Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 10, Number 3, July 2026
DOI : https://doi.org/10.33395/sinkron.v10i3.16227
e-ISSN : 2541-2019
p-ISSN : 2541-044X

Comparative Evaluation of Machine Learning Algorithms for Intrusion Detection Systems

Reza Amru Nismara[1)*], **Rama Aria Megantara**[2)],
[1,2)]Teknik Informatika, Fakultas Ilmu Komputer, Universitas Dian Nuswantoro
[1)]oreoblaster85@gmail.com, [2)] aria@dsn.dinus.ac.id

Submitted :May 17, 2026 | **Accepted** : June 14, 2026 | **Published** : July 5, 2026

Abstract: Performance estimates of intrusion detection models may vary depending on how the training and testing data are separated. Although random split is commonly used in IDS experiments, network traffic often follows time-dependent patterns that differ from one day to another. This study compares random split, single temporal split, and rolling temporal split to examine whether random evaluation produces overly optimistic performance estimates. The CIC-IDS2017 dataset was used because it contains network traffic collected across several days and includes benign as well as malicious activities. The evaluation involved five classical learning models: Decision Tree, Random Forest, Logistic Regression, K-Nearest Neighbors, and Linear Support Vector Machine. The dataset was prepared by combining daily traffic files, removing irrelevant and invalid features, converting labels into binary classes, and applying consistent preprocessing for all models. Performance was measured using accuracy, precision, recall, F1-score, ROC-AUC, PR-AUC, training time, and prediction time. The results show that random split produced very high scores, with several models reaching F1-scores close to 1.0. In contrast, temporal evaluation caused a clear performance decrease, with single temporal F1-scores ranging from approximately 0.60 to 0.71, while rolling temporal validation showed that model performance varied across different chronological testing periods. These findings indicate that random split may overestimate IDS model performance because similar traffic patterns can appear in both training and testing data. Therefore, time-aware evaluation provides a more realistic strategy for assessing IDS model generalization.

Keywords: CIC-IDS2017, Evaluation Bias, Intrusion Detection Systems, Machine Learning, Random Split, Temporal Evaluation, Rolling Temporal Split

INTRODUCTION

Intrusion Detection Systems (IDS) are used to monitor network traffic and identify activities that may indicate attacks, misuse, or abnormal behavior. As network communication becomes larger and more dynamic, traditional rule-based detection may not be sufficient to recognize new attack patterns. For this reason, machine learning has become widely used in IDS research because it can learn traffic characteristics from data and classify network activities into benign or malicious categories (Samantaray et al., 2024; Singh et al., 2024).

Several machine learning algorithms have been applied to intrusion detection tasks, including Logistic Regression, K-Nearest Neighbors (KNN), Support Vector Machine (SVM), Decision Tree, and Random Forest. Prior studies show that tree-based and ensemble methods often obtain strong results on benchmark datasets such as CIC-IDS2017, while SVM and KNN can also perform competitively depending on the attack scenario and dataset characteristics (Maseer et al., 2021; Oluwakemi et al., 2023; Tripathy & Behera, 2023). However, the reported performance of IDS models is not only affected by the algorithm, but also by preprocessing, feature selection, dataset structure, and the evaluation strategy used during training and testing (Samantaray et al., 2024; Singh et al., 2024).

In most existing studies, model evaluation is conducted using random split or cross-validation, where training and testing data are randomly sampled from the same dataset. This approach assumes that the data are independent and identically distributed (i.i.d.). However, in real-world scenarios, network traffic data exhibit temporal dependencies, where patterns of normal and malicious activities may change over time. Consequently, random split evaluation may lead to temporal data leakage, where similar patterns appear in both training and testing data,

*name of corresponding author

This is anCreative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

e-ISSN : 2541-2019
p-ISSN : 2541-044X

resulting in overly optimistic performance estimates (Chua & Salam, 2022; Mondragon et al., 2025; Singh et al., 2024).

