different feature selection methods and ensemble classifiers are rarely explored in a unified experimental setting.

In parallel, state-of-the-art deep learning techniques such as Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), and transformer-based models have been proposed to enhance detection accuracy. For example, Zhang et al. [8] used a CNN-RNN hybrid architecture to capture both spatial and temporal dependencies in traffic flow. Transformer-based models have shown promise in understanding complex feature relationships [9], while federated learning frameworks allow distributed model training across edge devices without exposing sensitive data [10]. Stream-based Intrusion Detection Systems (IDS) have also emerged to enable real-time threat detection on live traffic [11].

Despite their advantages, deep learning models typically require high computational resources, making them less feasible for resource-constrained environments such as IoT networks or embedded systems. In contrast, traditional ensemble classifiers with well-optimized feature selection offer a practical balance between accuracy and efficiency.

While Upadhyay et al. [12] and Barkah et al. [13] have attempted to combine feature selection with ensemble models, their studies often focus on performance metrics alone without considering resource utilization or reproducibility under varying conditions. Additionally, none provide detailed comparisons across multiple combinations of feature selection and classification techniques using a consistent framework.

2.1 Sampling/Selection Method

The sample used in this study is data from the CIC-DDoS2019 dataset. Sampling is done randomly to ensure a representative sample of DDoS attacks. The data is generally stored in user-friendly formats such as CSV and includes various features such as source and destination IP addresses, ports, number of bytes and packets transferred, as well as protocol information [6]. Further details on the class types and traffic volume for each DDoS attack category in the CIC-DDoS2019 dataset are presented in Table 1, titled "Class DDoS in the CIC-DDoS2019 Dataset."

Table 2: Class DDoS in the CIC-DDoS2019 Dataset [6].

Class	***Traffic***
Benign	56,863
DDoS_NetBIOS	4,093,279
DDoS_SNMP	5,159,870
DDoS_NTP	1,202,642
DDoS_TFTP	20,082,580
DDoS_SSDP	2,610,611
DDoS_SYN	1,582,289
DDoS_UDP-Lag	366,461
DDoS_DNS	5,071,011
DDoS_MSSQL	4,522,492
DDoS_LDAP	2,179,930
DDoS_UDP	3,134,645
DDoS_WebDDoS	439

The data will be divided into four groups: using raw data, processed data with RFE feature selection, PCA feature selection, and IG feature selection.

Dataset: CIC-DDoS2019

A. Data Source: Developed by the Canadian Institute for Cybersecurity specifically for DDoS attack detection research.
B. Data Characteristics: Includes normal network traffic and various types of DDoS attacks, stored in accessible formats such as CSV.
C. Evaluation and Comparison:
    a. Models (Random Forest, XGBoost, AdaBoost, and LGBM) will be trained and evaluated on the four types of datasets (raw, RFE, PCA, IG).
    b. Model effectiveness will be assessed based on metrics such as accuracy, precision, recall, and F1-score.
    c. Comparisons will assess the impact of feature selection on model performance and hardware requirements for DDoS attack detection.
D. Objective:
    a. Identify the most effective model and feature selection approach for DDoS attack detection.
    b. Enhance understanding of the performance and effectiveness of specific ML techniques in the context of cybersecurity.

c. Provide practical contributions to more efficient and effective information security practices.

2.2 Ensemble Classification Methods: Random Forest, XGBoost, AdaBoost, and LGBM

Theoretical Background and Model Operation

1. **Random Forest**
   Random Forest is an ensemble of decision trees trained on different data subsets and averaged to improve predictive accuracy. Each node selects a random set of features to compute results, and the aggregated outputs of all trees form the final classification [6].
2. **XGBoost**
   Ussatova et al. (2022) describe XGBoost as a high-performance machine learning algorithm introduced in 2014, featuring parallel tree boosting and optimized efficiency. One of its key features is the iterative boosting process, yielding superior classification performance over other algorithms [1].
3. **AdaBoost**
   AdaBoost is an ensemble technique that builds a strong classifier by combining multiple weak learners. It works in a sequential manner, where misclassified instances from previous models are emphasized in subsequent iterations. This process continues until a collection of base learners cooperatively achieve high classification accuracy [6].
4. **LGBM**
   Light Gradient Boosting Machine (LGBM) is a tree-based gradient boosting framework known for its efficiency and accuracy, particularly on large-scale datasets. LGBM is widely used for predictive modeling and classification tasks, makes it suitable for DDoS attack detection [8].

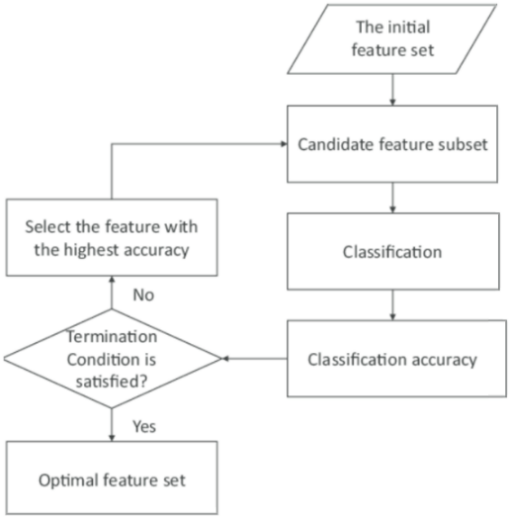
Given the unique characteristics of the CIC-DDoS2019 dataset and the challenges in DDoS attack detection, evaluating these ensemble models—Random Forest, XGBoost, AdaBoost, and LGBM—provides insights into their optimal application. This evaluation enables researchers to identify the most appropriate model by leveraging each algorithm's strengths while mitigating their limitations.

2.3 Influence of Feature Selection

Feature Selection Methods

Feature selection is a key process in machine learning that involves removing irrelevant or redundant attributes to enhance model performance. According to Omuya et al. (2021), eliminating unnecessary features improves the efficiency and accuracy of learning algorithms [9]. As Chandrashekar notes in the same study, selecting an optimal subset of features is crucial since these features serve as the core information source for building classification models [9].

