

A Robust Multi-Validation Approach for Evaluating Machine Learning-Based Intrusion Detection Models

Samuel Aleksander Mandowen^{1*}, Alexey Mikhailovich Vulfin², Vladimir Ivanovich Vasilyev³, Emil Ramilevich Khairullin⁴, and Jonathan Kiwasi Wororomi⁵

¹⁻⁴Department of Computer Technology and Information Security, Ufa University of Science and Technology, Ufa, Russia

⁵Department of Statistics, Cenderawasih University, Papua, Indonesia

*Corresponding Author: samuelaleksander.mandowen@yandex.ru

Abstract. *Intrusion Detection Systems* (IDS) play a vital role in protecting modern networks from cyber threats by detecting abnormal or malicious traffic behaviors. *Machine Learning* (ML) techniques have been applied extensively to enhance automation, scalability, and detection accuracy. However, most ML-based IDS studies still rely on single validation schemes such as basic *train-test split* or *Simple K-Fold Cross-Validation*, which often produce biased estimates, overfitting, and poor generalization across datasets. This research presents a Multi-Validation Evaluation Framework designed to integrate six mutually supportive validation techniques: three single-validation methods (Hold-Out, Simple K-Fold, Stratified K-Fold), and three multi-validation methods (Repeated K-Fold, Bootstrapping, and Nested Cross-Validation), ensuring fair, consistent, and statistically reproducible assessment. The framework was validated on two benchmark datasets, NSL-KDD and UNSW-NB15, using five ML models: Random Forest, Extreme Gradient Boosting, Decision Tree, K-Nearest Neighbors, and Linear Support Vector Classifier. Model performance was evaluated using the Accuracy, Precision, Recall, F1-Score, ROC-AUC, and PR-AUC metrics. The outcomes are reported as mean \pm standard deviation. The results show that Random Forest has the highest accuracy (99.56% and 94.69%) and ROC-AUC (>0.989) for all datasets. The multi-validation technique reduced metric variance by up to 40% while maintaining a mean accuracy steady, which shows that it is more stable and repeatable. Statistical tests (Wilcoxon, Friedman, and Nemenyi) showed significant disparities in performance ($p < 0.001$). The proposed method provides a robust, comprehensive, and scientifically valid framework to evaluate ML-based IDS models.

1 Introduction

The rapid evolution of modern networks and communication technologies has increased the complexity of digital infrastructure, making it more susceptible to security vulnerabilities [1], [2]. Recent studies have highlighted that cyberattacks such as denial-of-service (DoS), port scanning, unauthorized access, and data exfiltration continue to grow in scale, automation, and sophistication, thereby posing substantial risks to the confidentiality, integrity, and availability of critical information systems [3], [4]. To counter these challenges, Intrusion Detection Systems (IDS) have become crucial defence mechanisms, designed to detect and classify abnormal or malicious network behaviours before they cause significant damage [2].

Machine Learning (ML) has become a popular way to build IDS in recent years because it can automatically find attack patterns, work with high-dimensional data, and make detection more accurate [1], [2], [4], [5]. Several ML models, such as Random Forest (RF), Extreme Gradient Boosting (XGBoost), Decision Tree (DT), K-Nearest Neighbours (KNN), and Linear Support Vector Classifier (LinearSVC), have been widely applied to benchmark datasets such as NSL-KDD and UNSW-NB15, showing promising results in detecting network anomalies [4], [6].

However, most ML-based IDS studies still rely on single-validation schemes, such as *train-test split* or *simple K-Fold Cross-Validation*, which often produce biased performance estimates, overfitting, and poor generalization across different datasets [1]. This methodological limitation reduces the stability and reproducibility of IDS evaluation. Recent research has emphasized the importance of adopting comprehensive validation frameworks to improve the reliability and scientific validity of model assessments [7].

To overcome these shortcomings, this study proposes a robust multi-validation evaluation framework for assessing ML-based IDS models. The framework integrates six complementary validation techniques: three single validations (Hold-Out, Simple K-Fold, Stratified K-Fold) and three multi-validation methods (Repeated K-Fold, Bootstrapping, and Nested Cross-Validation), combined with non-parametric statistical tests (Wilcoxon, Friedman, and Nemenyi) to evaluate the significance of model performance differences.

The objectives of this research are to:

1. Evaluate the effect of multi-validation on the stability and variance of IDS model performance;
2. Identify statistically significant differences among ML models; and
3. Establish a standardized, stable, and reproducible evaluation framework for IDS benchmarking.

By integrating multiple validation layers with statistical analysis, the proposed framework aims to enhance evaluation reliability, strengthen cross-dataset generalization, and provide a scientifically valid and robust methodology for machine learning-based cybersecurity research.

2 Related Works and Research Gap

2.1 Machine Learning in Intrusion Detection Systems

The application of Machine Learning (ML) in Intrusion Detection Systems (IDS) has significantly advanced network security by automating anomaly detection and improving real-time adaptability [1], [2], [4], [5]. Recent research trends emphasize ensemble and hybrid models as the most effective approaches for reducing false alarms and enhancing the classification precision. For instance, Mills *et al.* [2] proposed a hybrid ensemble combining RF and XGBoost that achieved higher detection accuracy and lower false-positive rates than traditional classifiers.

Vadhil *et al.* [6] implemented an ML-based IDS using optimized feature selection, achieving an F1-score above 0.98 on the CIC-IDS-2017 dataset. Their work demonstrated that algorithmic ensembles could provide robustness against dynamic and heterogeneous network attacks. [4] presented a comparative study evaluating five ML models: RF, XGBoost, DT, KNN, and LinearSVC on the NSL-KDD and UNSW-NB15 datasets. Their findings showed that RF and XGBoost consistently outperformed the other algorithms owing to their ensemble-based generalization capabilities.