The CIC-IDS2017 dataset is widely used in IDS research because it provides realistic network traffic data collected over multiple days with various attack scenarios. This dataset inherently contains temporal characteristics, making it suitable for evaluating IDS models under time-aware conditions (Chua & Salam, 2022; Maseer et al., 2021; Mondragon et al., 2025). However, the improper use of random split on such temporal data may produce misleading conclusions regarding model performance (Mondragon et al., 2025; Singh et al., 2024).

Based on these issues, this study aims to evaluate the performance of several machine learning algorithms for intrusion detection while analyzing the impact of different data splitting strategies. Specifically, this research compares random split, single temporal split, and rolling temporal split using the CIC-IDS2017 dataset. The main research question addressed in this study is whether random split leads to optimistic evaluation results compared with time-aware evaluation strategies. The findings of this study are expected to provide a more realistic evaluation framework for IDS. In addition, this study highlights the importance of considering temporal characteristics in model evaluation (Chua & Salam, 2022; Mondragon et al., 2025).

LITERATURE REVIEW

Research on machine learning-based IDS has increased because network traffic is becoming more complex and attack behavior continues to evolve. Machine learning models are commonly used to learn traffic patterns and distinguish benign activities from malicious ones based on selected traffic features. Algorithms such as Decision Tree, Random Forest, Support Vector Machine, Logistic Regression, and K-Nearest Neighbors have been widely investigated because they represent different learning mechanisms, including linear classification, distance-based classification, and non-linear tree-based modeling (Alzahrani & Alenazi, 2021; Oluwakemi et al., 2023; Samantaray et al., 2024; Singh et al., 2024).

Despite the high performance reported in many studies, the evaluation of IDS models remains a critical challenge. Most existing research relies on random data splitting techniques, which may lead to overly optimistic performance results due to data leakage and the similarity between training and testing samples. Such evaluation approaches often fail to reflect real-world scenarios where network traffic evolves over time (Chua & Salam, 2022; Mondragon et al., 2025; Singh et al., 2024).

Therefore, it is important to review existing studies not only based on their performance metrics but also in terms of their evaluation strategies and limitations. A summary of relevant studies on machine learning-based IDS, including their datasets, algorithms, evaluation methods, key findings, and limitations, is presented in Table 1.

Machine Learning Algorithms for IDSs

Various machine learning algorithms have been widely applied in the development of Intrusion Detection Systems (IDS) to detect malicious network activities. Linear models such as Logistic Regression and Support Vector Machine are commonly used due to their stability and interpretability. In contrast, tree-based algorithms, including Decision Tree and Random Forest, are capable of capturing complex non-linear relationships between features and are often reported to achieve higher performance in IDS tasks (Oluwakemi et al., 2023; Singh et al., 2024; Thockchom et al., 2023).

In addition, distance-based methods such as K-Nearest Neighbors (KNN) are utilized to identify similarities in network traffic patterns. Several studies have shown that tree-based models tend to outperform other algorithms on datasets such as CIC-IDS2017 (Chen et al., 2021; Rodríguez et al., 2022). However, the reported performance is highly dependent on the evaluation strategy employed, particularly whether random or temporal data splitting is used.

Table 1. Summary of Related Studies on ML-based IDS Evaluation

Authors & Year	Dataset	Algorithms	Evaluation	Metrics	Key Findings	Limitation
(Chua & Salam, 2022)	CIC-IDS2017	Multiple ML algorithms (DT, RF, SVM, KNN)	Random split	Accuracy, Precision, Recall, F1-score	Machine learning algorithms achieved high intrusion detection performance on CIC-IDS2017 dataset and were effective for network traffic classification.	Evaluation relied on random split and did not analyze temporal dependency or data leakage effects.

