Performance Comparison of IoT Classification Models using Ensemble Stacking and Feature Importance

[1]Nabila Putri Setiawan, [2]Adhitya Nugraha, [3]Ardytha Lutfhiarta, [4]Yudha Mulyana
[1,2,3,4] Informatics Engineering, Faculty of Computer Science, Dian Nuswantoro University
[1,2,3,4]Jl. Imam Bonjol No.207, Pendrikan Kidul, Kec. Semarang Tengah, Kota Semarang, Jawa Tengah
*e-mail: nabilasei@gmail.com

(***received*****:** 9 October 2024, ***revised*****:** 24 October 2024, ***accepted*****:** 27 October 2024)

Abstract

Internet of Things (IoT) security is becoming a top priority as the number of connected devices increases online. This research utilizes the CIC IoT ATTACK 2023 dataset from the University of Brunswick, which consists of 46 million data on various types of attacks on IoT devices, such as DDoS, DoS, Brute Force, Spoofing, and Mirai attacks. To address the imbalance in the dataset, a random undersampling technique is applied to ensure the machine learning model is not biased towards the majority class. The ensemble learning approach was chosen due to its ability to combine the strengths of multiple algorithms, thus improving accuracy and stability in detecting complex IoT attacks. The algorithms used include gradient boosting, bagging, voting, and stacking. In particular, the stacking model, which combines the bagging classifier and gradient boosting, achieved the highest accuracy of 93%. Although the accuracy of the stacking model decreased to 92.4% after feature selection, the precision, recall, and F1-score remained high at 92.0. In addition, the computation time was also reduced from 2111.69 seconds to 1208.27 seconds. These findings indicate that ensemble learning approaches and feature selection techniques have great potential in improving IoT security, providing more reliable and efficient threat detection solutions.

Keywords: *internet of things (IoT) attack , ensemble learning, random undersampling, permutation feature selection*

1 Introduction

The Internet of Things (IoT) has changed the way we interact with our surroundings through billions of connected devices, such as smart devices and sensors. With the ability to collect, transmit, and analyze data in real-time, IoT improves efficiency and productivity across a wide range of sectors, including households, healthcare, industry, and smart cities. However, along with these benefits, security challenges also arise, given that many IoT devices are vulnerable to attacks[1].

Security in the IoT ecosystem is critical, given the potentially serious impact of attacks on devices, such as data theft, system manipulation, and safety risks. According to the Cybersecurity & Infrastructure Security Agency (CISA) report, attacks on IoT devices are increasing significantly, with more than 50% of organizations experiencing at least one IoT security incident by 2023. In addition, the number of DDoS attacks targeting IoT devices increased by 70% compared to the previous year, with 1.5 million IoT devices involved in DDoS attacks in the same year.

Machine learning can detect IoT attacks by analyzing data patterns to identify suspicious activity. Its ability to learn from historical data enables faster and more accurate detection of threats. However, the main challenge in applying machine learning for attack detection is data imbalance. This research utilizes the CIC IoT ATTACK 2023 dataset consisting of 46 million attack data records on IoT devices, focusing on attack class imbalance. This imbalance can result in machine learning models being skewed towards the majority class, making them less effective in detecting less common attacks. To address this imbalance, a *random undersampling* technique is applied. This technique reduces the number of samples from the majority class randomly resulting in a more balanced training dataset and ensuring that the machine learning model can detect different types of attacks more accurately [2]

IoT threat detection using machine learning, particularly with ensemble learning techniques, shows significant advantages in effectiveness and accuracy. Ensemble techniques, such as gradient boosting, bagging, voting, and stacking, combine predictions from multiple models to reduce overfitting and improve stability. By combining the strengths of multiple models, ensemble learning can handle data

 ISSN:2302-8149
e-ISSN:2540-9719

complexity and variability in attack patterns, thereby improving threat detection capabilities that may be missed by a single model [3].

In addition, this research will also perform feature importance analysis on the best model. The addition of feature importance aims to identify which attributes contribute most to the model's decision, which will be compared to evaluate computation and accuracy. With this approach, it is expected to increase the effectiveness of attack detection and create a more secure and reliable IoT ecosystem.

The contributions of this research not only strengthen the security of IoT devices, but also increase the understanding of the application of machine learning technology in the ever-evolving cybersecurity. This proactive and adaptive approach is increasingly important in maintaining the integrity of IoT-based systems in the future.