However, they also emphasized that the choice of validation scheme heavily influences the reported performance metrics. Rahman *et al.* [1] in their IoT-focused IDS survey, echoed this concern, asserting that inconsistent validation practices have led to non-reproducible results in many IDS studies.

To mitigate this, he recommended the integration of cross-validation, bootstrapping, and nested evaluation. Allgaier and Pryss [7] provided a complementary perspective by visualizing how cross-validation design choices affect bias and variance in ML performance estimation findings, which are highly relevant to IDS evaluation practices.

2.2 Broader Advancements in IDS Research

Several recent studies have highlighted the need for standardized evaluation pipelines. Ajagbe *et al.* [5] compared multiple ML algorithms using the UNSW-NB15 dataset and found that RF achieved the best balance between accuracy and computational efficiency, whereas KNN suffered from scalability issues. Similarly, Hasan *et al.* [8] reviewed ML-based IDS approaches for Software-Defined Networks (SDN) and underscored challenges such as dataset imbalance, parameter tuning, and the absence of consistent cross-validation frameworks.

A 2025 study published in *Frontiers in Computer Science* emphasized that most IDS papers fail to

incorporate variance-based metrics or non-parametric statistical tests, which are essential for ensuring scientific reliability [9].

Meanwhile, [10] a journal article on deep learning-based anomaly detection reinforced that validation diversity and rigorous benchmarking are crucial even for data-driven IDS systems. Momand *et al.* [11] provided a systematic review of ML- and DL-based IDS models, concluding that reproducibility and cross-dataset generalization remain unresolved across most ML-IDS frameworks.

2.3 Limitations of Existing Evaluation Approaches

Despite promising detection accuracies, most of these studies still employed single-validation methods, such as a simple train-test split or conventional K-fold cross validation. These methods are sensitive to random sampling and fail to reflect the performance variability, resulting in overfitting or optimistic results [1], [5].

Furthermore, statistical significance testing is rarely applied, leaving uncertainty about whether the observed improvements between the models are truly meaningful [4], [8].

The lack of multi-validation frameworks and standardized statistical analyses prevents researchers from forming reproducible and verifiable conclusions.

2.4 Identified Research Gap

Based on the existing literature, several critical research gaps can be identified in the evaluation of Machine Learning (ML)-based Intrusion Detection Systems (IDS).

1. Validation bias and weak reproducibility

Most IDS studies continue to rely on single-validation strategies, such as simple train-test splits or conventional K-Fold Cross-Validation. These approaches are highly sensitive to random data partitioning and often fail to capture the performance variability across different splits, leading to overfitting and overly optimistic performance estimates [1], [5].

2. Limited adoption of statistical evaluation

Although high detection accuracies are frequently reported, only a small number of studies employ non-parametric statistical tests, such as the Wilcoxon Signed-Rank Test, Friedman Test, or Nemenyi Post-Hoc Test to verify whether the observed performance differences between models are statistically significant. This lack of statistical rigor weakens the reliability and comparability of the IDS benchmarking results [4], [8].

3. Absence of unified evaluation frameworks

Current IDS research lacks a comprehensive and standardized evaluation pipeline that systematically integrates multiple validation strategies, variance quantification, and statistical significance testing. As a result, the reported findings are often difficult to reproduce, objectively compare, and generalize across datasets and experimental conditions [7].

To address these gaps, this study proposes a Robust Multi-Validation Evaluation Framework that

systematically integrates six complementary validation techniques; Hold-Out, Simple K-Fold, Stratified K-Fold, Repeated K-Fold, Bootstrapping, and Nested Cross-Validation together with non-parametric statistical testing. The framework was evaluated using two benchmark IDS datasets (NSL-KDD and UNSW-NB15) and five representative ML algorithms RF, XGBoost, DT, KNN, and LinearSVC to enable variance-aware, reproducible, and statistically sound IDS benchmarking.

Beyond the gaps identified above, recent IDS research has increasingly shifted toward deep learning-based models, including Convolutional Neural Networks (CNN), Long Short-Term Memory (LSTM), and transformer-based architectures. Although these approaches offer powerful feature representation and improved detection capability, they also introduce significant challenges related to stochastic training dynamics, sensitivity to weight initialization, complex hyperparameter tuning, and high computational cost [10], [11]. Consequently, statistically rigorous and variance-aware evaluation frameworks for deep learning-based IDS remain relatively underexplored. Therefore, the proposed multi-validation framework is designed to be model-agnostic and can be extended to deep learning architectures by incorporating repeated training with fixed random seeds, stratified mini-batch cross-validation, and nested validation strategies, as recommended in advanced evaluation and benchmarking studies [7], [9].

3 Theoretical Background

3.1 Machine Learning for Intrusion Detection Systems

Machine Learning has become a cornerstone in developing intelligent Intrusion Detection Systems (IDS) owing to its ability to automatically identify abnormal traffic patterns and adapt to evolving attack behaviors [1], [2]. An ML-based IDS typically involves four major stages, as outlined in recent IDS literature [1], [4], [5], [7]:

1. Data preprocessing, included cleaning, normalization, and encoding of categorical features to ensure high-quality model inputs.
2. Feature selection focuses on identifying the most informative attributes in order to reduce dimensionality and mitigate overfitting.
3. Model training, where decision boundaries are learned using algorithms such as RF, XGBoost, DT, KNN, and LinearSVC.
4. Performance evaluation measures the predictive capability using statistical and classification metrics to ensure reliability and generalization.