*name of corresponding author
This is anCreative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

(Singh et al., 2024)	CIC-IDS2017	Decision Tree, Random Forest, SVM, KNN	Random split	Accuracy, Precision, Recall, F1-score	Comparative analysis showed that tree-based models achieved better intrusion detection performance compared to several conventional machine learning algorithms.	Evaluation focused on random split and did not consider temporal dependency or real-world temporal generalization.
(Chen et al., 2021)	CIC-IDS2017	ADASYN + Random Forest	Comparative experiment	Precision, Recall, F1-score, AUC	ADASYN combined with Random Forest improved intrusion detection performance on imbalanced CIC-IDS2017 data.	Focuses on class imbalance handling and does not analyze temporal dependency or time-aware generalization.
(Mondragon et al., 2025)	14 network flow datasets	Seven categories of ML algorithms	Benchmark comparison	Classification and computational performance metrics	Provides a benchmark across multiple network flow datasets and highlights the need for standardized evaluation in IDS research.	Focuses on broad benchmark comparison and does not specifically compare random split with temporal split on CIC-IDS2017.
Rodríguez et al. (2022)	CIC-IDS2017	Decision Tree (DT), Random Forest (RF)	Random split	Accuracy, Precision, Recall, F1-score	Tree-based models achieve very high performance, with F1-scores close to 1.0 on the CIC-IDS2017 dataset.	Evaluation uses random split only, which may lead to overly optimistic performance and does not reflect temporal generalization.

Research Gap and Differentiation

Based on the comparison of previous studies presented in Table 1, several important observations can be identified. Although numerous studies have reported high performance in machine learning-based IDS, most of them still rely on random data splitting for model evaluation (Maseer et al., 2021; Oluwakemi et al., 2023; Singh et al., 2024). This evaluation approach may lead to overly optimistic results due to the similarity between training and testing data, which does not accurately represent real-world network conditions (Chua & Salam, 2022; Mondragon et al., 2025). Furthermore, limited attention has been given to temporal data characteristics and their impact on model generalization (Chua & Salam, 2022; Mondragon et al., 2025).

Therefore, there is a need for a more realistic evaluation framework that considers temporal data splitting to better simulate real-world intrusion detection scenarios. This study addresses this gap by comparing the performance of several machine learning algorithms using random split, single temporal split, and rolling temporal split strategies on the CIC-IDS2017 dataset. Unlike previous studies that mainly focus on achieving high classification accuracy, this study emphasizes the impact of evaluation strategy on model reliability and generalization under time-dependent network traffic conditions. The novelty of this study lies in explicitly evaluating the influence of data splitting strategies on the perceived generalization capability of machine learning-based IDS models. Unlike previous studies that mainly report high detection performance using random split evaluation, this study compares random split, single temporal split, and rolling temporal split on the CIC-IDS2017 dataset to reveal potential optimistic bias caused by randomly mixed traffic records. Therefore, the contribution of this study is not limited to comparing machine learning algorithms, but also to demonstrating the importance of time-aware evaluation in IDS performance assessment.

METHOD

This research was designed as an experimental comparison of machine learning-based IDS models under three different data separation scenarios. The main purpose is to observe how random split, single temporal split, and rolling temporal split influence the estimated generalization ability of each model.

*name of corresponding author

This is anCreative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

e-ISSN : 2541-2019
p-ISSN : 2541-044X

Dataset **Description**

This study uses the CIC-IDS2017 dataset, which contains network traffic records collected from Monday to Friday and includes both normal and attack traffic. Because the data are organized by day and include several attack scenarios, the dataset is suitable for comparing random and time-aware evaluation strategies (Maseer et al., 2021; Rodríguez et al., 2022).

Data Preparation and Preprocessing

All daily CSV files were combined into a single dataset before preprocessing. Data cleaning was performed by removing irrelevant and non-informative features, such as identifiers that do not contribute to model learning. Missing values and infinite values were handled to maintain data consistency. The original multiclass labels were converted into binary classes, namely benign and attack, to simplify the intrusion detection task.

Feature Selection

Non-numeric features, constant features, and features containing entirely missing values were removed before model training. This step ensures that only relevant numerical features are used and reduces noise that may degrade classification performance (Bakro et al., 2023; Mhawi et al., 2022).

Data Splitting Strategy

Three evaluation strategies were implemented in this study: random split, single temporal split, and rolling temporal split. In the random split strategy, the dataset was divided into training and testing sets using an 80:20 stratified approach to maintain class distribution. In the single temporal split strategy, data from Monday to Thursday were used for training, while data from Friday were used for testing. This setting was designed to simulate real-world IDS conditions, where models are trained using historical network traffic and evaluated on future unseen traffic.