This study compares three feature selection techniques—Recursive Feature Elimination (RFE), Principal Component Analysis (PCA), and Information Gain (IG)—to reduce data dimensionality and enhance classification performance by identifying the most relevant features. The architecture of the selection process is illustrated in Picture 2 (Feature Selection Flowchart).



Picture 2. Feature Selection Flowchart.

The applied methods are summarized below:

a) Recursive Feature Elimination (RFE)
   RFE iteratively eliminates less important features based on importance scores from trained models, identifying the most stable and predictive features for optimal model performance [10].
b) Information Gain (IG)
   IG evaluates feature relevance by measuring the reduction in entropy, helping select features that improve classification accuracy while keeping computational costs low [9].
c) Principal Component Analysis (PCA)

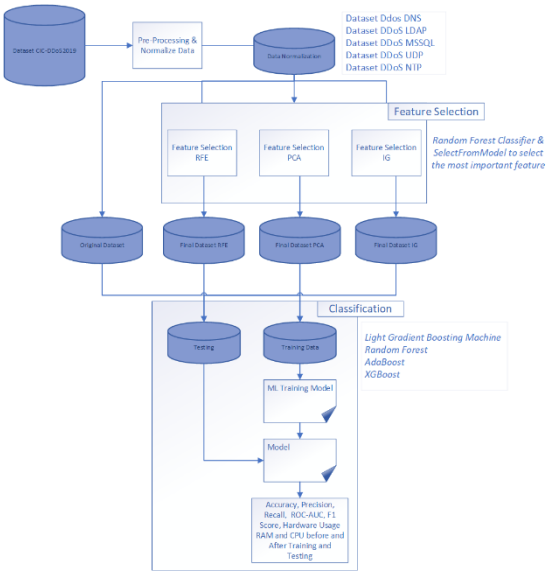
PCA is unsupervised linear method used for dimensionality reduction. It transforms the data into a lower-dimensional space while preserving as much statistical variance as possible, often selecting features most correlated with the principal components [7].

By reducing the number of features, these methods help minimize overfitting and improve the speed and efficiency of data processing. Effective feature selection is essential for developing accurate machine learning models for DDoS attack detection.

2.4 Analysis Technique, Model Design, and Testing Strategy

This study evaluates the performance of feature selection techniques—RFE, PCA, and Information Gain—combined with ensemble machine learning classifiers (Random Forest, XGBoost, AdaBoost, and LightGBM) using the CIC-DDoS2019 dataset. The analysis focuses on accuracy, processing time, and resource efficiency for DDoS detection.

Validating feature selection and ensemble methods requires multiple performance metrics such as precision, recall, F1-score, and failure rate. Experiments are conducted in a controlled environment to assess model performance under varying conditions [10]. The overall process is illustrated in Picture 3, outlining the research workflow.



Picture 3. Research workflow

2.5 Research Step

The research methodology, as visualized in Picture 3, consists of the following steps:

A. Data Collection and Preparation:

The CIC-DDoS2019 dataset is collected and preprocessed (including cleaning, handling missing values, and normalization). Four datasets are prepared: the raw dataset and three with features reduced via RFE, PCA, and Information Gain.

B. Model Training & Validation :

`n_estimators`: number of trees (range: 50 to 200)

`learning_rate`: boosting rate (range: 0.01 to 0.3)

`max_depth`: depth of each tree (range: 3 to 10)

Other model-specific parameters as applicable.

C. Model Implementation:

Ensemble classifiers—Random Forest, XGBoost, AdaBoost, and LightGBM—are implemented using appropriate libraries, with tuning of hyperparameters to optimize performance.

D. Performance Evaluation:

Accuracy: Overall correctness of predictions.

Precision : The ratio of true positives to predicted positives.

Recall : The ratio of true positives to actual positives.

F1-Score: Harmonic mean of precision and recall.

E. Experimental Environment

All experiments are conducted on a machine with the following specifications:

CPU: Intel Core i7-9750H @ 2.60GHz

RAM: 16 GB

OS: Ubuntu 20.04 LTS

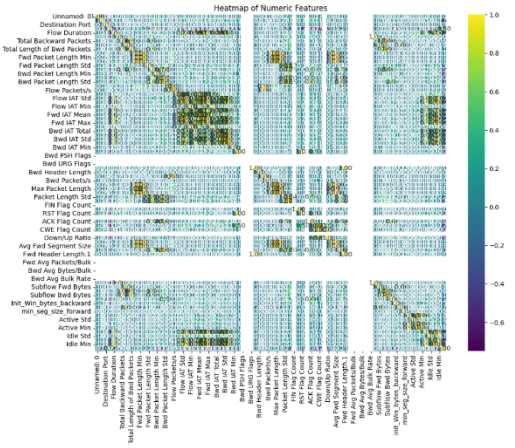
Libraries: Scikit-learn 1.2.2, LightGBM 3.3.2, XGBoost 1.7.5

3. RESULT AND DISCUSSION

3.1 Data Preparation

The CIC-DDoS2019 dataset was retrieved from the official repository of the Canadian Institute for Cybersecurity. It was selected due to its comprehensive coverage of various DDoS attack types alongside normal traffic, making it suitable for DDoS detection research. CIC-DDoS2019 consists of 83 features detailing network traffic attributes such as source/destination IPs, ports, protocols, packet sizes, and other traffic metrics. Descriptive statistics and data visualizations, such as heatmaps, were used to identify patterns and feature correlations, supporting informed preprocessing and feature selection.

Picture 4 illustrates a heatmap of numerical feature correlations for the DDoS MSSQL subset.



Picture 4. Heatmap Numerical Data DdoS MSSQL

3.2 Data Quality Assessment

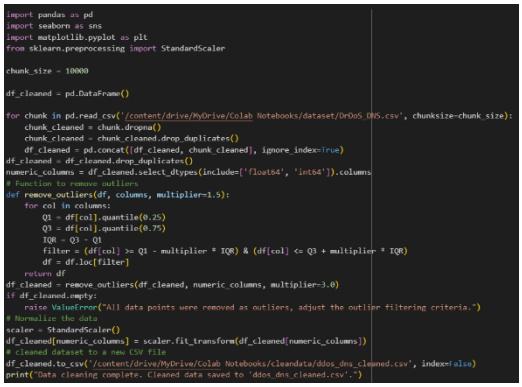
Ensuring data quality is a critical step in preprocessing to enhance model performance. This stage includes identifying and handling missing values, outliers, and performing data normalization.