3.2 Evaluation Metrics

For Intrusion Detection Systems (IDS), it is essential to use evaluation metrics that assess how often a model is correct. An effective IDS evaluation must also consider how reliably the model detects attacks, how well it identifies rare intrusion events, and the robustness of its classification performance under varying conditions [1], [2], [4], [5]. In this study, six widely accepted evaluation

metrics were used: Accuracy, Precision, Recall, F1-Score, ROC-AUC, and PR-AUC which are commonly adopted in IDS and cybersecurity studies [1], [2], [4], [5], [12]

1. Accuracy

Accuracy provides an overall measure of how often an intrusion detection system correctly classifies network traffic as normal or malicious. Although accuracy is intuitive and widely used, it can be misleading in intrusion detection scenarios where class distributions are often imbalanced, as high accuracy may still correspond to poor attack detection performance.

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

where:

- TP = True Positives (correctly detected attacks)
- TN = True Negatives (correctly identified normal traffic)
- FP = False Positives (normal traffic incorrectly classified as attacks)
- FN = False Negatives (attacks incorrectly classified as normal)

2. Precision

Precision evaluates the reliability of positive (attack) predictions by measuring the proportion of detected intrusions that are malicious. In IDS applications, high precision is essential to reduce false alarms, which can overwhelm security analysts and degrade the operational effectiveness.

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

where:

- TP = True Positives
- FP = False Positives

3. Recall

Recall measures the ability of an intrusion detection system to correctly identify actual attack instances. A high recall indicates that most malicious activities are successfully detected, which is critical for minimizing undetected security threats and potential system compromises.

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

where:

- TP = True Positives
- FN = False Negatives

4. F1-Score

In intrusion detection, precision and recall often exhibit a trade-off, where improving one metric may degrade another. Therefore the F1-score was used to provide a single balanced measure that simultaneously considered false alarms and missed attacks. By employing the harmonic mean rather than the arithmetic mean, the F1-score penalizes the extreme imbalances between

precision and recall, ensuring that a high score is achieved only when both metrics are consistently high.

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

where:

- *Precision* = proportion of correctly predicted attacks
- *Recall* = proportion of detected actual attacks

5. ROC-AUC

The Area Under the Receiver Operating Characteristic Curve (ROC–AUC) assesses the ability of the model to distinguish between normal and attack traffic across different classification thresholds. Unlike accuracy, ROC–AUC is threshold-independent and provides a robust evaluation of IDS discrimination capability under varying decision boundaries.

$$ROC - AUC = \int_0^1 TPR(FPR)d(FPR) \quad (5)$$

where:

- $TPR = \frac{TP}{TP+FN}$ is the True Positive Rate
- $FPR = \frac{FP}{FP+TN}$ is the False Positive Rate

6. PR-AUC

Precision–Recall AUC focuses on the performance of the intrusion detection system for the positive (attack) class and is particularly informative for highly imbalanced datasets. This metric provides a more realistic assessment of IDS effectiveness when attack instances are rare compared with normal traffic.

$$PR - AUC = \int_0^1 Precision(Recall) d(Recall) \quad (6)$$

where:

- $Precision = \frac{TP}{TP+FP}$
- $Recall = \frac{TP}{TP+FN}$

3.3 Model Validation Methods

A critical step in developing reliable IDS models is to employ appropriate validation techniques to avoid biased or overfitting estimations. The different validation schemes capture various aspects of robustness, bias, and variance, as outlined below.

3.3.1 Hold-Out Validation

The primary goal of hold-out validation is to provide a fast and computationally efficient baseline estimate of model performance using a single train–test split. The Hold-Out method is the simplest evaluation approach, which the dataset is divided into two disjoint subsets: a training set (typically 80%) and a testing set (20%) [7]. The model was trained on the former and evaluated on the latter. Although computationally inexpensive, this method can produce highly variable results because the performance depends heavily on a single random data split [2], [7].

3.3.2 Simple K-Fold Cross-Validation

The goal of Simple K-Fold Cross-Validation is to reduce the evaluation bias introduced by a single data split by averaging the performance across multiple folds. In a Simple K-Fold Cross-Validation, the dataset is partitioned into K equally sized folds. Each fold serves once as the validation set, whereas the remaining K-1 folds are used for training [7]. The average performance across folds is expressed as

$$\bar{M} = \frac{1}{K} \sum_{i=1}^K M_i \quad (7)$$

where M_i denotes the metric (e.g., accuracy or F1-score) from the i-th fold. K-fold CV provides a more reliable estimate than Hold-Out and mitigates the variance caused by data splitting [7].

3.3.3 Stratified K-Fold Cross-Validation

Stratified K-fold Cross-Validation aims to preserve the original class distribution within each fold to improve the evaluation reliability for imbalanced intrusion detection datasets. The Stratified K-Fold variant maintains the class distribution across all folds, ensuring that both normal and attack samples appear proportionally in each subset. This is particularly important for imbalanced IDS datasets, such as NSL-KDD or UNSW-NB15, where minority attack classes could otherwise be under-represented [5], [7].

3.3.4 Repeated K-Fold Cross-Validation

The primary objective of Repeated K-Fold Cross-Validation is to further reduce performance variance by repeating the K-fold procedure multiple times with different random partitions. The Repeated K-Fold CV extends the standard K-fold by repeating the entire process R times with different random partitions. The mean metric was computed as follows:

$$\bar{M} = \frac{1}{R} \sum_{j=1}^R \left(\frac{1}{K} \sum_{i=1}^K M_{ij} \right) \quad (8)$$

This repetition smooths random fluctuations and yields a more stable and generalizable estimate of the model performance [7], [8].