In addition to the single temporal holdout split, this study also applied rolling temporal split validation to provide a more robust time-aware evaluation. The rolling temporal evaluation consisted of three chronological splits: Monday–Tuesday for training and Wednesday for testing, Monday–Tuesday–Wednesday for training and Thursday for testing, and Monday–Tuesday–Wednesday–Thursday for training and Friday for testing. The Monday-only training split was not used because the Monday subset contains only benign traffic, while supervised binary classification requires both benign and attack classes during model training (Chua & Salam, 2022; Mondragon et al., 2025).

Model Implementation

Five machine learning algorithms are used in this study: Decision Tree (DT), Random Forest (RF), Logistic Regression (LR), K-Nearest Neighbors (KNN), and Support Vector Machine (SVM). These models are selected due to their effectiveness and widespread use in intrusion detection tasks (Maseer et al., 2021; Rodríguez et al., 2022; Singh et al., 2024).

Evaluation Metrics

Model performance is evaluated using multiple metrics, including accuracy, precision, recall, F1-score, ROC-AUC, and PR-AUC (Maseer et al., 2021; Rodríguez et al., 2022; Tripathy & Behera, 2023). In addition, computational performance is measured using training time and prediction time.

*name of corresponding author
This is anCreative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 10, Number 3, July 2026
DOI : https://doi.org/10.33395/sinkron.v10i3.16227
e-ISSN : 2541-2019
p-ISSN : 2541-044X

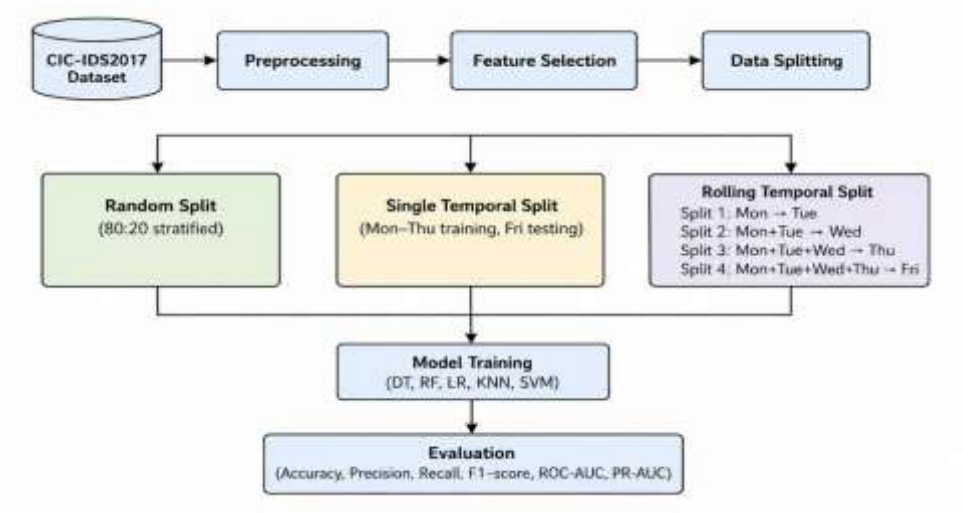

Fig. 1. Proposed machine learning evaluation pipeline for IDS using random split, single temporal split, and rolling temporal split.

As shown in Figure 1, the experimental workflow starts from the CIC-IDS2017 dataset, followed by preprocessing and feature selection. The processed data are then evaluated using three splitting strategies, namely random split, single temporal split, and rolling temporal split. After the splitting process, the selected machine learning models are trained and evaluated using accuracy, precision, recall, F1-score, ROC-AUC, and PR-AUC.

Evaluation Procedure

Each algorithm was first evaluated using random split and single temporal split. The results from these two strategies were compared to determine whether random split produced inflated performance estimates. After that, rolling temporal split validation was conducted to examine the stability of model performance across different chronological testing periods. This additional evaluation was used to strengthen the analysis of temporal generalization and reduce dependence on a single future testing day.

Validity Consideration

To maintain a fair comparison, the same preprocessing procedure, selected features, and evaluation metrics were applied to all models. Thus, the observed performance differences can be attributed mainly to the splitting strategy used in the experiment.

