Matrix: Jurnal Manajemen Teknologi dan Informatika
Volume 16 Issue 1 Year 2026 Pages 49-61
DOI: http://dx.doi.org/10.31940/matrix.v16i1.49-61
URL: http://ojs2.pnb.ac.id/index.php/matrix

Model creation for Denial of Service (DoS) attack classification using an ensemble learning approach on multi-dataset network traffic

Farhan Ainurrahman [1]*, Hariz Farisi [2], Diva Kurnianingtyas [3]

[1,2] *Information Technology Study Program, Universitas Brawijaya, Indonesia*
[3] *Informatics Engineering Study Program, Universitas Brawijaya, Indonesia*

*Corresponding Author: *farhanainurrahman2147@gmail.com*

Abstract: The rapid advancement of information technology has increased cybersecurity threats, one of which is the Denial of Service (DoS) attack that can disrupt service availability. Most existing studies on DoS attack classification rely on a single dataset and a single machine learning model, which limits the generalizability of their results across different network environments. This study addresses this gap by proposing an ensemble learning-based model for DoS attack classification using multi-dataset network traffic. The datasets used in this research are UNSW-NB15 and TON-IoT, which were combined based on feature compatibility. After the preprocessing stage, a final dataset consisting of 73,302 records was obtained, comprising 64,267 normal traffic instances and 9,035 DoS attack instances. The dataset was then split using stratified sampling with an 80:20 ratio for training and testing data. The ensemble learning methods applied include Random Forest (bagging) and XGBoost (boosting), with training scenarios using both the original dataset and data balanced using the Synthetic Minority Over-sampling Technique (SMOTE). Model evaluation was conducted using a confusion matrix and performance metrics including accuracy, precision, recall, F1-score, and ROC-AUC.The results show that the ensemble learning approach achieves high performance in classifying DoS attacks. However, the application of SMOTE did not improve model performance in this study. The best-performing model was Random Forest trained on the original dataset, achieving an accuracy of 0.9854, precision of 0.9515, recall of 0.928, F1-score of 0.9402, and ROC-AUC of 0.996. These results indicate that the proposed model is effective for DoS attack classification across heterogeneous network traffic data.

Keywords: Attack classification, Denial of Service (DoS) ensemble learning, Random Forest, XGBoost

History Article: Submitted 22 January 2026 | Revised 6 February 2026 | Accepted 16 February 2026

How to Cite: F. Ainurrahman, H. Farisi, and D. Kurnianingtyas, "Model creation for Denial of Service (DoS) attack classification using an ensemble learning approach on multi-dataset network traffic," *Matrix: Jurnal Manajemen Teknologi dan Informatika*, vol. 16, no. 1, pp. 49–61, 2026, doi: 10.31940/matrix.v16i1.49-61.

Introduction

The rapid development of information technology has facilitated data management and processing across various sectors; however, it has also increased the complexity of cybersecurity threats. Indonesia's National Cyber and Crypto Agency (BSSN) reported 3.64 billion cyber attacks or network traffic anomalies between January and July 2025, highlighting the urgent need for comprehensive network protection. Network anomalies refer to activities that deviate from normal traffic patterns, such as abnormal packet spikes, unusual port usage, or irregular communication behavior, making early detection a critical component of network security systems.

One of the most common threats is Denial of Service (DoS) attacks, which aim to overwhelm systems with excessive requests, rendering services inaccessible to legitimate users. According to the Common Vulnerability Scoring System (CVSS), DoS attacks are categorized as high-severity threats because they directly target service availability. The sophistication of these attacks continues to grow, as demonstrated by the mitigation of a 3.47 Tb/s Distributed Denial of Service (DDoS) attack with a packet rate of up to 340 million packets per second handled by Microsoft Azure in 2021 [1]. Such attacks not only degrade system performance but also cause financial losses and reputational damage for organizations.

Various automated detection approaches have been developed, one of which is the application of machine learning techniques. Ensemble learning has emerged as a widely adopted method due to its ability to improve prediction stability and performance by combining multiple base models [2]. This concept is based on the limitation of single models, which may not perform optimally for all problem types, while model combinations can compensate for individual weaknesses and produce more reliable predictions [3].

Previous studies have demonstrated the effectiveness of machine learning techniques for detecting DoS attacks. Primadya *et al.* applied Logistic Regression on the CIC IoT Attacks 2023 dataset and achieved an accuracy of 97% using random undersampling and Recursive Feature Elimination (RFE) [4]. Harto and Basuki employed Random Forest for DDoS detection in Software Defined Networking (SDN) environments and obtained 90% accuracy with a detection time of 0.3 seconds [5]. Ariyanto *et al.* developed a K-Nearest Neighbor-based intrusion detection system using the KDDCUP 1999 dataset and reported an accuracy of 90% [6]. Meanwhile, Firdaus *et al.* evaluated low-rate DDoS detection using Naive Bayes with the CICIDS2017 dataset and achieved an accuracy of 83.45% [7]. Despite these promising results, most existing studies are still limited to single algorithms or single datasets, which may restrict model generalization. as such approaches do not fully capture the diversity and complexity of real-world network traffic and are rarely evaluated under heterogeneous or imbalanced data conditions.