2 Literature Review

Various studies on Internet of Things (IoT) attack classification show varied approaches and interesting results. In a study conducted by Euklides, the AdaBoost algorithm was used to classify IoT attacks into eight classes. However, this study recorded unsatisfactory results with an accuracy of only 35%. This low accuracy rate may be due to the limitations of the AdaBoost algorithm in handling the complexity and variety of IoT attacks, especially in the absence of applying feature selection techniques that can help reduce data dimensionality and improve model performance [4]. On the other hand, research conducted by Edi Ismanti gave much more positive results. In her research, Edi conducted a comparison between various ensemble learning algorithms before and after the optimization process. The results showed very high accuracy: 98.56% for XGBoost algorithm, 98.47% for Gradient Boosting Machines (GBM), 98.36% for AdaBoost, and 98.33% for Random Forest. This success was achieved thanks to the application of ensemble learning techniques and algorithm optimization through tuning using GridSearchCV, which allowed researchers to find the best combination of parameters that improved the overall performance of the model [5]. Dicky's study also showed the positive impact of applying feature selection to the Decision Tree model, where the accuracy increased to 87.32% by using the Wrapped Based method. This finding confirms that proper feature selection can have a significant impact on the performance of classification models [6]. Meanwhile, Primadya's research adopted the Recursive Feature Elimination (RFE) method, which not only managed to increase the accuracy of the model to 97.80%, but also accelerated the execution time, showing better efficiency in the classification process [7].

Overall, the findings from these studies underscore the importance of applying feature selection techniques and algorithm optimization to improve the accuracy and efficiency of IoT attack classification. The contributions of these studies are significant in the development of better security solutions in the IoT field, given the challenges faced by systems in dealing with increasingly complex and diverse attacks.

3 Research Method

The flowchart in figure 1 illustrates the process flow in a study that uses ensemble learning techniques for data classification. The flowchart includes essential steps from data preprocessing to model evaluation, highlighting the critical role of each component in achieving accurate classification results.

 ISSN:2302-8149
e-ISSN:2540-9719

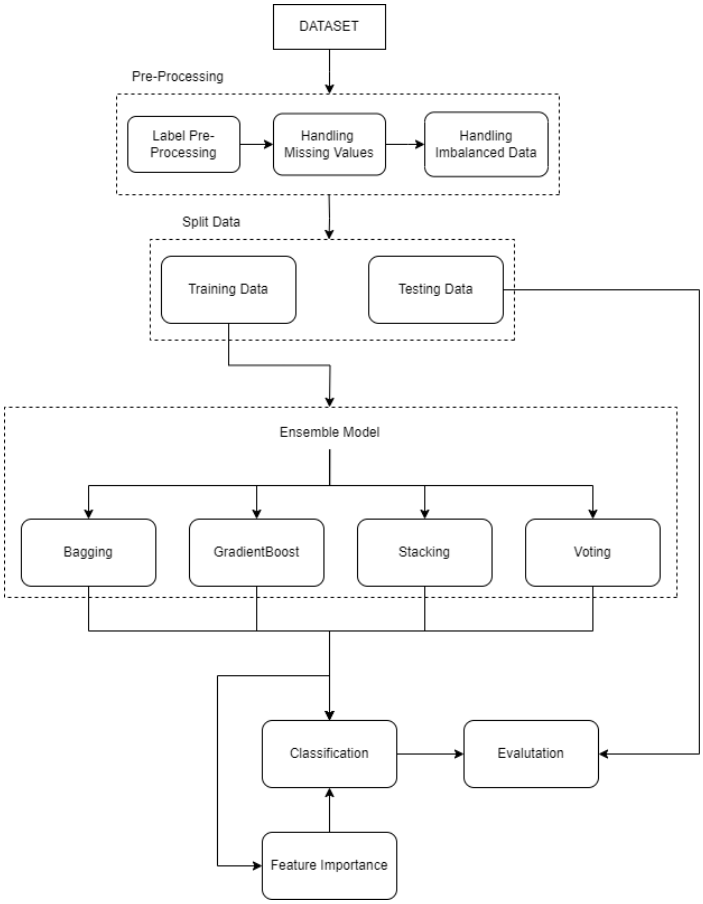

Figure 1. Research flow

The process starts with pre-processing the dataset, which includes handling labels, missing values, and data imbalance. After this stage, the data is divided into training data and testing data. Next, the training data is used in the ensemble model, which consists of several methods such as Bagging, Gradient Boosting, Stacking, and Voting. The result of the ensemble model then goes through a classification stage, where predictions are made, and ends with an evaluation to measure the model's performance. In addition, there is also a step to determine feature importance, which helps understand the contribution of each feature in the model. The overall process aims to improve classification accuracy through the application of ensemble techniques to the processed dataset.