RESULT

This section reports the experimental results obtained from evaluating five machine learning models on CIC-IDS2017 using random split, single temporal split, and rolling temporal split. The evaluated models consist of Decision Tree (DT), Random Forest (RF), Logistic Regression (LR), K-Nearest Neighbors (KNN), and Linear Support Vector Machine (SVM). The evaluation was conducted using multiple performance metrics, namely accuracy, precision, recall, F1-score, ROC-AUC, and PR-AUC. In addition, rolling temporal split validation was used to examine the stability of model performance across different chronological testing periods.

Random Split Performance

Under the random split scenario, most models achieved high classification performance. Tree-based models such as Random Forest and Decision Tree demonstrated near-perfect results across multiple metrics. Logistic Regression and Linear SVM also showed strong performance, while KNN achieved competitive results despite higher computational cost.

Temporal Split Performance

However, when evaluated using temporal split, where training data was taken from Monday to Thursday and testing data from Friday, a noticeable decrease in performance was observed across all models. This indicates that models trained on historical data may struggle to generalize when evaluated on future unseen data with different traffic patterns (Chua & Salam, 2022; Mondragon et al., 2025). Among the evaluated models, Linear SVM achieved the highest F1-score under temporal conditions, indicating relatively better generalization capability compared to other models. Logistic Regression also maintained stable performance, while tree-based models such

*name of corresponding author

This is anCreative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 10, Number 3, July 2026
DOI : https://doi.org/10.33395/sinkron.v10i3.16227
e-ISSN : 2541-2019
p-ISSN : 2541-044X

as Random Forest and Decision Tree experienced a more significant decline in performance. KNN showed the lowest performance under temporal evaluation, along with the highest prediction time.

Comparison Among Random, Single Temporal, and Rolling Temporal Split

The comparison among the three evaluation strategies highlights a consistent pattern: models evaluated using random split tend to produce overly optimistic performance results. In contrast, single temporal split provides a more realistic evaluation scenario because the models are trained on historical traffic and tested on future unseen traffic. Rolling temporal split further extends this analysis by evaluating model performance across multiple chronological testing periods. This additional validation shows that model performance is not always stable across different days, indicating that IDS models may be sensitive to temporal changes in network traffic patterns (Chua & Salam, 2022; Mondragon et al., 2025).

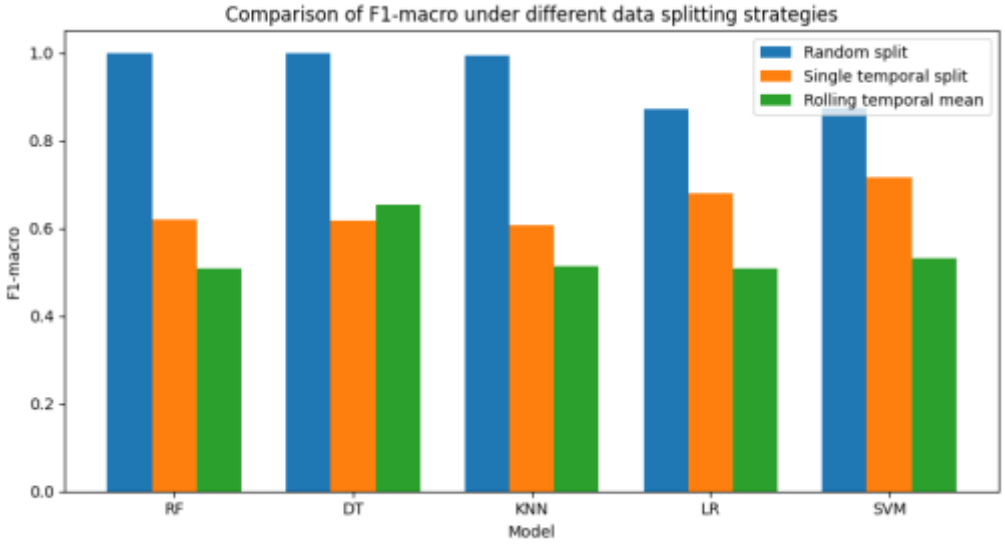

Fig. 2. Grouped bar chart comparison of F1-macro under random split, single temporal split, and rolling temporal split.