The choice of datasets also plays a crucial role in building generalized detection models. The UNSW-NB15 dataset provides a modern representation of network traffic with diverse attack scenarios, while the TON-IoT dataset introduces additional variations from Industrial IoT environments, which are inherently more complex [8]. Combining these datasets enables the model to learn more comprehensive and realistic attack patterns, thereby improving the robustness and generalizability of the proposed detection model across different network environments. Therefore, this study explicitly addresses the limitations of prior work by proposing an ensemble learning-based DoS attack classification model trained on integrated multi-dataset network traffic and evaluated under both original and balanced data distributions.

Based on these considerations, this study proposes a DoS attack classification model using an ensemble learning approach with the UNSW-NB15 and TON-IoT datasets. The model aims to accurately classify network anomalies and enhance detection reliability. Model performance is evaluated using accuracy, precision, recall, F1-score, and ROC-AUC metrics derived from the confusion matrix. This research is expected to contribute to the development of more robust intrusion detection systems and support improved network security in the digital era.

Methodology

This study aims to develop a classification model for Denial of Service (DoS) attacks using an ensemble learning approach to classify normal traffic and DoS attack traffic. The model is built using two datasets, namely UNSW-NB15 and TON-IoT. The results of this study are expected to contribute to the development of DoS attack classification models. The overall research workflow is illustrated in Figure 1.

The research procedure consists of data collection, data preprocessing, modeling, and model evaluation. The data used in this study were obtained from two secondary datasets, namely UNSW-NB15 and TON-IoT, with a focus on DoS-labeled and normal traffic data as the basis for model training.

The data preprocessing stage was conducted to ensure that the data were suitable for modeling. This stage included data cleaning to remove irrelevant or inconsistent values, feature selection to identify relevant variables, and data splitting into training and testing subsets. In addition, class imbalance was handled using the Synthetic Minority Over-sampling Technique (SMOTE), and data transformation was applied to prepare the data for modeling.

Modeling was performed using an ensemble learning approach by combining base learners through bagging and boosting techniques. Random Forest was selected as the bagging-based model because previous studies have shown that it outperforms other algorithms such as Support Vector Machine (SVM) in classifying network anomalies in industrial traffic, including systems based on Modbus and OPC UA protocols [9]. Meanwhile, XGBoost was selected as the boosting-based model due to its proven effectiveness in distinguishing between normal and attack traffic

in network environments, particularly based on Simple Network Management Protocol (SNMP) data [10].

The evaluation stage was then carried out using testing data by applying performance metrics including accuracy, precision, recall, F1-score, and AUC-ROC. The model evaluation was used to assess how well the trained model performed in classifying network traffic.

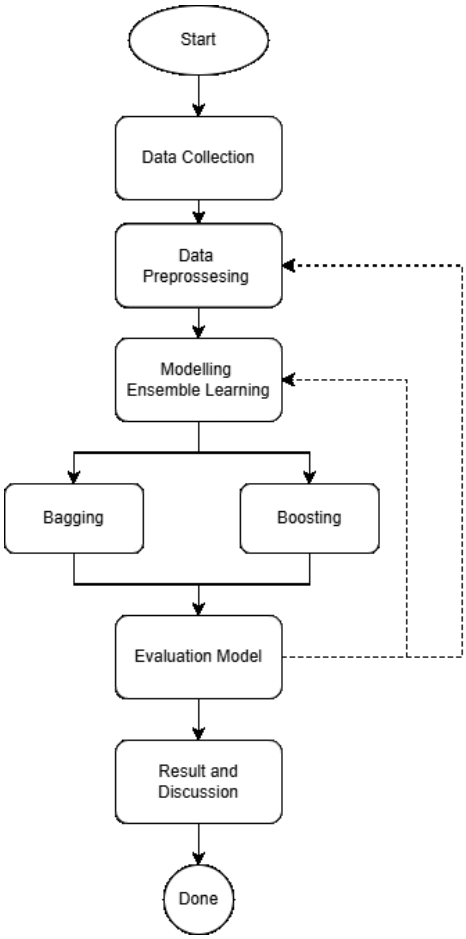


Figure 1. Research workflow

Data Collection and Preprocessing

Data Collection

This study uses two datasets, namely UNSW-NB15 and TON-IoT, both of which were developed by the University of New South Wales (UNSW), Australia. Both datasets contain normal traffic data and Denial of Service (DoS) attack data for model training.

The UNSW-NB15 dataset was generated using IXIA PerfectStorm in the ACCS UNSW Cyber Range Lab. The dataset is available in several formats, and this study uses the CSV format from the provided training and testing sets. Meanwhile, the TON-IoT dataset was developed at the UNSW Canberra IoT Lab and contains data collected from IoT/IIoT sensors. This study utilizes the network traffic subset in CSV format. The multi-source data collection process is illustrated in Figure 2.