3.1 Dataset

This research utilizes the CIC IOT ATTACK 2023 dataset provided by the University of Brunswick. This dataset consists of 46 million data with 34 labels in it, namely ACK fragmentation, UDP flood, SlowLoris, ICMP flood, RSTFIN flood, PSHACK flood, HTTP flood, UDP fragmentation, TCP flood, SYN flood, SynonymousIP flood, Dictionary brute force, Arp Spoofing, DNS Spoofing, TCP Flood, Http Flood, SYN Flood, UDP Flood, Ping Sweep, OS Scan, Vulnerability Scan, Port Scan, Host Dicovery, Sql Injection, Command Injection, Backdoor Malware, Uploading Attack, XSS, Browser Hijacking, GREIP Flood, Greeth Flood and UDPPlain. The selection of this dataset is based on two main factors, namely the substantial volume of data and the diversity of attack types covered in it.

3.2 Label Pre-processing

Label processing is an important step in data classification in machine learning, which involves identifying and preparing labels for each data for use in model training. This process includes determining the appropriate class, handling missing or unclear data, and customizing the label format

 ISSN:2302-8149
e-ISSN:2540-9719

according to the needs of the model. The goal is to ensure that the training data has consistent and relevant labels, so that the algorithm can learn effectively and produce accurate predictions.

3.3 Random Undersampling

Random undersampling is a technique often used in unbalanced data processing to overcome class imbalance by randomly reducing the number of samples from the majority class so that it is balanced with the minority class [8]. The process involves randomly selecting from the majority class a certain proportion, followed by the removal of those samples. Random undersampling can help improve model performance in the case of unbalanced data.

3.4 Ensemble Learning

Ensemble learning is an approach in machine learning that combines multiple models to improve prediction accuracy. The individual models used in an ensemble often have different strengths and weaknesses or come from different algorithms. By integrating predictions from multiple models, ensemble learning can reduce the weaknesses present in each single model and produce more stable and reliable predictions. The main techniques in ensemble learning include bagging, boosting, voting, and stacking[9].

3.4.1 Stacking

Stacking is one of the techniques in ensemble learning that involves using multiple base models to generate predictions, which are then combined by a "meta-learner" or "meta-model" [10]. This meta-learner is trained to optimize the fusion of predictions from these base models with the ultimate goal of improving the overall quality of predictions. This process involves training the base models on a subset of the data and then using their output as input to the meta-learner, which then generates a final prediction based on the most optimized combination of the base models' predictions [11].

3.4.2 Bagging

Bagging (bootstrap aggregating) is a technique in machine learning used to improve classification accuracy by combining multiple learner models trained randomly on a subset of the training data[12]. This technique generates multiple subsets of the original dataset through random sampling with returns (booststrap). Each subset is used to train a different learner model. The final result of bagging is determined through voting, where the class that gets the most votes from all models becomes the final classification result[13]. By combining predictions from multiple models, bagging can reduce variance, improve stability and accuracy of the model, and make it more resistant to overfitting.

3.4.3 Voting

Voting is a technique in machine learning where several different models, which can use the same or different algorithms, provide predictions against a dataset, and the final result is determined based on the majority of votes from all models. This technique aims to improve prediction accuracy by utilizing variations in predictions from different models. By combining predictions from multiple models, voting can reduce individual prediction errors and produce a more reliable and accurate model [14].

3.4.4 Gradient Boosting

Gradient Boosting is a machine learning technique in which predictive models are built incrementally, with each model attempting to correct the prediction error of the previous model[15]. This process is done by minimizing a loss function using the gradient descent method. The final model is a combination of all the individual models, often in the form of decision trees. Gradient boosting is very effective for handling regression and classification problems, resulting in models with high performance and good accuracy[16].