Missing Values: Missing entries were detected and addressed using techniques such as row/column removal.

Outliers: Outliers, which may distort model performance, were identified through statistical analysis and visual tools like box plots.

Normalization: To bring all features onto a comparable scale—essential for many machine learning algorithms—data normalization was applied using Min-Max Scaling or Z-score normalization.

These preprocessing steps ensured a clean and consistent dataset for model training, thereby improving the reliability and accuracy of DDoS detection. Picture 5 presents the Python code used for handling missing values, outlier removal, and normalization, applied across all tested DDoS datasets prior to feature selection and classification.



```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.preprocessing import StandardScaler

chunk_size = 10000

df_cleaned = pd.DataFrame()

for chunk in pd.read_csv('/content/drive/MyDrive/Colab Notebooks/dataset/DrDoS_DNS.csv', chunksize=chunk_size):
    chunk_cleaned = chunk.dropna()
    chunk_cleaned = chunk_cleaned.drop_duplicates()
    df_cleaned = pd.concat([df_cleaned, chunk_cleaned], ignore_index=True)
df_cleaned = df_cleaned.drop_duplicates()
numeric_columns = df_cleaned.select_dtypes(include=['float64', 'int64']).columns
# Function to remove outliers
def remove_outliers(df, columns, multiplier=1.5):
    for col in columns:
        Q1 = df[col].quantile(0.25)
        Q3 = df[col].quantile(0.75)
        IQR = Q3 - Q1
        filter = (df[col] >= Q1 - multiplier * IQR) & (df[col] <= Q3 + multiplier * IQR)
        df = df.loc[filter]
    return df
df_cleaned = remove_outliers(df_cleaned, numeric_columns, multiplier=3.0)
if df_cleaned.empty:
    raise ValueError("All data points were removed as outliers, adjust the outlier filtering criteria.")
# Normalize the data
scaler = StandardScaler()
df_cleaned[numeric_columns] = scaler.fit_transform(df_cleaned[numeric_columns])
# Cleaned dataset to a new CSV file
df_cleaned.to_csv('/content/drive/MyDrive/Colab Notebooks/cleandata/ddos_dns_cleaned.csv', index=False)
print("Data cleaning complete. Cleaned data saved to 'ddos_dns_cleaned.csv'.")
```

Picture 5. Python code used for handling missing values, outlier removal, and normalization

3.3 Modeling

This study employs ensemble machine learning techniques—Random Forest, XGBoost, AdaBoost, and LightGBM—selected for their ability to enhance model accuracy and stability by aggregating multiple base learners. These algorithms are known for strong performance in classification tasks, including DDoS attack detection.

A. Test Case and Model Development

The dataset was split into 70% training and 30% testing subsets. For each feature selection method (RFE, PCA, IG) and the raw dataset, models were trained and evaluated. The process includes:

Data Preparation: Train-test split.

Feature Selection: Applying RFE, PCA, and IG.

Model Training: Using both selected and original features.

Model Testing: Performance evaluation on test data.

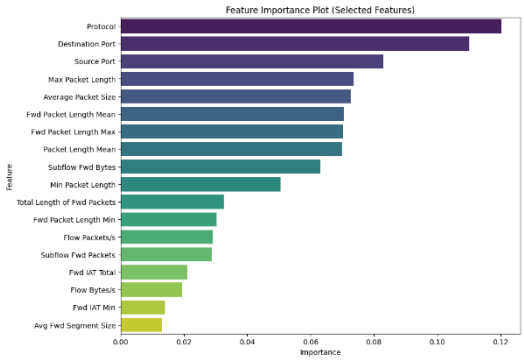
B. Feature Selection Implementation

Feature selection was applied to five DDoS datasets: DNS, MSSQL, LDAP, UDP, and NTP. Methods included Recursive Feature Elimination (RFE), Principal Component Analysis (PCA), and Information Gain (IG), aiming to reduce computational load and improve model efficiency. Each method produced new datasets tailored to the selected features.

Recursive Feature Elimination

The RFE process begins by loading cleaned datasets in chunks. Mutual information is calculated for each feature to assess relevance to the classification target. Scores from all chunks are aggregated, and a threshold is applied to retain only the most informative features. Selected features are visualized using

a horizontal bar chart to support interpretation, as illustrated in Picture 6.S

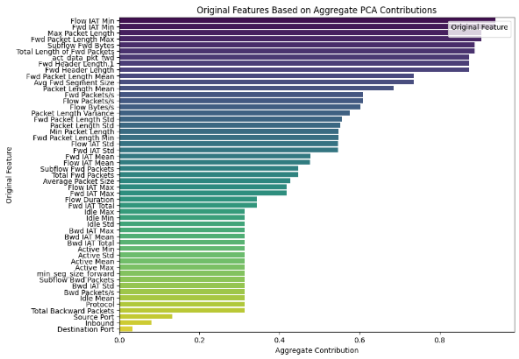


Picture 6. Feature Importance Plot Recursive Feature Elemination

Principal Component Analysis (PCA)

The PCA-based feature selection process begins with dataset loading and preprocessing. A scree plot is used to determine the number of components required to explain at least 90% of the variance. PCA is then re-applied with the selected components, and the contribution of each original feature to the principal components is calculated.

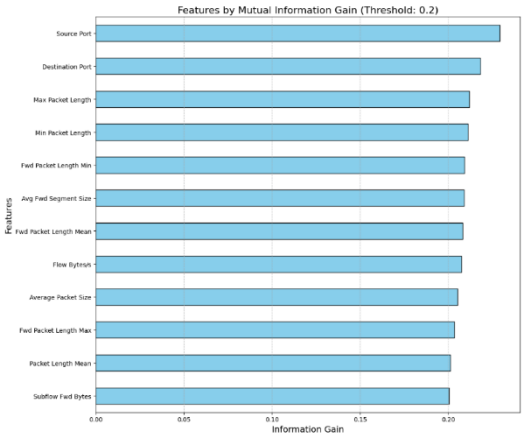
Features are ranked based on their contributions to identify the most important ones. The aggregated contributions are visualized in a bar chart, as shown in Picture 7.