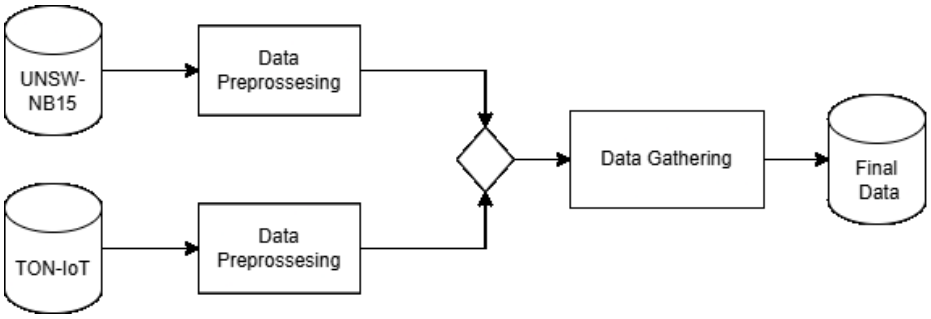


Figure 2. Multi-source data collection

This study employs a multi-source data collection approach using two different datasets, namely UNSW-NB15 and TON-IoT. Both datasets were collected independently, then underwent preprocessing and were integrated through a data integration process to generate the final dataset. The focus of this research is on normal traffic and DoS attack traffic. The UNSW-NB15 dataset provides 56,000 normal traffic instances and 12,264 DoS attack instances, while the TON-IoT dataset contains 50,000 normal traffic instances and 20,000 DoS attack instances. Overall, this study utilizes 106,000 normal traffic records and 32,264 DoS attack records.

The UNSW-NB15 dataset consists of 49 features representing various network traffic characteristics. Meanwhile, the TON-IoT dataset contains 46 features describing network traffic behavior. Features that are representative of DoS attack patterns and normal traffic were selected to optimize the performance of the developed model.

Data Cleaning

The data cleaning stage was conducted to ensure data quality prior to modeling by removing missing values, duplicate records, and invalid data from the UNSW-NB15 and TON-IoT datasets. Missing values in small proportions were removed, while significant missing values were imputed using the median or mean. Duplicate data were eliminated to prevent bias, and illogical values such as negative durations or irrelevant byte values were removed as noise. This process ensured that the dataset used was consistent, valid, and ready for the modeling stage.

Feature Selection

The feature selection stage was conducted to select relevant features from the UNSW-NB15 and TON-IoT datasets so that they could be integrated and analyzed as training and testing data for model development. Features with similar semantic meanings were selected to form a uniform data structure. This approach reduces model complexity, improves computational efficiency, and preserves essential information for detecting DoS attack patterns. As shown in Table 1, the selected features represent the most relevant attributes for the classification task.

Tabel 1. List of selected features

No	Feature Name	Definition
1	dur; duration	Total duration of activity
2	sbytes; src_bytes	Number of bytes transmitted from source to destination
3	dbytes; dst_bytes	Number of bytes transmitted from destination
4	spkts; src_pkts	Number of packets sent from source to destination
5	dpkts; dst_pkts	Number of packets sent from destination to source
6	attack_cat; type	Attack category

Data Gathering

At this stage, feature names from the UNSW-NB15 dataset were renamed to match features with similar meanings in the TON-IoT dataset. he mapping of the original and renamed feature names is presented in Table 2. After the renaming process, both datasets were merged into a single dataset for the analysis of normal traffic and DoS attacks. This integrated dataset was then used for model training in the modeling stage.

Table 2. Renamed features (UNSW-NB15)

No	Original Feature	Renamed Feature
1	dur	duration
2	sbytes	src_bytes
3	dbytes	dst_bytes
4	spkts	src_pkts
5	dpkts	dst_pkts
6	attack_cat	type

The initial integrated dataset consisted of 88,194 records. After data quality verification, 14,892 duplicate records were identified and removed. The final clean dataset contained 64,267 normal traffic records and 9,035 DoS attack records, which were subsequently used for model development.

Data Splitting

At this stage, the dataset was split into training and testing data with an 80:20 ratio using the train_test_split() function from the scikit-learn library. The stratify parameter was applied to maintain the class distribution of normal and DoS traffic in both subsets. This process resulted in 58,641 training samples and 14,661 testing samples, consistent with the original class distribution.

SMOTE was applied only to the training data (80% of the total dataset) to prevent data leakage, with an initial composition of 51,413 normal traffic samples and 7,228 DoS attack samples. SMOTE generates synthetic samples by identifying k-nearest neighbors within the minority class using Euclidean distance.

$$d(x_i, x_j) = \sqrt{\sum_{t=1}^{n} (x_{it} - x_{jt})^2} \quad (1)$$

The new synthetic data were generated using the following formula:

$$x_{new} = x_i + \delta(x_{zi} - x_i) \quad (2)$$

where δ is a random number between 0 and 1. As a result, a total of 43,685 synthetic samples were generated, increasing the number of DoS class samples to 51,413. Consequently, the training dataset became balanced between the normal and DoS classes.

SMOTE for Data Imbalanced

The dataset in this study exhibits an imbalanced class distribution between normal traffic and DoS attacks; therefore, handling this issue is necessary to prevent model bias toward the majority class. The Synthetic Minority Over-sampling Technique (SMOTE) was applied due to its effectiveness in increasing minority class representation and reducing the risk of overfitting [11].

Data Transformation

The final dataset consists of six features with data types as shown in Table 3.

Tabel 3. Feature data types

No	Feature	Data Type
1	Duration	*Float*
2	src_bytes	*Integer*
3	dst_bytes	*Integer*
4	src_pkts	*Integer*
5	dst_pkts	*Integer*
6	Type	*Object*

Five features are numeric, while one feature, namely *Type*, is of object type. Since machine learning models can only process numerical data, the *Type* feature was converted using label

encoding. This process transformed the *normal* category into the value 1 and the *DoS* category into the value 0, following the alphabetical order applied by the label encoder.