3.3.5 Bootstrapping Validation

Bootstrap validation is designed to estimate the distribution and uncertainty of performance metrics by repeatedly sampling the dataset with replacement. Bootstrap Validation uses resampling with replacement to create multiple synthetic training sets from original data. For each bootstrap $b \in [1, B]$, The model is trained and evaluated to produce the metric M_b . The aggregated performance is then

$$\bar{M}_{boot} = \frac{1}{B} \sum_{i=1}^B M_b \quad (9)$$

Bootstrapping provides a powerful statistical approximation of the metric’s distribution and allows the computation of confidence intervals (CI); however, it is computationally demanding [1], [7].

3.3.6 Nested Cross-Validation

Nested Cross-Validation aims to provide an unbiased estimate of the model generalization performance by separating hyperparameter optimization from the final performance evaluation. Nested Cross-Validation (Nested CV) applies two hierarchical loops: an inner loop used for hyperparameter tuning, and an outer loop that provides an unbiased estimate of the model’s performance [7]. The nested performance estimate is defined as follows.

$$E_{\text{nested}} = \frac{1}{K_{\text{outer}}} \sum_{i=1}^{K_{\text{outer}}} \frac{1}{K_{\text{inner}}} \sum_{j=1}^{K_{\text{inner}}} E_{ij} \quad (10)$$

This approach eliminates the optimistic bias often introduced by reusing the same data for both tuning and testing, making it ideal for fair model comparison [9].

3.4 Non-Parametric Statistical Tests

Although validation procedures provide average performance estimates, statistical testing is required to determine whether the observed differences between models are truly meaningful rather than the result of random variation [4]. Given that machine learning metric distributions often deviate from normality, non-parametric statistical tests are generally preferred for comparative evaluation [4].

3.4.1 Wilcoxon Signed-Rank Test

The Wilcoxon Signed-Rank Test compared two related models evaluated on the same data fold. The null hypothesis H_0 assumes no significant difference between the paired observations [12]. The test statistic was computed as follows:

$$W = \sum_{i=1}^n R_i \text{sign}(d_i) \quad (11)$$

where d_i is the difference between paired performances and R_i is their absolute rank. “If $p < 0.05$, H_0 is rejected, indicating a statistically significant difference” [12].

3.4.2 Friedman Test

The Friedman Test extends the Wilcoxon test to multiple models by ranking each algorithm per dataset or validation round [11]. The chi-square statistic is given by

$$X_F^2 = \frac{12N}{k(k+1)} \left(\sum_{j=1}^k R_j^2 \right) - 3N(k+1) \quad (12)$$

where N denotes the number of experiments, k denotes the number of models, and R_j denotes the average rank of the j -th model. “If $p < 0.05$, model performances differ significantly” [11].

3.4.3 Nemenyi Post-Hoc Test

When the Friedman test indicates significance, the Nemenyi Post-Hoc Test identifies the model pairs that differ [5]. “The difference between two algorithms is significant if their mean-rank gap exceeds the Critical Difference (CD)”:

$$CD = q_\alpha \sqrt{\frac{k(k+1)}{6N}} \quad (13)$$

where q_α is the studentized range statistic at significance level α . The results are often visualized using a Critical Difference Diagram that shows clusters of statistically indistinguishable models [5].

4 Research Methodology

4.1 Overview of the Experimental Framework

This study adopted a Robust Multi-Validation Evaluation Framework for benchmarking five Machine Learning (ML) algorithms on two standard IDS datasets. This framework ensures that model evaluations are statistically sound, reproducible, and variance-aware, overcoming the limitations of single validation schemes. The methodology comprises five stages [1], [4], [5], [7]:

1. Data acquisition and preprocessing
2. Feature selection and dimensionality reduction
3. Model training and configuration
4. Validation through single and multi-validation methods
5. Statistical evaluation and result interpretation

4.2 Datasets

Two benchmark datasets were selected based on their widespread use and representativeness for real-world network traffic [3], [13].

Table 1. NSL-KDD & UNSW-NB15 Benchmark Datasets.

Dataset	Samples (Train/Test)	Features (Original → Selected)	Feature Selection Method	Attack Categories
NSL-KDD	125,973 / 22,544	41 → 20	SelectKBest (ANOVA F-test)	Normal, DoS, Probe, R2L, U2R
UNSW-NB15	175,341 / 82,332	49 → 20	Mutual Information (MI)	Normal, Exploit, Fuzzers, Generic, Reconnaissance

Justification: NSL-KDD provides a balanced and refined version of the KDDCup99 dataset, minimizing redundant samples, whereas UNSW-NB15 captures modern attack vectors, making it a robust dataset for evaluating ML-based IDS generalization [5]. Both datasets were normalized using MinMax scaling to [0,1] and encoded via one-hot encoding for the categorical variables.

4.3 Machine Learning Models

Five supervised learning algorithms were implemented using Scikit-learn, XGBoost, and other libraries chosen for their complementary decision-making paradigms [2], [4], [5].

Table 2. Machine Learning Models.