 ISSN:2302-8149
e-ISSN:2540-9719

3.5 Confusion Matrix

A confusion matrix as shown in Table 1, is a tabular representation that illustrates the performance of a classification model by comparing the model's predicted results with the actual labels in the test data. It organizes predictions into four categories: True Positives, True Negatives, False Positives, and False Negatives, which allow for a comprehensive analysis of the model's behavior. From this matrix, various evaluation metrics such as accuracy, precision, recall, and F1-score can be derived, providing valuable insights into the strengths and weaknesses of the model. By visualizing misclassifications, the confusion matrix helps determine how well the model distinguishes between different classes and highlights areas that may require further optimization or adjustment. It plays a crucial role in model diagnostics, guiding improvements for better classification performance[17].

Table 1. Confusion matrix

		Actual Class	
		+	-
Predict Class	+	TP	FP
	-	FN	TN

Accuracy is the proportion of total correct predictions out of all predictions made by the model. Accuracy gives a general idea of how well the model can predict overall. The formula is given in Equation (1) :

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

Precission measures the proportion of correct positive predictions out of all positive predictions made by the model. It tells us how many positive predictions are correct compared to all positive predictions made by the model. The formula is shown in Equation (2) :

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

Recall also known as sensitivity or true positive rate (TPR), measures the proportion of positive instances that have been correctly identified out of all the positive instances that the model should have identified. It tells us how well the model can recognize all the positive instances that should have been identified. The formula is provided in Equation (3) :

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

F1-Score is the average between Precission and Recall It gives us a single measure of model performance that combines information from both metrics. The formula is defined in Equation (4) :

$$Accuracy = \frac{2\ x\ Precission\ x\ Recall}{Precission+Recall} \quad (4)$$

3.6 Feature Importances

In this research, feature selection is done using permutation feature importances. The process starts with training the model using all features, recording the accuracy as a baseline, and then permuting the value of one feature to measure the change in accuracy. A significant decrease in accuracy indicates that the feature is important, while a small change indicates a less relevant feature. Feature importances refer to the relative contribution of each feature to model predictions. This method can be applied to a wide range of models and does not rely on any particular assumptions, although it can require significant computational resources, especially on large datasets.

 ISSN:2302-8149
e-ISSN:2540-9719

4 Results and Analysis

In this section, we present the results and analysis of research that classifies IoT attacks using machine learning techniques. The main focus is on label processing, handling imbalanced data, model development with Ensemble Learning techniques, as well as evaluation of the importance of features that contribute to model performance. Through these analytical steps, we aim to provide insights into the effectiveness of the approach in detecting and addressing various complex cyberattacks.

4.1 Label Processing

In the label processing stage, there are 8 attack categories used for classification. Based on the dataset distribution in Table 2, the DDoS label has the highest amount of data, which is 33.984.560 data. After the label adjustment process is completed, the next step is to clean the data, including addressing missing values and removing duplicate data. This cleaning process is important to ensure better data quality, so that the machine learning model can be trained with accurate and relevant data, which will ultimately support more optimized analysis results.

Table 2. Number of dataset for 8 labels

Label	Jumlah
DDoS	33.984.560
Dos	8.090.738
Mirai	2.634.124
Benign	1.098.195
Spoofing	486.504
Recon	354.565
Web-Based	24.829
Brute-Force	13.064

4.2 Imbalanced Data

To overcome the large data imbalance, the Random Undersampling technique was used for 8 IoT attack labels. Here are the results of Random Undersampling