Figure 2 presents a grouped bar chart comparison of F1-macro scores under random split, single temporal split, and rolling temporal split. The figure shows that random split produced the highest scores for most models, while temporal evaluation resulted in lower and more variable performance. This confirms that random split may overestimate IDS performance when traffic records with similar temporal characteristics are randomly distributed into both training and testing sets.

Table 2. Summary of model performance under random and temporal split.

Model	Random F1-score	Temporal F1-score	Performance Drop
RF	1.00	0.62	↓ 38%
DT	1.00	0.61	↓ 39%
KNN	0.99	0.60	↓ 39%
LR	0.87	0.68	↓ 19%
SVM	0.88	0.71	↓ 17%

Table 3. F1-macro performance under rolling temporal split validation.

Split	Split Scenario	Testing Day	DT	RF	KNN	LR	SVM
Split 1	Monday+Tuesday	Wednesday	0.388	0.388	0.390	0.389	0.390
Split 2	Monday+Tuesday+Wednesday	Thursday	0.951	0.517	0.540	0.458	0.491
Split 3	Monday+Tuesday+Wednesday+Thursday	Friday	0.618	0.619	0.606	0.680	0.716

Table 3 shows the F1-macro performance under rolling temporal split validation. The results indicate that model performance varied across different chronological testing periods. In Split 1, all models achieved low F1-macro scores, suggesting that the traffic patterns in Wednesday data were difficult to generalize from Monday–Tuesday training data. In Split 2, Decision Tree achieved the highest F1-macro score, while other models remained considerably lower. In Split 3, Linear SVM achieved the best F1-macro score when the models were trained on

*name of corresponding author

This is anCreative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 10, Number 3, July 2026
DOI : https://doi.org/10.33395/sinkron.v10i3.16227
e-ISSN : 2541-2019
p-ISSN : 2541-044X

Monday–Thursday data and tested on Friday data. These findings show that temporal generalization is unstable across different testing days and support the need for time-aware evaluation in IDS experiments. Based on the average rolling temporal results, Decision Tree achieved the highest mean F1-macro score of 0.652, followed by Linear SVM with 0.532, KNN with 0.512, Logistic Regression with 0.509, and Random Forest with 0.508. However, the relatively high variation across splits indicates that a single temporal holdout may not fully represent the stability of IDS model performance over time.

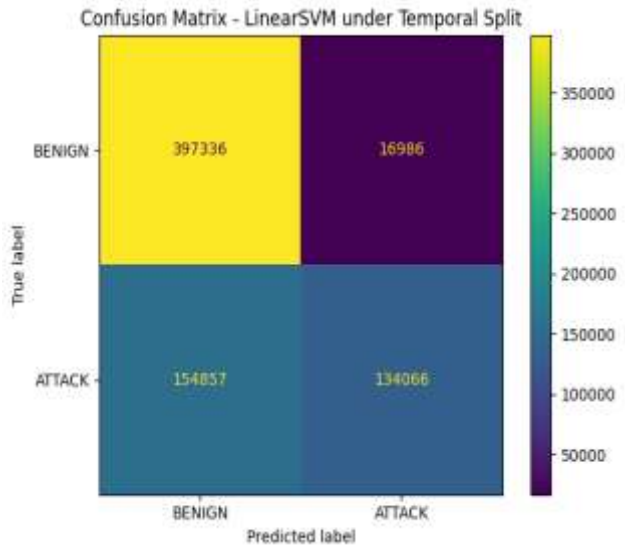

Fig. 3. Confusion matrix of Linear SVM under single temporal split evaluation.

Figure 3 presents the confusion matrix of Linear SVM, which achieved the best F1-macro under the single temporal split. The model produced 397,336 true negatives, 16,986 false positives, 154,857 false negatives, and 134,066 true positives. Although Linear SVM showed the best overall temporal F1-macro, the high number of false negatives indicates that many attack instances were still misclassified as benign. This result highlights the difficulty of detecting future attack traffic when the model is trained only on historical traffic patterns.