Modelling and Evaluation

In the modelling stage, the training data that had undergone preprocessing were used to train the classification models. The trained models were then evaluated using the testing data, which were separated from the training data to prevent overfitting and to ensure an objective assessment of model performance.

In ensemble learning, a base learner refers to the fundamental model that is combined with other models to improve prediction accuracy. In this study, the Decision Tree algorithm was selected as the base learner because it is capable of effectively capturing patterns based on important features and has been proven to deliver strong performance in classification task [12]. he general structure of a Decision Tree is illustrated in Figure 3.

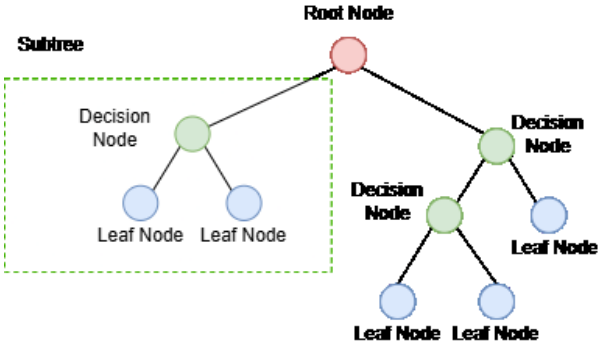


Figure 3. Decision tree

A Decision Tree is a hierarchical model that splits data through a series of tests on feature threshold values. The tree structure consists of nodes and branches, which represent the decision-making process. The selection of the best split is determined using impurity measures such as the Gini Index and Entropy, which quantify the level of uncertainty at each node.The entropy is calculated using the following formula:

$$Entrophy = -\sum_{i}^{C} P_i \log_2 Pi \tag{3}$$

Information Gain is used to determine the most optimal attribute for splitting the data by comparing the decrease in entropy before and after the split. The tree construction process is performed iteratively by selecting the attribute that produces the highest Information Gain until the nodes become homogeneous or meet predefined stopping criteria. The final result is a Decision Tree structure that can be used to classify or predict new data.The Information Gain is defined as:

$$Gain(S, A) = Entrophy(S) - \sum_{v \in V(A)} \frac{|S_v|}{|S|} \times Entrophy(S_v) \tag{4}$$

Where S is the initial dataset before splitting, A is the tested attribute (feature), $V(A)$ represents all possible values of attribute A, S_v is the subset of $S$for attribute value v.

Random Forest (Bagging)

The Random Forest is an ensemble learning algorithm that utilizes multiple Decision Trees to improve prediction accuracy and stability [13]. Each Decision Tree is constructed from a different subset of the data through bootstrap sampling, which is a random sampling technique with replacement, allowing each data instance to be selected more than once. The overall architecture of the Random Forest algorithm is illustrated in Figure 4.

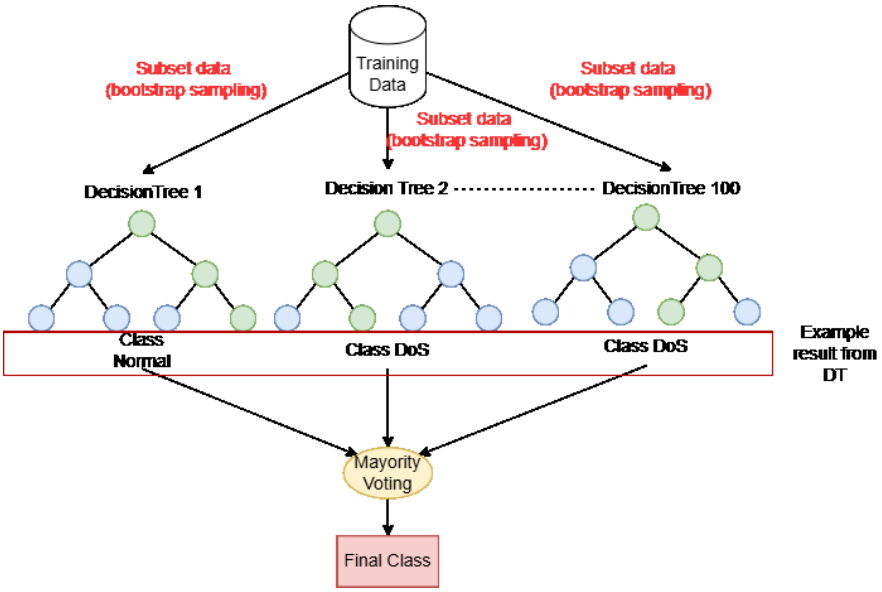


Figure 4. Random Forest Algorithm

During the prediction stage, Random Forest combines the outputs of all trees using a majority voting mechanism, where the class receiving the highest number of votes is selected as the final prediction. This approach is effective in reducing overfitting, which often occurs in a single Decision Tree model, and produces a more robust and reliable model for unseen data. The Random Forest prediction can be expressed as:

$$y(x) = mode(\{h_1(x), h_2(x), \dots, h_T(x)\}) \quad (5)$$

In the implementation of the Random Forest model for DoS attack classification, the model is trained using the prepared training dataset. In this study, the Random Forest model is implemented using the scikit-learn library without applying hyperparameter tuning. Therefore, the model utilizes the default parameter settings provided by scikit-learn for DoS attack classification.