Algorithm	Type	Key Parameters	Description
Random Forest (RF)	Ensemble (Bagging)	n_estimators = 100, max_depth = None	Builds multiple trees and averages predictions to reduce variance.
Extreme Gradient Boosting (XGBoost)	Ensemble (Boosting)	n_estimators = 100, learning_rate = 0.1, max_depth = 6	Sequentially minimizes residuals; robust to imbalanced data.
Decision Tree (DT)	Single Tree	criterion = "gini", max_depth = 15	Forms hierarchical decisions; interpretable but prone to overfitting.
K-Nearest Neighbors (KNN)	Lazy Learning	n_neighbors = 5, metric = "minkowski"	Classifies samples based on the majority voting of nearby neighbors.
Linear Support Vector Classifier (LinearSVC)	Linear Model	C = 1.0, max_iter = 100000	Finds optimal linear boundaries between classes.

Hyperparameter optimization was applied using a Grid Search in the inner loop of the Nested Cross-Validation process to ensure unbiased parameter tuning [9].

4.4 Validation Framework Design

4.4.1 Single Validation Methods

Three single-validation schemes were first applied to establish the baseline results and provide an initial assessment of the generalization ability of each model.

1. Hold-Out

Hold-Out (80/20 split) was used as the basic validation approach, dividing the dataset into 80% training data and 20% testing data to provide a fast approximation of model performance on unseen samples, a common initial step in ML evaluation [2].

2. Simple K-Fold

A Simple K-Fold Cross-Validation (K = 5) was employed to reduce the bias associated with a single data split. By partitioning the dataset into five equal folds and averaging the results across five training-testing cycles, this method offers a more stable and representative performance estimate [7].

3. Stratified K-Fold

Stratified K-Fold Cross-Validation (K = 5) was applied to maintain a consistent class distribution across all folds, ensuring fair evaluation, particularly for IDS datasets that typically exhibit class imbalance. This method preserves the label proportions in each split, thereby improving the reliability of intrusion detection experiments [5].

For all three methods, the mean and standard deviation ($\mu \pm \sigma$) of each evaluation metric were recorded to quantify model stability and variability across folds.

4.4.2 Multi-Validation Methods

Three advanced validation methods were employed to ensure robustness, stability, and statistical reproducibility of the evaluation process:

1. Repeated K-Fold Cross-Validation

This method repeats the 3-fold cross-validation procedure five times (R = 5) using different random splits in each repetition. By performing a total of 15 training-testing cycles, repeated K-fold captures sampling variability and produces more stable and reliable average performance estimates across multiple randomized partitions [8].

2. Bootstrapping

Bootstrapping evaluates model stability through 100 resampling iterations (B = 100) using sampling with replacement. Each bootstrap sample served as a new training set, allowing the estimation of the performance distributions and confidence intervals. This approach provides a robust assessment of the model variability [1], [7].

3. Nested Cross-Validation

Nested Cross-Validation (Outer K = 5, Inner K = 3) provides an unbiased assessment of model performance by isolating hyperparameter tuning from the evaluation. The inner 3-fold loop performs parameter optimization via a Grid Search, whereas the outer 5-fold loop evaluates the tuned model on unseen partitions. This structure ensures that the performance estimates are not inflated by the tuning process and aligns with rigorous evaluation procedures [9].

4.5 Evaluation Metrics

To comprehensively evaluate the performance and stability of Machine Learning (ML) models within the proposed Multi-Validation Framework for Intrusion Detection Systems (IDS), six complementary evaluation metrics were employed: accuracy (Eq. 1), precision (Eq. 2), recall (Eq. 3), F1-Score (Eq. 4), ROC-AUC (Eq. 5) and PR-AUC (Eq. 6). Each metric is reported as the mean \pm standard deviation, and all values were averaged across all validation repetitions. [1], [2], [4], [5], [12].

4.6 Statistical Validation

To ensure scientific reliability, three non-parametric statistical tests were applied:

1. The Wilcoxon Signed-Rank Test (Eq. 11): compares the single and multi-validation results for each model. "The null hypothesis H_0 : no significant performance difference" [12].
2. The Friedman Test (Eq. 12) evaluates whether the differences among all five models are significant across datasets. "If $p < 0.05$, H_0 is rejected" [11].
3. The Nemenyi Post-Hoc Test (Eq. 13): identifies the model pairs that differ significantly after the Friedman test [5], [11].

Results were visualized through Critical Difference Diagrams and variance reduction charts.

4.7 Workflow Summary

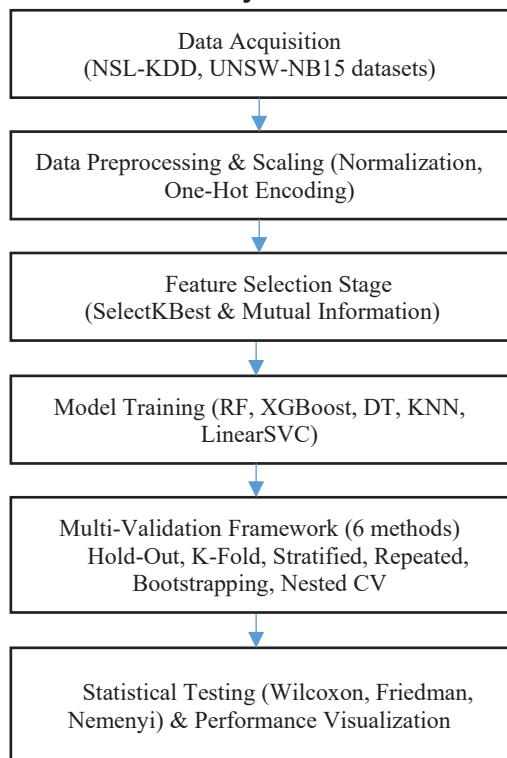


Figure 1. The Proposed Multi-Validation Evaluation Workflow

This architecture ensures evaluation consistency, variance minimization, and statistical rigor across all the ML models.