Picture 7. Feature Importance Plot Principal Component Analysis

Information Gain

The feature selection process using Information Gain starts with loading the cleaned dataset in chunks. Mutual information is calculated for each feature to assess its relevance to the target. Scores from all chunks are aggregated, and a threshold is applied to select the most relevant features. These selected features are then visualized using a horizontal bar chart for further analysis, as illustrated in Picture 8.



Picture 8. Feature Importance Plot Information Gain

3.3 Ensemble Machine Learning Algorithm Implementation

Random Forest

Implemented using Scikit-learn, with the following parameters based on Castillo-Olea et al. (2019) [22]:

n_estimators: 10
max_depth: 3
max_features: 'auto'

XGBoost

Implemented using the XGBoost library, following Kurnia D et al. (2019) [23]:

n_estimators: 100
learning_rate: 0.1
max_depth: 6
reg_lambda: 1

AdaBoost

Implemented with Scikit-learn as per Saifulah H et al. (2023) [24]:

n_estimators: 50
learning_rate: 1.0

LGBM

Implemented using LightGBM based on Kim J et al. (2020) [25]:

n_estimators: 100
learning_rate: 0.1
max_depth: -1 (unlimited)
reg_lambda: 0.1

3.4 Model Training with Raw and Processed Datasets

Training with Recursive Feature Elemination

The RFE-processed dataset was trained and tested using ensemble classifiers: Random Forest, LGBM, XGBoost, and AdaBoost. The results are summarized in the following table.

A. Random Forest

Table 3: DDoS Detection Matrix Using Random Forest Classification with RFE

No	Type of DdoS	Accuracy	Precision	Recall	F1 Score	Misclassification Rate	ROC-AUC Score	Training Time (seconds)	Avg CPU usage during training	Avg Memory usage during training (MB)	Avg CPU usage during testing	Avg Memory usage during testing (MB)
1	DNS	0,9997	0,9998	0,9997	0,9997	0,0003	0,9999	8,4480	0,20%	1938,32	0,30%	1805,09
2	LDAP	0,9998	0,9999	0,9998	0,9999	0,0002	1	1,6028	0,10%	1016,72	0,10%	1048,81
3	MSSQL	0,9998	0,9999	0,9998	0,9998	0,0002	1	3,2956	0,10%	1827,41	0,10%	1684,69
4	UDP	0,9995	0,9997	0,9995	0,9995	0,0005	0,9999	3,8557	0,50%	1412,13	0,10%	1437,66
5	NTP	0,9985	0,9987	0,9985	0,9986	0,0015	1	3,8557	0,30%	574,6	0,10%	574,78

B. Light Gradient Boosting Machine

Table 4: DDoS Detection Matrix Using Light Gradient Boosting Machine Classification with RFE

No	Type of DdoS	Accuracy	Precision	Recall	F1 Score	Misclassification Rate	ROC-AUC Score	Training Time (seconds)	Avg CPU usage during training	Avg Memory usage during training (MB)	Avg CPU usage during testing	Avg Memory usage during testing (MB)
1	DNS	0,9999	0,9999	0,9999	0,9999	0,0001	1	3,71	0,20%	1679,04	0,20%	1679,23
2	LDAP	0,9999	0,9999	0,9999	0,9999	0,0001	1	5,66	0,60%	828,58	0,20%	830,44
3	MSSQL	0,9999	0,9999	0,9999	0,9999	0,0001	1	15,97	0,40%	1547,68	0,10%	1530,05
4	UDP	0,9999	0,9999	0,9999	0,9999	0,0001	1	3,91	0,30%	1149,06	0,30%	1149,18
5	NTP	0,9999	0,9999	0,9999	0,9999	0,0001	1	5,12	0,40%	597,22	0,20%	599,98

C. XGBOOST

Table 5: DDoS Detection Matrix Using XGBoost Classification with RFE

No	Type of DdoS	Accuracy	Precision	Recall	F1 Score	Misclassification Rate	ROC-AUC Score	Training Time (seconds)	Avg CPU usage during training	Avg Memory usage during training (MB)	Avg CPU usage during testing	Avg Memory usage during testing (MB)
1	DNS	0,9999	0,9999	0,9999	0,9999	0,0001	1,00	8,26	0,20%	1457,56	0,30%	1438,84
2	LDAP	0,9999	0,9999	0,9999	0,9999	0,0001	1,00	2,95	0,20%	852,64	0,30%	852,85
3	MSSQL	0,9999	0,9999	0,9999	0,9999	0,0001	1,00	9,45	0,50%	852,65	0,30%	852,73
4	UDP	0,9999	0,9999	0,9998	0,9998	0,0002	1,00	7,19	0,30%	1144,94	0,20%	1145,35
5	NTP	0,9998	0,9998	0,9998	0,9998	0,0002	1,00	1,82	0,20%	562,82	0,30%	563,00

D. ADABoost

Table 6: DDoS Detection Matrix Using ADABoost Classification with RFE

No	Type of DdoS	Accuracy	Precision	Recall	F1 Score	Misclassification Rate	ROC-AUC Score	Training Time (seconds)	Avg CPU usage during training	Avg Memory usage during training (MB)	Avg CPU usage during testing	Avg Memory usage during testing (MB)
1	DNS	0,9998	0,9999	0,9998	0,9998	0,0002	0,999	84,01	0,30%	1728,29	0,10%	1720,74
2	LDAP	0,9999	0,9999	0,9999	0,9999	0,0001	1,000	44,18	0,20%	868,44	0,20%	868,56
3	MSSQL	0,9999	0,9999	0,9999	0,9999	0,0001	0,997	95,26	0,30%	1563,5	0,10%	1563,5
4	UDP	0,9997	0,9998	0,9997	0,9997	0,0003	1,000	72,76	0,30%	1170,04	0,20%	1170,09
5	NTP	0,9993	0,9993	0,9993	0,9993	0,0007	1,000	26,62	0,10%	567,09	0,20%	567,09