XGBoost (Boosting)

XGBoost (Extreme Gradient Boosting) is an advanced development of the Gradient Boosting algorithm introduced by Dr. Tianqi Chen as a faster, more efficient, and more scalable implementation [14]. This algorithm constructs multiple Decision Trees sequentially, where each new tree is designed to correct the errors (residuals) of the previous tree until the model achieves optimal predictive performance. The general workflow of the XGBoost algorithm is illustrated in Figure 5.

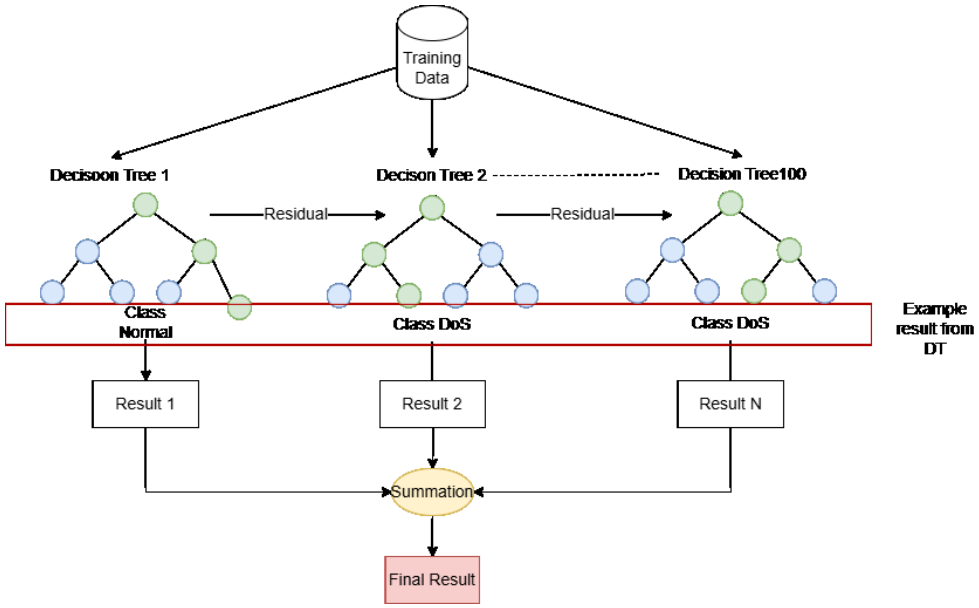


Figure 5. XGBoost Algorithm

In XGBoost, the learning process is optimized through an objective function that combines training loss to measure prediction errors and regularization to prevent overfitting. Residuals are computed at each iteration and are then used to build new Decision Trees through gradient and Hessian calculations. The weight of each leaf node is determined to minimize the objective function, allowing the model accuracy to improve progressively.
The objective function is defined as:

$$obj(\theta) = L(\theta) + \Omega(\theta) \tag{6}$$

The final prediction is obtained through a summation process, where the contributions of all constructed Decision Trees are aggregated and a logistic function is applied for classification tasks. In this study, the scikit-learn library is used to implement XGBoost, which provides high computational performance and ease of use. The complete objective function can be expressed as:

$$obj(\theta) = \sum_{i=i}^{n} l(y_i, \hat{y}_i) + \sum_{k=1}^{t} \Omega(f_k) \tag{7}$$

In this study, XGBoost library is utilized to implement the XGBoost model for DoS attack classification. XGBoost is selected because it is an optimized implementation of gradient boosting that is distributed, efficient, flexible, and portable, enabling parallel processing of boosted trees. In this study, the XGBoost model is implemented using the default parameter settings.

Confusion Matrix and ROC-AUC Evaluation

The Confusion Matrix is an evaluation method that provides a comprehensive overview of the number of correct and incorrect predictions for each class. Through this matrix, classification performance metrics can be derived, allowing the evaluation of model performance in distinguishing between normal traffic and DoS attack traffic. An example of a Confusion Matrix is illustrated in Figure 6.

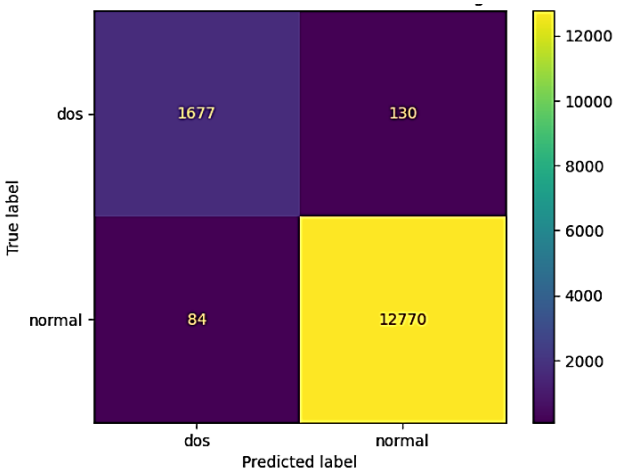


Figure 6. Example of confusion matrix

In addition, this study also employs ROC-AUC as a model evaluation metric. The Receiver Operating Characteristic (ROC) curve illustrates the relationship between the True Positive Rate (TPR) and the False Positive Rate (FPR) across different threshold values, thereby representing the trade-off between sensitivity and misclassification errors. The Area Under the Curve (AUC) measures the area under the ROC curve, with values ranging from 0 to 1. A value closer to 1 indicates better model performance in distinguishing between the two classes. ROC-AUC is particularly effective for binary classification problems with imbalanced class distributions. An example of the ROC-AUC curve is illustrated in Figure 7.