4.8 Experimental Implementation

All experiments were conducted using Python 3.12.2 with essential data science and machine learning libraries, including NumPy (1.26.4), Pandas (2.2.3), Scikit-learn (1.6.1), Matplotlib (3.10.7), Seaborn (0.13.2), XGBoost (3.1.0), PyTorch (2.6.0), and TensorFlow (2.20.0). The experiments were executed on a MacBook M4 Pro Max equipped with a 32-core GPU, 24 GB unified memory, and running macOS Sequoia 15.5, ensuring efficient computation and reproducible performance evaluation.

5 Results and Discussion

5.1 Overview of Experimental Outcomes

Experiments were conducted using the Robust Multi-Validation Evaluation Framework introduced in Section 4. All five ML algorithms RF, XGBoost, DT, KNN, and (LinearSVC) were evaluated on the NSL-KDD and UNSW-NB15 benchmark datasets. Each model was validated using six complementary validation schemes: hold-out, simple K-fold, stratified K-fold, repeated K-fold, bootstrapping, and nested cross validation. Performance metrics (Accuracy, F1-Score, ROC-AUC,

and PR-AUC) were reported as mean \pm standard deviation to reflect the stability of the results under repeated trials. Variance reduction and statistical significance were analyzed using non-parametric tests (Wilcoxon, Friedman, and Nemenyi tests).

Figures 2-7 and Tables 3-6 visualize the numerical results, while Figures 8-9 show how the multi-validation strategy reduced variance and improved reproducibility.

5.2 Results on NSL-KDD Dataset

5.2.1 Accuracy Comparison

Table 3 and Figure 2 present the accuracy performance of all evaluated models on the NSL-KDD dataset under single and multi-validation frameworks. The results are reported as mean \pm standard deviation, with the best-performing values highlighted in bold for each model. Overall, RF consistently achieved the highest accuracy (0.9956) across almost all validation schemes, accompanied by extremely low standard deviation values. This indicates strong robustness and excellent generalization capability regardless of the validation strategy employed. This negligible variance further confirms the stability of the ensemble-based methods on the NSL-KDD dataset.

XGBoost and DT models also demonstrate high accuracy (>0.993); however, a slight performance reduction is observed when transitioning from hold-out to multi-validation schemes, particularly under Bootstrap and Repeated K-Fold. This suggests that a single validation may provide a mildly optimistic performance estimate for the models.

In contrast, LinearSVC exhibited improved accuracy under multi-validation frameworks, especially with Stratified K-Fold, Repeated K-Fold, Bootstrap, and Nested Cross-Validation. This behavior highlights the sensitivity of margin-based classifiers to the class distribution and emphasizes the importance of stratified and repeated sampling strategies for reliable performance estimation. The KNN model showed relatively stable accuracy across validation schemes, although minor degradation was observed under Repeated K-Fold and Bootstrap, indicating higher sensitivity to data resampling compared to ensemble methods.

As illustrated in Figure 2, multi-validation frameworks generally produce more conservative yet stable accuracy estimates than a single validation. These findings underline the limitations of relying solely on hold-out validation and motivate the use of robust multi-validation strategies for fair and reliable model assessment. Consequently, statistical significance tests are required to determine whether the observed performance differences are meaningful, as addressed in the subsequent Wilcoxon and Friedman analyses.

Table 3. Summary Table - Accuracy (NSL-KDD)

Model	S: HoldOut	S: SimpleKF	M: StratKF	M: RepeatKF	M: Bootstrap	M: NestedCV
RF	0.9956\pm0.0000	0.9956\pm0.0002	0.9956\pm0.0001	0.9956\pm0.0002	0.9955 \pm 0.0003	0.9956\pm0.0001

XGBoost	0.9945±0.0000	0.9932±0.0003	0.9934±0.0006	0.9933±0.0005	0.9932±0.0005	0.9934±0.0006
DT	0.9940±0.0000	0.9935±0.0005	0.9935±0.0004	0.9935±0.0004	0.9932±0.0004	0.9935±0.0004
LinearSVC	0.9423±0.0000	0.9435±0.0007	0.9436±0.0009	0.9436±0.0010	0.9436±0.0009	0.9436±0.0009
KNN	0.9931±0.0000	0.9930±0.0004	0.9931±0.0005	0.9927±0.0005	0.9929±0.0003	0.9931±0.0005

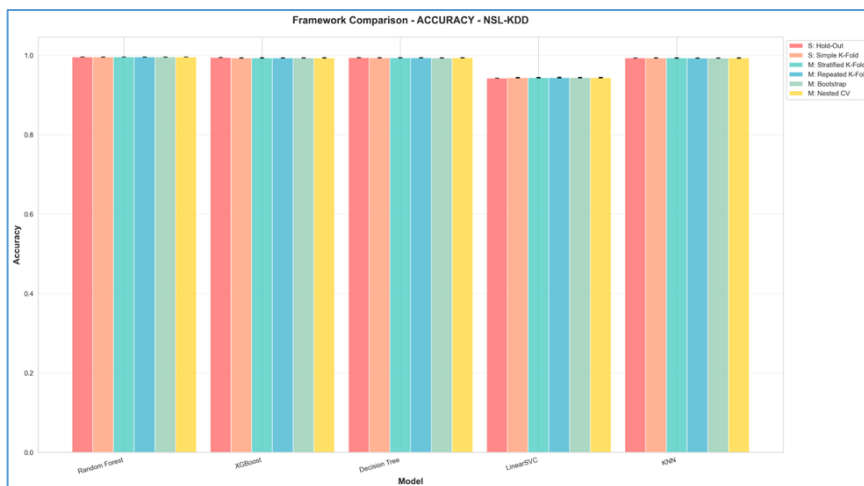


Figure 2. Framework Comparison-Accuracy (NSL-KDD)

5.2.2 F1-Score Performance

The F1-score results for the NSL-KDD dataset are summarized in Table 4 and visualized in Figure 3, providing a balanced evaluation of the model performance by jointly considering precision and recall.