Model Training with PCA

A. Random Forest

Table 7: DDoS Detection Matrix Using Random Forest Classification with PCA

No	Type of DdoS	Accuracy	Precision	Recall	F1 Score	Misclassification Rate	ROC-AUC Score	Training Time (seconds)	Avg CPU usage during training	Avg Memory usage during training (MB)	Avg CPU usage during testing	Avg Memory usage during testing (MB)
1	DNS	0,9951	0,9994	0,9951	0,997	0,0049	0,9943	4,4973	0,10%	2004,15	0,20%	1857,98
2	LDAP	0,9991	0,9996	0,9991	0,9992	0,0009	1	2,6617	0,20%	1025,35	0,10%	1055,46
3	MSSQL	0,9989	0,9997	0,9989	0,9992	0,0011	0,9985	4,9557	0,20%	1855,72	0,20%	1727,24
4	UDP	0,9991	0,9996	0,9991	0,9993	0,0009	0,9999	4,1061	0,10%	1365,99	0,10%	1383,14
5	NTP	0,9462	0,9894	0,9462	0,964	0,0538	0,9859	1,2723	0,10%	645,17	0,20%	650,83

B. Light Gradient Boosting Machine

Table 8: DDoS Detection Matrix Using Light Gradient Boosting Machine Classification with PCA

No	Type of DdoS	Accuracy	Precision	Recall	F1 Score	Misclassification Rate	ROC-AUC Score	Training Time (seconds)	Avg CPU usage during training	Avg Memory usage during training (MB)	Avg CPU usage during testing	Avg Memory usage during testing (MB)
1	DNS	0,9947	0,9994	0,9947	0,9968	0,0053	0,99	10,8	0,30%	1723,77	0,10%	1725,91
2	LDAP	0,9999	0,9999	0,9999	0,9999	0,0001	1,00	4,2	0,50%	891,43	0,20%	913,25
3	MSSQL	0,9999	0,9999	0,9999	0,9999	0,0001	0,99	8,1	0,50%	1571,79	0,10%	1572,68
4	UDP	0,9999	0,9999	0,9999	0,9999	0,0001	1,00	8,9	0,40%	1129,13	0,10%	1129,23
5	NTP	0,9427	0,9894	0,9427	0,9620	0,0573	0,99	3,3	0,40%	570,77	0,30%	575,80

C. XGBOOST

Table 9: DDoS Detection Matrix Using XGBoost Classification with PCA

No	Type of DdoS	Accuracy	Precision	Recall	F1 Score	Misclassification Rate	ROC-AUC Score	Training Time (seconds)	Avg CPU usage during training	Avg Memory usage during training (MB)	Avg CPU usage during testing	Avg Memory usage during testing (MB)
1	DNS	0,9944	0,9993	0,9944	0,9966	0,0056	0,99	10,79	0,20%	1423,32	0,30%	1402,32
2	LDAP	0,9999	0,9999	0,9999	0,9999	0,0001	1,00	6,41	0,00%	1107,39	0,10%	1110,59
3	MSSQL	0,9999	0,9999	0,9999	0,9999	0,0001	1,00	13,80	0,00%	1691,42	0,30%	1683,52
4	UDP	0,9998	0,9998	0,9998	0,9998	0,0002	1,00	13,41	0,20%	1121,46	0,00%	1102,95
5	NTP	0,944	0,9894	0,944	0,9628	0,056	0,99	17,80	0,10%	561,24	0,10%	561,35

D. ADABoost

Table 10: DDoS Detection Matrix Using ADABoost Classification with PCA

No	Type of DdoS	Accuracy	Precision	Recall	F1 Score	Misclassification Rate	ROC-AUC Score	Training Time (seconds)	Avg CPU usage during training	Avg Memory usage during training (MB)	Avg CPU usage during testing	Avg Memory usage during testing (MB)
1	DNS	0,9925	0,9993	0,9925	0,9956	0,0075	0,993	98,47	0,30%	1729,35	0,30%	1729,35
2	LDAP	0,9999	0,9999	0,9999	0,9999	0,0001	1,000	40,63	0,30%	878,92	0,20%	878,92
3	MSSQL	0,9997	0,9998	0,9997	0,9998	0,0003	0,997	101,16	0,20%	1575,09	0,30%	1575,09
4	UDP	0,9994	0,9997	0,9994	0,9995	0,0006	1,000	80,74	0,10%	1145,93	0,20%	1145,93
5	NTP	0,9372	0,9892	0,9327	0,9563	0,0673	0,989	25,84	0,50%	558,4	0,30%	558,4

Model Training with Information Gain

A. Random Forest

Table 11: DDoS Detection Matrix Using Random Forest Classification with Information Gain

No	Type of DdoS	Accuracy	Precision	Recall	F1 Score	Misclassification Rate	ROC-AUC Score	Training Time (seconds)	Avg CPU usage during training	Avg Memory usage during training (MB)	Avg CPU usage during testing	Avg Memory usage during testing (MB)
1	DNS	0,9997	0,9998	0,9997	0,9997	0,0003	0,9999	6,3	0,20%	2020,08	0,20%	1863,43
2	LDAP	0,9998	0,9999	0,9998	0,9998	0,0002	0,9999	2,1	0,50%	1030,45	0,20%	1052,20
3	MSSQL	0,9998	0,9999	0,9998	0,9998	0,0002	0,9999	5,0	0,40%	1902,15	0,10%	1680,79
4	UDP	0,9993	0,9996	0,9993	0,9994	0,0007	0,9998	4,5	0,20%	1386,15	0,20%	1424,76
5	NTP	0,9970	0,9976	0,9970	0,9971	0,0030	0,9998	0,9	0,10%	646,03	0,30%	660,57

B. Light Gradient Boosting Machine

Table 12: DDoS Detection Matrix Using Light Gradient Boosting Machine Classification with Information Gain

No	Type of DdoS	Accuracy	Precision	Recall	F1 Score	Misclassification Rate	ROC-AUC Score	Training Time (seconds)	Avg CPU usage during training	Avg Memory usage during training (MB)	Avg CPU usage during testing	Avg Memory usage during testing (MB)
1	DNS	0,9999	0,9999	0,9999	0,9999	0,0001	1	13,45	0,30%	1711,44	0,10%	1711,52
2	LDAP	0,9998	0,9998	0,9998	0,9998	0,0002	1	5,05	0,30%	833,11	0,10%	826,73
3	MSSQL	0,9999	0,9999	0,9999	0,9999	0,0001	1	8,58	0,40%	1567,12	0,10%	1568,00
4	UDP	0,9999	0,9999	0,9999	0,9999	0,0001	1	3,86	0,30%	1144,03	0,10%	1144,12
5	NTP	0,9985	0,9987	0,9985	0,9986	0,0015	0,9999	25,36	0,40%	550,11	0,10%	540,20