DISCUSSIONS

Detection Accuracy

The experimental results demonstrate that all evaluated machine learning models achieved high classification performance under the random split scenario. Tree-based models such as Random Forest and Decision Tree obtained near-perfect F1-scores, while Logistic Regression and Linear SVM also showed competitive results. However, this performance must be interpreted cautiously because random splitting may introduce data leakage due to the similarity between training and testing data (Chua & Salam, 2022; Mondragon et al., 2025).

Generalization and Data Splitting Impact

A significant finding of this study is the performance discrepancy between random and temporal data splitting strategies. When evaluated using temporal split, where models were trained on earlier data and tested on future data, all models experienced a noticeable decline in performance. This indicates that models trained using random split tend to overestimate their generalization capability. Temporal evaluation provides a more realistic representation of IDS deployment because network traffic patterns may evolve over time (Chua & Salam, 2022; Mondragon et al., 2025).

False Positive Analysis

Although overall classification performance was high, the confusion matrix analysis shows that temporal evaluation still produced a considerable number of false negatives. As shown in Figure 3, Linear SVM produced 397,336 true negatives, 16,986 false positives, 154,857 false negatives, and 134,066 true positives under the single temporal split. The high number of false negatives indicates that many attack instances were misclassified as benign. This is critical in intrusion detection systems because false negatives may allow malicious traffic to remain undetected. Therefore, temporal evaluation not only reduces overall performance scores but also reveals detection risks that may be hidden under random split evaluation.

*name of corresponding author

This is anCreative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

e-ISSN : 2541-2019
p-ISSN : 2541-044X

Computational Performance

In addition to classification performance, computational aspects were also considered. Tree-based models, particularly Decision Tree, demonstrated faster training and prediction times. In contrast, KNN required longer prediction time due to its distance-based computation. Linear models such as Logistic Regression and SVM provided a balance between performance and computational efficiency, making them suitable for practical deployment scenarios (Arcos-Argudo et al., 2025).

Limitations

This study has several limitations. First, the evaluation was conducted using a single dataset (CIC-IDS2017), which may not fully represent diverse real-world network environments. Second, only classical machine learning algorithms were considered, without exploring deep learning or hybrid approaches. Third, temporal splitting was applied in a simplified manner (based on weekdays), which may not capture more complex temporal patterns in network traffic. Future work should consider multiple datasets, more advanced models, and more robust temporal validation strategies to further improve generalization analysis (Udurume et al., 2024).

CONCLUSION

This study evaluated five machine learning algorithms for intrusion detection using the CIC-IDS2017 dataset under random split, single temporal split, and rolling temporal split strategies. The results show that random split produced very high performance scores, with several models achieving F1-scores close to 1.0. However, performance decreased substantially under time-aware evaluation, indicating that random split may produce overly optimistic estimates due to the similarity between training and testing traffic patterns.

The single temporal split results showed that Linear SVM achieved the best temporal F1-macro score, while tree-based models such as Random Forest and Decision Tree experienced larger performance drops compared with their random split results. Rolling temporal split validation further showed that model performance varied across different chronological testing periods. This finding indicates that IDS model generalization is not always stable over time and that evaluation based on only one temporal holdout may not fully represent real-world deployment conditions.

The confusion matrix analysis also showed that false negatives remain a major concern under temporal evaluation. Many attack instances were still misclassified as benign, which is critical in intrusion detection because undetected attacks may cause security risks. Therefore, this study concludes that time-aware evaluation is necessary for assessing IDS model reliability more realistically. Future work should extend the evaluation to additional IDS datasets, more attack scenarios, and advanced learning models to further validate the robustness of temporal evaluation strategies.