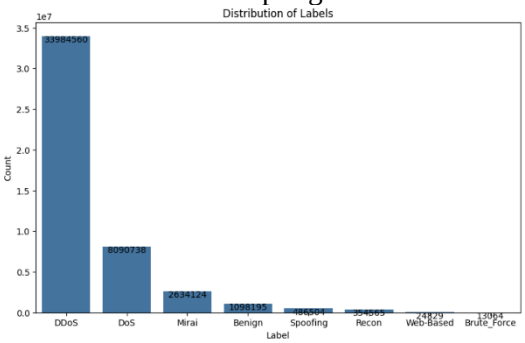

Figure 2. Before random undersampling

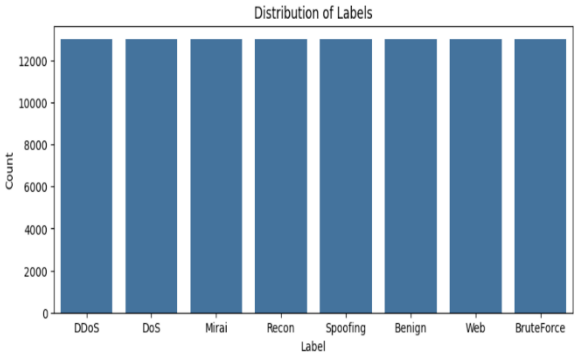

Figure 3. After random undersampling

 ISSN:2302-8149
e-ISSN:2540-9719

The Random Undersampling technique was applied to 8 IoT attack labels to address the significant data imbalance. In Figure 2, it can be seen that all the labels in the undersampling follow the label with the least number of rows, which is Brute-Force with 13.064 rows. Therefore, the labels DDoS with 33.984.586 rows, DoS with 8.090.738 rows, Mirai with 2.634.124 rows, Benign with 1.098.195 rows, Spoofing with 486.504 rows, Recon with 354.565 rows, and Web-based with 24.829 rows are changed to 13.000 rows after undersampling, as shown in Figure 3.

This is done to create a balance between the number of samples in each class, so that the machine learning model can be more effective in classifying the data without being disproportioned by the majority class.

4.3 Modelling

In this research, the modeling stage is carried out by applying Ensemble Learning techniques, namely Bagging, Stacking, Boosting and Voting. For voting and stacking algortima, two algorithms are combined, namely bagging and gradient boosting.

Table 3. Classification using ensemble learning model

Classifier	Accuracy	Precision	Recall	F1-Score
GradientBoost	90.0	90.0	90.0	90.0
Stacking	93.0	93.0	93.0	93.0
Bagging	92.0	92.0	92.0	92.0
Voting	92.0	92.0	92.0	92.0

Table 3 shows that the Stacking method provides the highest accuracy rate of 93%, combining the advantages of Bagging and Gradient Boosting. This technique shows great potential in improving IoT security by providing a more reliable method to detect and address a variety of complex cyber threats.

4.4 Feature Importance

In this section, we will explore the most important features that significantly contribute to the model's performance. The analysis focuses on identifying which features have the highest impact on the prediction accuracy and overall effectiveness of the model.

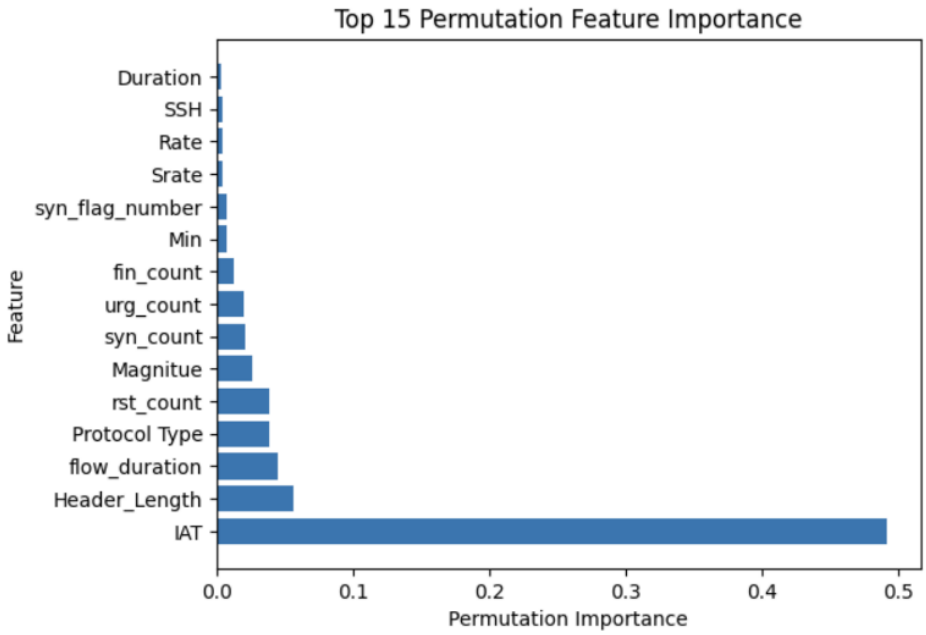

Figure 4. 15 Most importance features of the stacking model

The graph in Figure 4 presents the top 15 features selected from a total of 46 features that were utilized in the model. These features were identified based on their high importance scores, which