C. XGBOOST

Table 13: DDoS Detection Matrix Using XGBoost Classification with Information Gain

No	Type of DdoS	Accuracy	Precision	Recall	F1 Score	Misclassification Rate	ROC-AUC Score	Training Time (seconds)	Avg CPU usage during training	Avg Memory usage during training (MB)	Avg CPU usage during testing	Avg Memory usage during testing (MB)
1	DNS	0,9999	0,9999	0,9999	0,9999	0,0001	1,00	21,04	0,10%	1668,89	0,10%	1671,57
2	LDAP	0,9998	0,9999	0,9998	0,9998	0,0002	1,00	6,32	0,10%	1220,64	0,30%	1220,95
3	MSSQL	0,9999	0,9999	0,9999	0,9999	0,0001	1,00	12,57	0,10%	1497,45	0,20%	1476,95
4	UDP	0,9998	0,9998	0,9998	0,9998	0,0002	1,00	22,40	0,00%	1517,80	0,10%	1530,65
5	NTP	0,9985	0,9987	0,9985	0,9986	0,0015	1,00	17,82	0,30%	701,14	0,10%	703,10

D. ADABoost

Table 14: DDoS Detection Matrix Using ADABoost Classification with Information Gain

No	Type of DdoS	Accuracy	Precision	Recall	F1 Score	Misclassification Rate	ROC-AUC Score	Training Time (seconds)	Avg CPU usage during training	Avg Memory usage during training (MB)	Avg CPU usage during testing	Avg Memory usage during testing (MB)
1	DNS	0,9998	0,9998	0,9998	0,9998	0,0002	1,00	117,19	0,20%	1722,06	0,20%	1716,23
2	LDAP	0,9998	0,9999	0,9998	0,9999	0,0002	1,00	37,30	0,20%	865,10	0,10%	865,24
3	MSSQL	0,9998	0,9998	0,9998	0,9998	0,0002	1,00	110,85	0,20%	1567,51	0,30%	1567,51
4	UDP	0,9995	0,9997	0,9995	0,9996	0,0005	1,00	93,08	0,10%	1129,35	0,20%	1125,18
5	NTP	0,9979	0,9982	0,9979	0,998	0,0021	1,00	24,38	0,40%	558,25	0,20%	558,25

Model Training on Raw Dataset

A. Random Forest

Table 15: DDoS Detection Matrix Using Random Forest Classification with Raw Dataset

No	JENIS DDOS	Accuracy	Precision	Recall	F1 Score	Misclassification Rate	ROC-AUC Score	Training Time (seconds)	Avg CPU usage during training	Avg Memory usage during training (MB)	Avg CPU usage during testing	Avg Memory usage during testing (MB)
1	DNS	0,9997	0,9998	0,9997	0,9997	0,0003	0,9999	22,11	1,20%	16863,78	0,50%	16868,96
2	LDAP	0,9999	0,9999	0,9999	0,9999	0,0001	1	0,89	1,00%	8846,01	0,40%	8846,38
3	MSSQL	0,9999	0,9999	0,9999	0,9999	0,0001	1	2,03	0,90%	11796,32	0,50%	11792,75
4	UDP	0,9996	0,9997	0,9996	0,9997	0,0004	0,9765	1,43	1,20%	10330,49	0,40%	10330,33
5	NTP	0,9967	0,9974	0,9967	0,9969	0,0033	0,9999	0,6444	1,20%	7969,71	0,50%	7969,26

B. Light Gradient Boosting Machine

Table 16: DDoS Detection Matrix Using Light Gradient Boosting Machine Classification with Raw Dataset

No	JENIS DDOS	Accuracy	Precision	Recall	F1 Score	Misclassification Rate	ROC-AUC Score	Training Time (seconds)	Avg CPU usage during training	Avg Memory usage during training (MB)	Avg CPU usage during testing	Avg Memory usage during testing (MB)
1	DNS	0,9778	0,9986	0,9778	0,9881	0,0222	0,2135	16,47	72,70%	19533,12	5,29%	19543,92
2	LDAP	0,9821	0,9985	0,9821	0,9902	0,0179	0,0932	10,12	73,61%	14619,93	6,99%	14619,93
3	MSSQL	0,9809	0,9991	0,9809	0,9899	0,0191	0,0148	15,48	73,33%	17601,87	5,59%	17603,73
4	UDP	0,9918	0,9986	0,9918	0,9952	0,0082	0,0623	12,17	72,10%	15710,93	5,59%	15711,98
5	NTP	0,9880	0,9882	0,9880	0,9825	0,0120	0,6106	8,33	73,70%	11782,53	7,09%	11784,38

C. XGBOOST

Table 17: DDoS Detection Matrix Using XGBoost Classification with Raw Dataset

No	JENIS DDOS	Accuracy	Precision	Recall	F1 Score	Misclassification Rate	ROC-AUC Score	Training Time (seconds)	Avg CPU usage during training	Avg Memory usage during training (MB)	Avg CPU usage during testing	Avg Memory usage during testing (MB)
1	DNS	0,9921	0,9986	0,9921	0,9953	0,0079	0,1272	32,52	141,41%	11918,26	15,38%	11919,15
2	LDAP	0,9941	0,9985	0,9941	0,9963	0,0059	0,0859	15,21	143,91%	7887,83	15,78%	7887,39
3	MSSQL	0,9952	0,9991	0,9952	0,9971	0,0048	0,1026	22,19	144,50%	11210,88	15,48%	11210,88
4	UDP	0,9974	0.9986	0.9974	0.9980	0.0026	0.2329	19,02	140,60%	9290,65	15,88%	9290,65
5	NTP	0,9858	0.9768	0.9858	0.9812	0.0142	0.3079	18,41	144,22%	6497,39	16,18%	6497.39