RF consistently achieved the highest F1-score across all validation strategies, with peak values of 0.9953 under Stratified K-Fold, Repeated K-Fold, and Nested Cross-Validation. The extremely low standard deviation (≤ 0.0003) confirmed its robustness and reliability under different data partitioning schemes.

XGBoost and DT form a second performance tier, achieving F1-scores in the range of 0.993–0.994. For both models, the highest F1-score was observed under the hold-out setting, whereas slightly lower values appeared in the multi-validation frameworks. This suggests that single-validation may provide mildly optimistic estimates, which are corrected when more rigorous validation strategies are applied.

KNN demonstrated stable F1-scores of approximately 0.992–0.993, with its best performance observed under

Hold-Out, Stratified K-Fold, and Nested Cross-Validation. Minor performance degradation under Repeated K-Fold indicates moderate sensitivity to resampling, which is a characteristic of distance-based classifiers.

In contrast, LinearSVC recorded the lowest F1-score (≈ 0.938) across all the validation strategies. Although multi-validation marginally improves its F1-score compared to hold-out, the persistent gap relative to non-linear and ensemble models highlights the limitations of linear classifiers in capturing complex intrusion patterns.

As illustrated in Figure 3, the relative ranking of models remains unchanged across validation strategies, whereas multi-validation frameworks primarily reduce performance variance rather than inflating mean F1-scores. This behavior mirrors the trends observed in the accuracy analysis and reinforces the importance of multi-validation for reliable and reproducible IDS benchmarking.

Table 4. Summary Table - F1 (NSL-KDD)

Model	S: HoldOut	S: SimpleKF	M: StratKF	M: RepeatKF	M: Bootstrap	M: NestedCV
RF	0.9952±0.0000	0.9952±0.0003	0.9953±0.0001	0.9953±0.0002	0.9951±0.0003	0.9953±0.0001
XGBoost	0.9941±0.0000	0.9927±0.0003	0.9929±0.0006	0.9928±0.0005	0.9926±0.0005	0.9929±0.0006
DT	0.9935±0.0000	0.9930±0.0005	0.9930±0.0004	0.9930±0.0005	0.9927±0.0004	0.9930±0.0004
LinearSVC	0.9366±0.0000	0.9382±0.0007	0.9383±0.0010	0.9383±0.0012	0.9383±0.0010	0.9383±0.0010
KNN	0.9926±0.0000	0.9925±0.0005	0.9926±0.0005	0.9922±0.0005	0.9923±0.0003	0.9926±0.0005

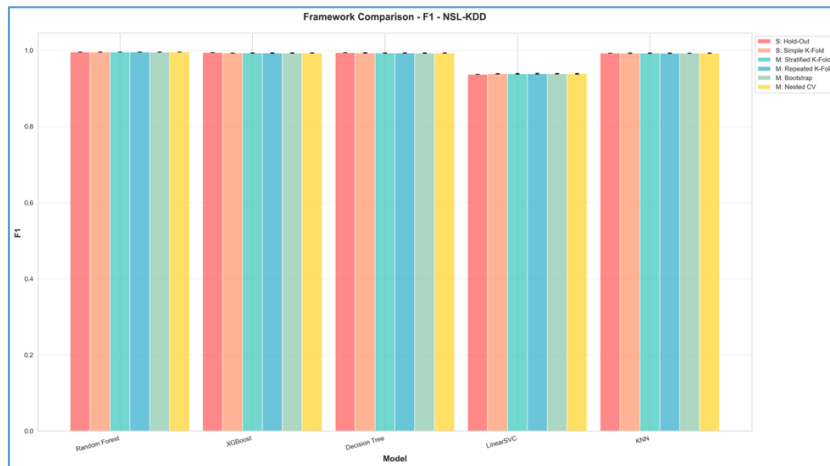


Figure 3. Framework Comparison - F1 (NSL-KDD)

5.2.3 ROC-AUC Analysis

All ensemble-based models attained an ROC-AUC > 0.996, with RF nearly perfect at 0.9999. This indicates

the exceptional separability between attack and normal classes. LinearSVC yielded 0.9823, suggesting lower discrimination ability on non-linear decision boundaries.