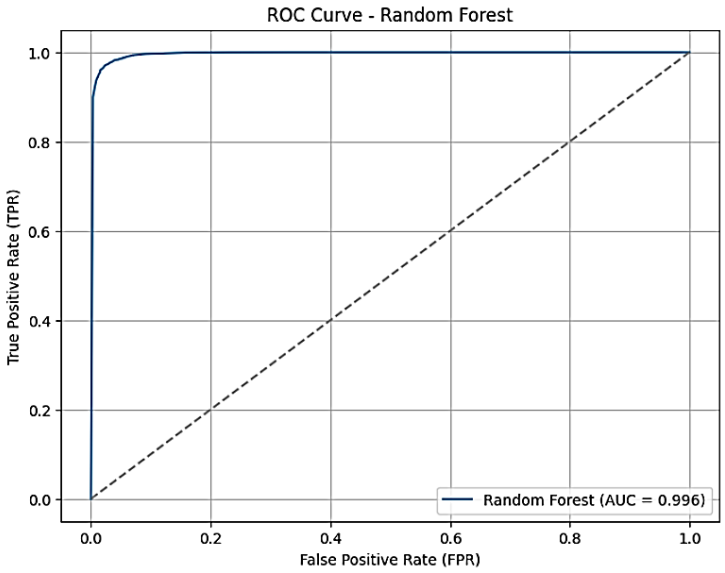


Figure 7. Example of ROC-AUC

Results and Discussions

Results

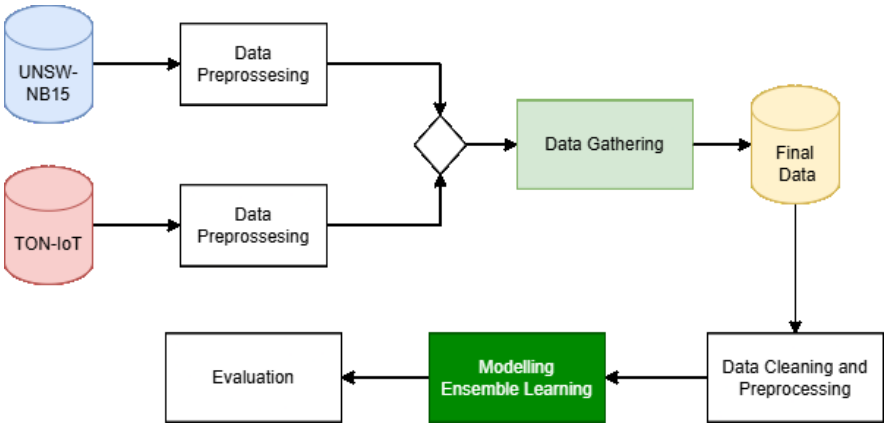
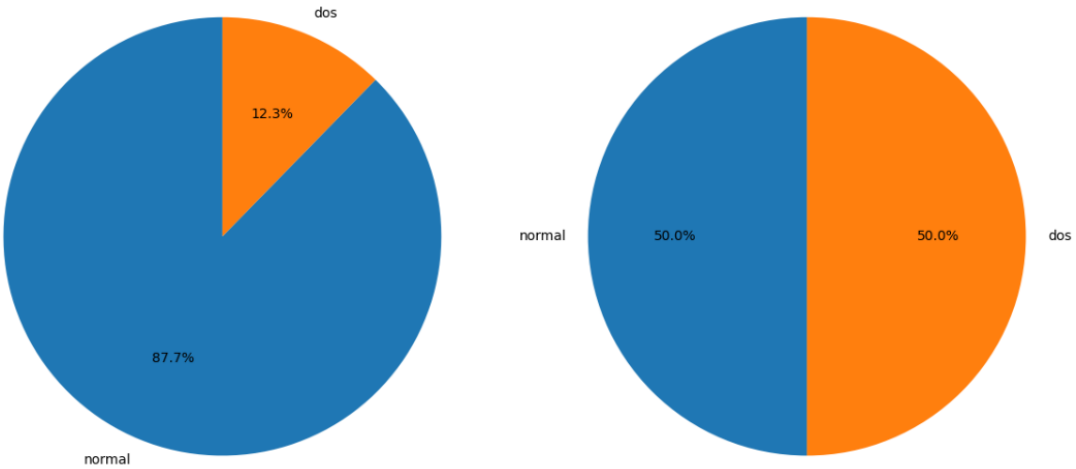


Figure 8. Model development process for DoS attack classification

Based on Figure 8, the UNSW-NB15 and TON-IoT datasets are network traffic datasets used as the basis for developing the DoS attack detection model. Both datasets were combined using a horizontal integration approach, in which only common features were merged to obtain a uniform data structure. Prior to integration, each dataset underwent a preprocessing stage.

The UNSW-NB15 dataset consists of 68,264 records, while the TON-IoT dataset contains 70,000 records. After integration and additional preprocessing, a final dataset of 73,302 records was obtained, consisting of 64,267 normal traffic instances and 9,035 DoS attack instances. The final dataset was split using stratified sampling with an 80:20 ratio, resulting in 58,642 training samples and 14,660 testing samples.