D. ADABoost

Table 18: DDoS Detection Matrix Using ADABoost Classification with Raw Dataset

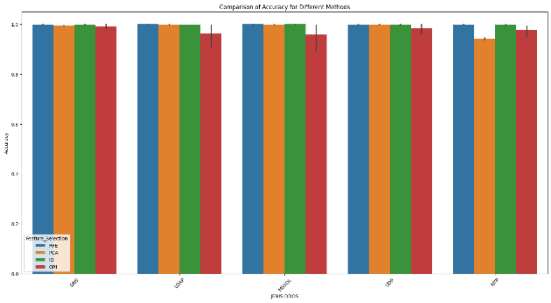
No	JENIS DDOS	Accuracy	Precision	Recall	F1 Score	Misclassification Rate	ROC-AUC Score	Training Time (seconds)	Avg CPU usage during training	Avg Memory usage during training (MB)	Avg CPU usage during testing	Avg Memory usage during testing (MB)
1	DNS	0,9999	0,9999	0,9999	0,9999	0,0001	1,00	3.048,02	0,30%	11043,87	0,40%	11044,18
2	LDAP	0,8803	0,9985	0,8803	0,9357	0,1197	0,4879	565,64	0,50%	7495,96	0,30%	7496,04
3	MSSQL	0,8565	0,9991	0,8565	0,9223	0,1435	0,4881	1.886,43	0,40%	10995,48	0,40%	10995,48
4	UDP	0,9496	0,9986	0,9496	0,9735	0,0504	0,4892	1.533,98	0,20%	9314,25	0,30%	9314,25
5	NTP	0,941	0,9767	0,941	0,9584	0,059	0,5025	687,77	0,50%	6910,29	0,20%	6910,62

3.3 Evaluation

Model Accuracy Evaluation on Dataset

The evaluation results show that the models are capable of detecting DDoS attacks with high accuracy. A summary of the evaluation based on the defined metrics is presented in Picture 9.

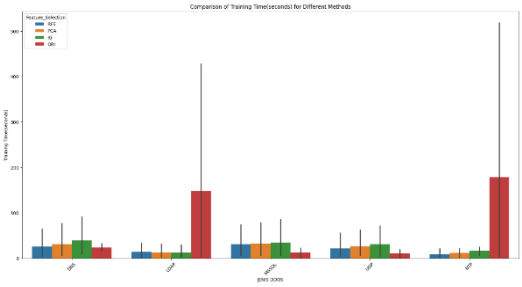
Picture 9. Comparison of DDoS Detection



Accuracy between Feature Selection and Raw Datasets

Evaluation of Model Training Time on Dataset

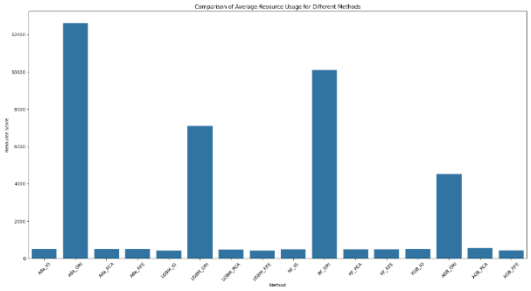
The evaluation results show variations in training time across different datasets. A visual summary is presented in Picture 10.



Picture 10. Comparison of Training Time between Feature-Selected and Raw Datasets, based on the defined metrics

Evaluation of Computational Resource Usage During the Training and Testing Process

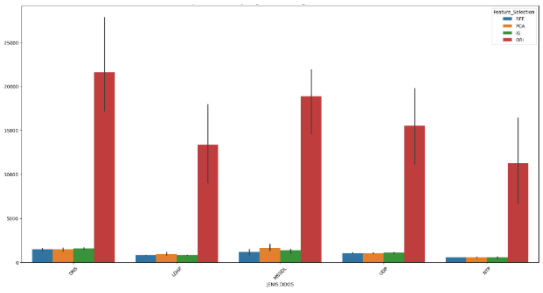
The evaluation results of computational resource usage during the training and testing processes using various feature selection methods and raw datasets processed with classification are presented. The comparison of RAM usage required for training the Feature Selection dataset and the raw dataset is visually illustrated in Picture 11. The evaluation results are as follows:



Picture 11. Comparison Of Ram Usage Required For Training Using Feature-Selection Datasets And Raw Datasets

Evaluation of RAM Usage During the Model Training Process on the Dataset

The evaluation results of RAM usage during the model training process on the dataset indicate variations in the amount of RAM resources required by the model for training on each dataset. The following is a summary of the evaluation results based on predefined metrics, as illustrated in Picture 12, which presents a comparison of RAM usage required for training using feature-selection datasets and raw datasets:



Picture 12. Comparison Of Ram Usage Required For Training Using Feature-Selection Datasets And Raw Datasets

4. CONCLUSION

This study found that feature selection methods can significantly improve model performance in detecting DDoS attacks. The combination of Recursive Feature Elimination (RFE) and Light Gradient Boosting Machine (LGBM) demonstrated a strong balance between accuracy and computational efficiency compared to other methods, as presented in Table 19, which ranks the best Feature Selection and Classification methods.

Table 19: Ranking the Best Feature Selection

Rank	Method	Metric Score	Resource Score	Overall Score
1	LGBM_RFE	0.833287	411.254.444	206.043.866
2	LGBM_IG	0.833140	418.261.511	209.547.326
3	XGB_RFE	0.833270	425.381.044	213.107.157

The results of this research have significant practical implications for cybersecurity systems. Implementing feature selection methods such as RFE, PCA, and IG can reduce the need for computational resources, enabling real-time DDoS attack detection at lower costs. This is especially critical for organizations that need to process large volumes of data quickly and efficiently. Additionally, the findings suggest that by selecting only relevant features, systems can become more responsive and reliable in dealing with cyber threats.