 ISSN:2302-8149
e-ISSN:2540-9719

indicate their significant influence on the model's prediction outcomes. The selection process involved evaluating the contribution of each feature to the overall performance of the model, and the top 15 features were chosen because they had the most impact on improving the accuracy and reliability of the model's predictions. Table 4 shows the top 15 features along with their importance scores and standard deviation values. The standard deviation indicates the variability of each feature, helping to assess the stability and consistency of these features in contributing to the model's predictions. By focusing on these key features, the model can make more informed and precise classifications, ultimately enhancing its performance in recognizing patterns and making accurate predictions.

Table 4. Standard deviation value of each feature

Fitur	Importance	Std
IAT	0.4923	0.0010
Header_Length	0.0599	0.0006
Flow_duration	0.0499	0.0009
Protocol Type	0,0390	0.0008
Rst_count	0,0389	0.0008
Magnitue	0,0263	0.0008
Syn_count	0.0207	0.0008
Urg_count	0.2030	0.0004
Fin_count	0.0126	0.0006
Min	0.0078	0.0005
Syn_flag_number	0.0069	0.0005
Srate	0.0047	0.0005
Rate	0.0044	0.0004
SSH	0.0038	0.0002
Duration	0.0035	0.0004

After feature selection using the feature importance method, remodeling is applied to the stacking model that previously showed the highest accuracy.

Table 5. Comparison of stacking evaluation with and without feature importances

Classifier	Accuracy	Precision	Recall	F1-Score	Time(Sec)
Stacking	93.0	93.0	0.93	0.93	2111.69
Stacking + Feature Importances	92.4	92.4	92.4	92.4	1208.27

Modeling was performed using a supercomputer DGX A100. In Table 5, it can be seen that the stacking model without feature selection produces 93% accuracy, with precision, recall, and F1-score of 93.0 each, and computation time of 2111.69 seconds. After feature selection using the feature importance technique, the accuracy of the model slightly decreased to 92.4%. However, the precision, recall, and F1-score remained at 92.0, with the computation time significantly reduced to 1208.27 seconds. These results show that feature selection can improve computational efficiency without significantly sacrificing model performance.

5 Conclusion

This research shows that Ensemble Learning techniques, especially the Stacking method that combines Bagging and Gradient Boosting algorithms, are very effective in detecting attacks on IoT devices, with accuracy reaching 93%. The CIC IoT ATTACK 2023 dataset used has data imbalance, but it is successfully overcome by the Random Undersampling technique, so that the model is able to recognize various types of attacks more evenly. In addition, the application of feature selection reduced computation time by almost half without significantly compromising model performance, with accuracy remaining high at 92.4%. The overall results of this study confirm that the combination of Ensemble Learning and feature selection techniques provides great benefits in improving IoT security. This

 ISSN:2302-8149
e-ISSN:2540-9719

method not only improves the accuracy of attack detection, but also provides a faster and more efficient solution, so it has the potential to be applied in the development of more reliable IoT security systems in the future.

Reference

[1] D. Ratna Sari, "Analisis Keamanan Sistem Informasi dalam Era Internet of Things (IoT)," *Technologia Journal: Jurnal Informatika*, vol. 1, no. 2, pp. 3046–9163, 2024, doi: 10.62872/v2tffe44.

[2] E. Saputro and D. Rosiyadi, "Bianglala Informatika Penerapan Metode Random Over-Under Sampling Pada Algoritma Klasifikasi Penentuan Penyakit Diabetes," vol. 10, no. 1, p. 2022, [Online]. Available: https://archive.ics.uci.edu/ml/machine-learning-

[3] R. I. Arumnisaa and A. W. Wijayanto, "SISTEMASI: Jurnal Sistem Informasi Perbandingan Metode Ensemble Learning: Random Forest, Support Vector Machine, AdaBoost pada Klasifikasi Indeks Pembangunan Manusia (IPM) Comparison of Ensemble Learning Method: Random Forest, Support Vector Machine, AdaBoost for Classification Human Development Index (HDI)." [Online]. Available: http://sistemasi.ftik.unisi.ac.id

[4] E. C. P. Neto, S. Dadkhah, R. Ferreira, A. Zohourian, R. Lu, and A. A. Ghorbani, "CICIoT2023: A Real-Time Dataset and Benchmark for Large-Scale Attacks in IoT Environment," *Sensors*, vol. 23, no. 13, Jul. 2023, doi: 10.3390/s23135941.