REFERENCES

Alzahrani, A. O., & Alenazi, M. J. F. (2021). *future internet Designing a Network Intrusion Detection System Based on Machine Learning for Software Defined Networks*. https://doi.org/10.3390/fi

Arcos-Argudo, M., Bojorque, R., & Torres, A. (2025). A Deterministic Comparison of Classical Machine Learning and Hybrid Deep Representation Models for Intrusion Detection on NSL-KDD and CICIDS2017. *Algorithms*, *18*(12). https://doi.org/10.3390/a18120749

Bakro, M., Kumar, R. R., Alabrah, A. A., Ashraf, Z., Bisoy, S. K., Parveen, N., Khawatmi, S., & Abdelsalam, A. (2023). Efficient Intrusion Detection System in the Cloud Using Fusion Feature Selection Approaches and an Ensemble Classifier. *Electronics (Switzerland)*, *12*(11). https://doi.org/10.3390/electronics12112427

Chen, Z., Yu, W., & Zhou, L. (2021). ADASYN-Random Forest Based Intrusion Detection Model. *Proceedings of the 2021 4th International Conference on Signal Processing and Machine Learning*, 152–159. https://doi.org/10.1145/3483207.3483232

Chua, T.-H., & Salam, I. (2022). *Evaluation of Machine Learning Algorithms in Network-Based Intrusion Detection System*. http://arxiv.org/abs/2203.05232

Maseer, Z. K., Yusof, R., Bahaman, N., Mostafa, S. A., & Foozy, C. F. M. (2021). Benchmarking of Machine Learning for Anomaly Based Intrusion Detection Systems in the CICIDS2017 Dataset. *IEEE Access*, *9*, 22351–22370. https://doi.org/10.1109/ACCESS.2021.3056614

Mhawi, D. N., Aldallal, A., & Hassan, S. (2022). Advanced Feature-Selection-Based Hybrid Ensemble Learning Algorithms for Network Intrusion Detection Systems. *Symmetry*, *14*(7). https://doi.org/10.3390/sym14071461

Mondragon, J. C., Branco, P., Jourdan, G. V., Gutierrez-Rodriguez, A. E., & Biswal, R. R. (2025). Advanced IDS: a comparative study of datasets and machine learning algorithms for network flow-based intrusion detection systems. *Applied Intelligence*, *55*(7). https://doi.org/10.1007/s10489-025-06422-4

*name of corresponding author

This is anCreative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

e-ISSN : 2541-2019
p-ISSN : 2541-044X

Oluwakemi, O. O., Abdullahi, M. U., & Anyachebelu, K. T. (2023). Comparative Evaluation of Machine Learning Algorithms for Intrusion Detection. *Asian Journal of Research in Computer Science*, *16*(4), 8–22. https://doi.org/10.9734/ajrcos/2023/v16i4366

Rodríguez, M., Alesanco, Á., Mehavilla, L., & García, J. (2022). Evaluation of Machine Learning Techniques for Traffic Flow-Based Intrusion Detection. *Sensors*, *22*(23). https://doi.org/10.3390/s22239326

Samantaray, M., Barik, R. C., & Biswal, A. K. (2024). A comparative assessment of machine learning algorithms in the IoT-based network intrusion detection systems. *Decision Analytics Journal*, *11*. https://doi.org/10.1016/j.dajour.2024.100478

Singh, A., Prakash, J., Kumar, G., Jain, P. K., & Ambati, L. S. (2024). Intrusion Detection System: A Comparative Study of Machine Learning-Based IDS. *Journal of Database Management*, *35*(1). https://doi.org/10.4018/JDM.338276

Thockchom, N., Singh, M. M., & Nandi, U. (2023). A novel ensemble learning-based model for network intrusion detection. *Complex and Intelligent Systems*, *9*(5), 5693–5714. https://doi.org/10.1007/s40747-023-01013-7

Tripathy, S. S., & Behera, B. (2023). *PERFORMANCE EVALUATION OF MACHINE LEARNING ALGORITHMS FOR INTRUSION DETECTION SYSTEM*. https://doi.org/10.17605/OSF.IO/WX6CS

Udurume, M., Shakhov, V., & Koo, I. (2024). Comparative Analysis of Deep Convolutional Neural Network—Bidirectional Long Short-Term Memory and Machine Learning Methods in Intrusion Detection Systems. *Applied Sciences (Switzerland)*, *14*(16). https://doi.org/10.3390/app14166967

*name of corresponding author

This is anCreative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.