This study analyzes the performance of models trained using both the original imbalanced data and data balanced using the Synthetic Minority Over-sampling Technique (SMOTE). The use of SMOTE is based on previous studies that demonstrated its effectiveness in addressing class imbalance and improving model performance. The distribution of training data classes before and after the application of SMOTE is illustrated in Figure 9.



(a) Before SMOTE (b) After SMOTE

Figure 9. Distribution of training data classes before and after SMOTE

An ensemble learning approach was employed to build the DoS attack detection model due to its ability to improve accuracy, reduce false positives, and adapt to complex attack patterns. Random Forest and XGBoost were implemented using the scikit-learn library. Random Forest was

selected for its stability through majority voting, while XGBoost was chosen for its superior capability in handling complex attack patterns and imbalanced data.

The developed models were evaluated using a confusion matrix to compute performance metrics, including accuracy, precision, recall, F1-score, and ROC-AUC. The evaluation results are presented in Table 4.

Tabel 4. Model evaluation results

Model	Accuracy	Precision	Recall	F1-Score	ROC-AUC
Random Forest (Original Data)	0.9854	0.9515	0.928	0.9402	0.996
Random Forest (SMOTE Data)	0.981	0.956	0.892	0.923	0.996
XGBoost (Original Data)	0.9849	0.9568	0.908	0.9316	0.996
XGBoost (SMOTE Data)	0.972	0.841	0.961	0.961	0.997

Discussions

This study developed four ensemble learning models, namely Random Forest and XGBoost, each trained using both the original dataset and the dataset balanced using the Synthetic Minority Over-sampling Technique (SMOTE), to analyze the impact of class imbalance handling on Denial of Service (DoS) attack detection. The experimental results indicate that the application of SMOTE did not improve model performance and instead reduced precision values, which suggests an increase in false positives. This finding contrasts with previous studies which reported that SMOTE improved classification performance, highlighting that the effectiveness of SMOTE strongly depends on dataset characteristics and the algorithms used [11]. This result may occur because SMOTE generates synthetic samples that do not fully represent real network traffic patterns, causing the model to learn less discriminative features and increasing false positive predictions.

The ensemble learning approach proved effective for DoS attack traffic classification due to its ability to combine multiple base models. Both Random Forest, which applies a bagging approach, and XGBoost, which uses a boosting strategy, demonstrated high performance. These results are consistent with previous studies reporting that ensemble learning significantly enhances network anomaly detection performance [15]. Ensemble models are capable of handling the complexity and variability of DoS attack patterns effectively.

The best-performing model in this study was Random Forest trained on the original dataset, achieving an accuracy of 0.9854 and a ROC-AUC value of 0.996. The balanced precision, recall, and F1-score values indicate that the model can accurately detect DoS attacks with low false positive and false negative rates. The Random Forest model was built using 100 decision trees combined through a majority voting mechanism, resulting in stable and robust predictions across diverse network traffic conditions. The performance evaluation of the best-performing model is illustrated in Figure 10.

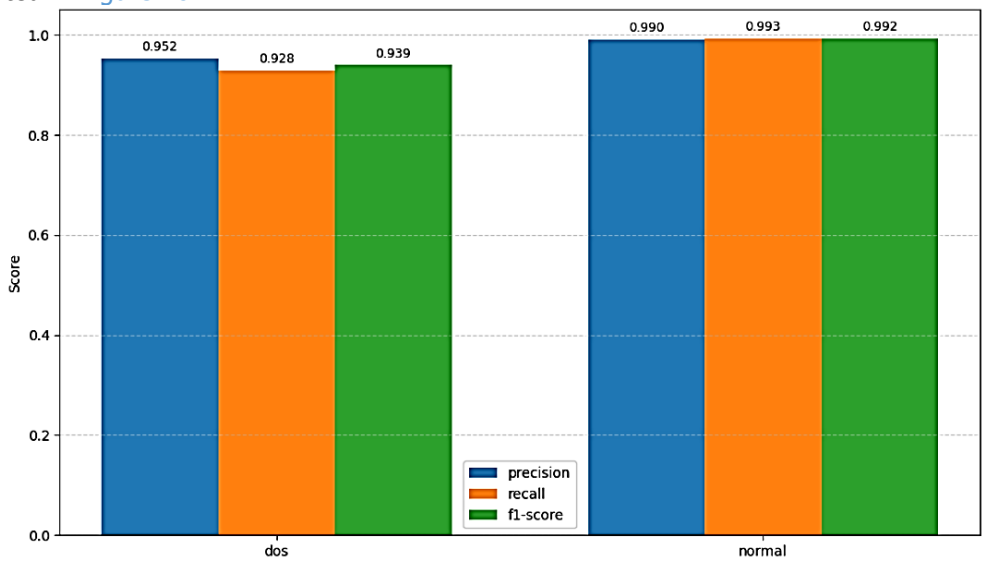


Figure 10. Performance evaluation of the best model

Conclusion

This study utilized network traffic data from the UNSW-NB15 and TON-IoT datasets, which were integrated based on feature compatibility. After the data preprocessing stage, a total of 73,302 records were obtained, consisting of 64,267 normal traffic instances and 9,035 DoS attack instances. The dataset was then split using stratified sampling with an 80:20 ratio into training and testing sets. Four ensemble learning models, namely Random Forest and XGBoost, were trained using both the original dataset and the dataset balanced with SMOTE to analyze the impact of class imbalance handling. The model evaluation stage demonstrated the performance of each developed model.