Recommendations for Future Research

Based on the findings of this study, the following recommendations are proposed for future research:

- Evaluate other feature selection methods and their combinations to determine whether further improvements in model performance can be achieved.
- Develop and implement more advanced machine learning algorithms or hybrid models to enhance detection accuracy.
- Apply this research to other types of cyberattacks to assess the effectiveness of feature selection methods in different contexts.
- Explore the use of real-time and streaming data to test model performance under more dynamic and realistic conditions

REFERENCES

[1] O. Ussatova, A. Zhumbekova, Y. Begimbayeva, E. Matson, and N. Ussatov, "Comprehensive DDoS attack classification using machine learning algorithms," Computers, Materials & Continua, vol. 73, no. 1, pp. –, 2022.

[2] I. Cvitić, "An overview of distributed denial of service traffic detection approaches," Promet – Traffic & Transportation, vol. 31, no. 4, pp. 453–464, 2019.

[3] K. Sahoo, B. Tripathy, K. Naik, S. Ramasubbareddy, B. Balusamy, M. Khari, and D. Burgos, "An evolutionary SVM model for DDoS attack detection in software defined networks," *IEEE Access*, vol. 8, pp. 13502–132513, 2020.

[4] A. Aljuhani, "Machine learning approaches for combating distributed denial of service attacks in modern networking environments," *IEEE Access*, vol. 9, pp. –, 2021.

[5] A. J. Perez-Diaz, A. I. Valdovinos, R. K. Choo, and D. Zhu, "A flexible SDN-based architecture for identifying and mitigating low-rate DDoS attacks using machine learning," IEEE Access, vol. 8, pp. 155859–155872, 2020.

[6] K. Dasari and N. Devarakonda, "Detection of TCP-based DDoS attacks with SVM classification with different kernel functions using common uncorrelated feature subsets," Int. J. Safety and Security Eng., vol. 12, no. 2, pp. 239–249, 2022.

[7] S. Dheyab, S. Abdulameer, and S. Mostafa, "Efficient machine learning model for DDoS detection system based on dimensionality reduction," Acta Informatica Pragensia, vol. 11, no. 3, pp. 348–360, 2022.

[8] M. Marvi, A. Arfeen, and R. Uddin, "A generalized machine learning-based model for the detection of DDoS attacks," Int. J. Network Management, vol. 31, no. 6, 2021.

[9] O. E. Omuya, O. G. Okeyo, and W. M. Kimwele, "Feature selection for classification using principal component analysis and information gain," Expert Syst. Appl., vol. 174, 2021.

[10] D. Upadhyay, J. Manero, M. Zaman, and S. Sampalli, "Intrusion detection in SCADA based power grids: recursive feature elimination model with majority vote ensemble algorithm," IEEE Trans. Netw. Sci. Eng., vol. 8, no. 3, pp. 2559–2574, 2021.

[11] Z. Ahmad, A. Khan, C. W. Shiang, J. Abdullah, and F. Ahmad, "Network intrusion detection system: a systematic study of machine learning and deep learning approaches," Trans. Emerg. Telecommun. Technol., vol. 32, no. 1, 2021.

[12] I. Ismail, M. Mohmand, H. Hussain, A. Khan, U. Ullah, M. Zakarya, A. Ahmed, M. Raza, I. Rahman, and M. Haleem, "A machine learning-based classification and prediction technique for DDoS attacks," IEEE Access, vol. 10, pp. 21443–21454, 2022.
[13] J. Bhayo, S. Shah, S. Hameed, A. Ahmed, J. Nasir, and D. Draheim, "Towards a machine learning-based framework for DDoS attack detection in software-defined IoT (SD-IoT) networks," Eng. Appl. Artif. Intell., vol. 123, 2023.
[14] M. Aslam, D. Ye, A. Tariq, M. Asad, M. Hanif, D. Ndzi, S. Chelloug, M. Elaziz, M. Al-Qaness, and S. Jilani, "Adaptive machine learning based distributed denial-of-services attacks detection and mitigation system for SDN-enabled IoT," Sensors, vol. 22, no. 7, 2022.
[15] M. Alshahrani, "A secure and intelligent software-defined networking framework for future smart cities to prevent DDoS attack," Appl. Sci., vol. 13, no. 17, 2023.
[16] H. Polat, O. Polat, and A. Cetin, "Detecting DDoS attacks in software-defined networks through feature selection methods and machine learning models," Sustainability, vol. 12, no. 3, 2020.
[17] A. Azhari, A. Muhammad, and C. Foozy, "Machine learning-based distributed denial of service attack detection on intrusion detection system regarding to feature selection," Int. J. Artif. Intell. Res., vol. 4, no. 1, 2020.
[18] C. Bagyalakshmi and S. Samundeeswari, "DDoS attack classification on cloud environment using machine learning techniques with different feature selection methods," Int. J. Adv. Trends Comput. Sci. Eng., vol. 9, no. 5, pp. 7301–7308, 2020.
[19] A. Barkah, S. Selamat, Z. Abidin, and R. Wahyudi, "Impact of data balancing and feature selection on machine learning-based network intrusion detection," Int. J. Informatics Visualization, vol. 7, no. 1, 2023.
[20] M. Gupta and P. Chandra, "A comprehensive survey of data mining," Int. J. Inf. Technol., vol. 12, no. 4, pp. 1243–1257, 2020.
[21] J. Bian and S. Fu, "Application of data mining in predictive analysis of network security model," Security Commun. Netw., vol. 2022, no. 8, 2022.
[22] C. Castillo-Olea, B. Soto, C. Lozano, and C. Zuñiga, "Automatic classification of sarcopenia level in older adults: a case study at Tijuana General Hospital," Int. J. Environ. Res. Public Health, vol. 16, no. 18, 2019.
[23] D. Kurnia, M. Mazdadi, D. Kartini, R. Nugroho, and F. Abadi, "Seleksi fitur dengan particle swarm optimization pada klasifikasi penyakit Parkinson menggunakan XGBoost," J. Teknol. Inf. dan Ilmu Komput., vol. 10, no. 5, pp. 1083–1094, 2023.
[24] H. Saifullah, N. Adiwijaya, and T. Dharmawan, "Optimization of machine learning algorithms with bagging and AdaBoost methods for stroke disease prediction," Appl. Med. Inform., vol. 45, no. 2, pp. 70–81, 2023.
[25] J. Kim, H. Lee, and J. Oh, "Study on prediction of ship's power using light GBM and XGBoost," J. Adv. Mar. Eng. Technol., vol. 44, no. 2, pp. 174–180.