[5] E. Ismanto, J. Al Amien, and V. Vitriani, "A Comparison of Enhanced Ensemble Learning Techniques for Internet of Things Network Attack Detection," *MATRIK : Jurnal Manajemen, Teknik Informatika dan Rekayasa Komputer*, vol. 23, no. 3, pp. 543–556, Jun. 2024, doi: 10.30812/matrik.v23i3.3885.

[6] D. Setiawan, A. Nugraha, and A. Luthfiarta, "Komparasi Teknik Feature Selection Dalam Klasifikasi Serangan IoT Menggunakan Algoritma Decision Tree," *JURNAL MEDIA INFORMATIKA BUDIDARMA*, vol. 8, no. 1, p. 83, Jan. 2024, doi: 10.30865/mib.v8i1.6987.

[7] N. Dwi Primadya, A. Nugraha, A. Luthfiarta, and Y. Fahrezi, "Optimasi Logistic Regression untuk Deteksi Serangan DoS pada Keamanan IoT", doi: 10.30864/eksplora.v13i2.1065.

[8] R. Amelia *et al.*, "Komparasi Teknik Undersampling Dan Oversampling Pada Regresi Logistik Biner Dalam Menduga Faktor Determinan Berhenti Merokok Penduduk Lanjut Usia," 2021.

[9] Y. Pristyanto, "Penerapan Metode Ensemble Untuk Meningkatkan Kinerja Algoritme Klasifikasi Pada Imbalanced Dataset," 2019. [Online]. Available: https://archive.ics.uci.edu/ml/datasets/User+Knowledge

[10] B. Sunarko *et al.*, "Edu Komputika Journal Penerapan Stacking Ensemble Learning untuk Klasifikasi Efek Kesehatan Akibat Pencemaran Udara," *Edu Komputika*, vol. 10, no. 1, 2023, [Online]. Available: http://journal.unnes.ac.id/sju/index.php/edukom

[11] Y. Zhang, J. Liu, and W. Shen, "A Review of Ensemble Learning Algorithms Used in Remote Sensing Applications," Sep. 01, 2022, *MDPI*. doi: 10.3390/app12178654.

[12] M. Ibnu, U. Rosyidi, and N. Rochmawati, "Teknik Bagging Pada Algoritma Klasifikasi Decision Tree dan SVM Untuk Klasifikasi SMS Berbahasa Indonesia," *Journal of Informatics and Computer Science*, vol. 05, 2023.

[13] F. Churniansyah, D. Wahyu Utomo, and D. Redaksi, "Jurnal Nasional Teknologi dan Sistem Informasi Attribution-ShareAlike 4.0 International Some rights reserved Artikel Penelitian Teknik Bagging pada Ensemble Learning untuk Kategorisasi Produk E-Commerce Sejarah Artikel," *Pendrikan Kidul, Kec. Semarang Tengah*, vol. 50131, no. 207, doi: 10.25077/TEKNOSI.v10i1.2024.92-80.

[14] S. Joses, S. Quinevera, R. Mardianto, D. Yulvida, and A. Mazharuddin Shiddiqi, "JEPIN (Jurnal Edukasi dan Penelitian Informatika) Pendekatan Metode Ensemble Learning untuk Deteksi Serangan DDoS menggunakan Soft Voting Classifier," 2024.

[15] L. Maretva Cendani and A. Wibowo, "Perbandingan Metode Ensemble Learning pada Klasifikasi Penyakit Diabetes," 2022.

[16] M. Labib Mu'tashim *et al.*, "Klasifikasi Ketepatan Lama Studi Mahasiswa Dengan Algoritma Random Forest Dan Gradient Boosting (Studi Kasus Fakultas Ilmu Komputer Universitas Pembangunan Nasional Veteran Jakarta)," 2023.

[17] M. K. Suryadewiansyah, T. Endra, and E. Tju, "Jurnal Nasional Teknologi dan Sistem Informasi Naïve Bayes dan Confusion Matrix untuk Efisiensi Analisa Intrusion Detection System Alert", doi: 10.25077/TEKNOSI.v8i2.2022.081-088.