The ensemble learning approach proved effective for DoS attack traffic classification by combining base learners through Bagging and Boosting mechanisms, thereby improving model performance. In this study, both Random Forest (Bagging) and XGBoost (Boosting) achieved high performance, consistent with previous studies that reported the effectiveness of ensemble methods in handling the complexity of network traffic characteristics. Based on the evaluation results, the best and most stable model was Random Forest trained on the original dataset, achieving an accuracy of 0.9854, precision of 0.9515, recall of 0.928, F1-score of 0.9402, and ROC-AUC of 0.996. These results indicate strong performance in DoS attack classification and outperform the models trained using SMOTE-balanced data.

For future work, this study suggests extending the classification scope to include other types of network attacks beyond Denial of Service (DoS), such as probing, infiltration, and other anomaly-based threats, in order to develop a more comprehensive intrusion detection system. In addition, further experiments should be conducted using alternative class imbalance handling techniques besides SMOTE, such as undersampling methods or hybrid resampling strategies, to investigate their impact on model performance and generalization.

Acknowledgments

The authors would like to express their sincere appreciation to all institutions and researchers who provided the datasets and open-source resources used in this study. Special thanks are also extended to all parties who offered valuable academic guidance and technical support throughout the research process.

References

[1] A. Toh, "Azure DDoS Protection—2021 Q3 and Q4 DDoS attack trends." Accessed: Sep. 24, 2025. [Online]. Available: https://azure.microsoft.com/en-us/blog/azure-ddos-protection-2021-q3-and-q4-ddos-attack-trends/

[2] N. Jeffrey, Q. Tan, and J. R. Villar, "Using Ensemble Learning for Anomaly Detection in Cyber–Physical Systems," *Electronics*, vol. 13, no. 7. 2024. doi: 10.3390/electronics13071391.

[3] J. Vanerio and P. Casas, "Ensemble-learning approaches for network security and anomaly detection," in *Proceedings of the workshop on big data analytics and machine learning for data communication networks*, 2017, pp. 1–6.

[4] N. D. Primadya, A. Nugraha, A. Luthfiarta, and S. Y. Fahrezi, "Optimasi Logistic Regression untuk Deteksi Serangan DoS pada Keamanan IoT," *J. Eksplora Inform.*, vol. 13, no. 2, pp. 245–252, 2024.

[5] A. Harto, M.K. and Basuki, "Deteksi Serangan DDoS pada Jaringan Berbasis SDN dengan Klasifikasi Random Forest," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 5, no. 4, pp. 1329–1333, 2021, [Online]. Available: https://j-ptiik.ub.ac.id/index.php/j-ptiik/article/view/8795

[6] Y. Ariyanto, V. A. H. Firdaus, and H. Pramana, "Klasifikasi Jenis serangan DOS dan Probing pada IDS menggunakan metode K-Nearest Neighbor," in *Seminar Informatika Aplikatif Polinema (SIAP)*, 2020.

[7] D. Firdaus, F. Fahira, and R. Rianti, "Deteksi Anomali dan Serangan Low Rate DDOS dalam Lalu Lintas Jaringan Menggunakan Naive Bayes," *Naratif J. Nas. Riset, Apl. Dan Tek. Inform.*, vol. 5, no. 2, pp. 140–148, 2023.

[8] N. Moustafa and J. Slay, "UNSW-NB15: a comprehensive data set for network intrusion

detection systems (UNSW-NB15 network data set)," in *2015 Military Communications and Information Systems Conference (MilCIS)*, 2015, pp. 1–6. doi: 10.1109/MilCIS.2015.7348942.

[9] S. D. D. Anton, S. Sinha, and H. D. Schotten, "Anomaly-based Intrusion Detection in Industrial Data with SVM and Random Forests," in *2019 International Conference on Software, Telecommunications and Computer Networks (SoftCOM)*, 2019, pp. 1–6. doi: 10.23919/SOFTCOM.2019.8903672.

[10] A. M. A. Rudianto, E. S. Pramukantoro, and D. Kurnianingtyas, "Implementasi Sistem Deteksi Anomali pada Jaringan Komputer dengan Pendekatan XGBoost dan Data SNMP," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 9, no. 2, 2025.

[11] M. Sulistiyono, Y. Pristyanto, S. Adi, and G. Gumelar, "Implementasi algoritma synthetic minority over-sampling technique untuk menangani ketidakseimbangan kelas pada dataset klasifikasi," *Sist. J. Sist. Inf.*, vol. 10, no. 2, pp. 445–459, 2021.

[12] M. M. Ghiasi and S. Zendehboudi, "Application of decision tree-based ensemble learning in the classification of breast cancer," *Comput. Biol. Med.*, vol. 128, p. 104089, 2021.

[13] G. Biau and E. Scornet, "A random forest guided tour," *Test*, vol. 25, no. 2, pp. 197–227, 2016.

[14] T. Chen, "XGBoost: A Scalable Tree Boosting System," *Cornell Univ.*, 2016.

[15] R. Sudiyarno, A. Setyanto, and E. T. Luthfi, "Peningkatan Performa Pendeteksian Anomali Menggunakan Ensemble Learning dan Feature Selection," *Creat. Inf. Technol. J.*, vol. 7, no. 1, pp. 1–9, 2021.