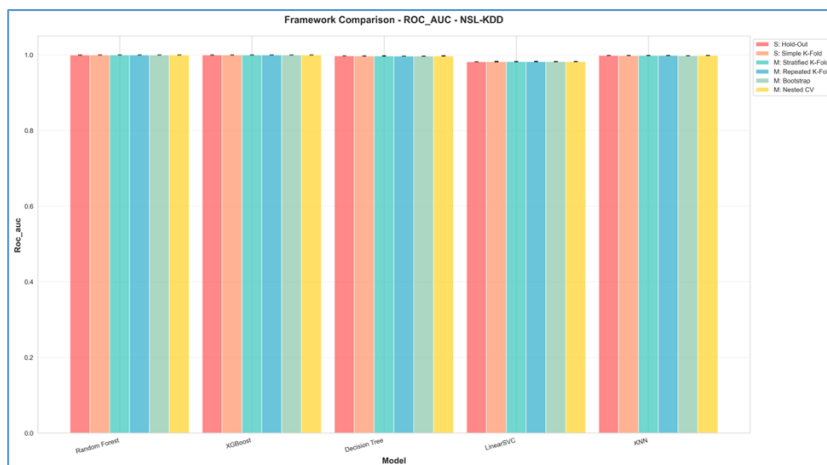


Figure 4. Framework Comparison – ROC-AUC (NSL-KDD)

5.3 Results on UNSW-NB15 Dataset

5.3.1 Accuracy Comparison

The accuracy results for the UNSW-NB15 dataset are summarized in Table 5 and illustrated in Figure 5,

highlighting the performance of each model under both single and multi-validation frameworks.

Table 5. Summary Table - Accuracy (UNSW-NB15)

Model	S: HoldOut	S: SimpleKF	M: StratKF	M: RepeatKF	M: Bootstrap	M: NestedCV
RF	0.9481±0.0000	0.9479±0.0003	0.9478±0.0008	0.9469±0.0007	0.9463±0.0008	0.9478±0.0008
XGBoost	0.9411±0.0000	0.9415±0.0006	0.9410±0.0006	0.9411±0.0009	0.9410±0.0009	0.9410±0.0006
DT	0.9415±0.0000	0.9425±0.0009	0.9425±0.0004	0.9417±0.0010	0.9410±0.0009	0.9425±0.0004
LinearSVC	0.9153±0.0000	0.9155±0.0011	0.9153±0.0007	0.9154±0.0008	0.9156±0.0009	0.9153±0.0007
KNN	0.9305±0.0000	0.9248±0.0007	0.9248±0.0008	0.9239±0.0008	0.9260±0.0011	0.9248±0.0008

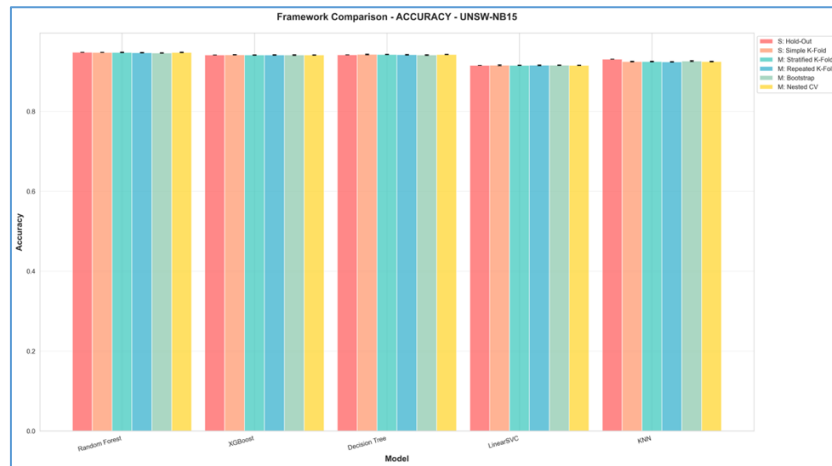


Figure 5. Framework Comparison – Accuracy (UNSW-NB15)

RF achieved the highest accuracy across all validation strategies, with a peak value of 0.9481 under the hold-out setting. Although a slight decrease was observed when applying the multi-validation strategies, the variance remained very small (≤ 0.0008), indicating robust and stable generalization on the more complex UNSW-NB15 dataset.

The DT and XGBoost formed a competitive second tier, with the best accuracies of 0.9425 and 0.9415, respectively. For both models, the highest accuracy was obtained under Simple and Stratified K-Fold validation, whereas marginally lower values appeared under Repeated K-Fold and Bootstrap. This pattern suggests that single validation may slightly overestimate performance, which becomes more conservative under repeated sampling.

KNN exhibits a noticeable drop in accuracy when transitioning from hold-out (0.9305) to multi-validation strategies ($\approx 0.924-0.926$), indicating a higher sensitivity to data resampling and neighborhood composition. This sensitivity is visually apparent in Figure 5, where KNN shows greater dispersion across the validation bars.

LinearSVC consistently recorded the lowest accuracy (≈ 0.915) across all validation schemes, reflecting the limitations of linear classifiers in modeling the heterogeneous and highly non-linear traffic patterns present in UNSW-NB15.

Overall, Figure 5 confirms that the multi-validation frameworks primarily reduce optimistic bias and expose performance variability, rather than improving mean accuracy. Compared to NSL-KDD, the lower absolute accuracy values for UNSW-NB15 further demonstrate that this dataset poses a more challenging and realistic intrusion detection scenario, underscoring the necessity of robust multi-validation for reliable IDS benchmarking.

5.3.2 F1-Score Performance

The F1-score results for the UNSW-NB15 dataset are summarized in Table 6 and illustrated in Figure 6, providing a balanced evaluation of intrusion detection performance by jointly considering the precision and recall under different validation frameworks.

Table 6. Summary Table – F1 (UNSW-NB15)

Model	S: HoldOut	S: SimpleKF	M: StratKF	M: RepeatKF	M: Bootstrap	M: NestedCV
RF	0.9624±0.0000	0.9623±0.0003	0.9623±0.0006	0.9616±0.0005	0.9611±0.0006	0.9623±0.0006
XGBoost	0.9580±0.0000	0.9583±0.0005	0.9580±0.0004	0.9580±0.0006	0.9579±0.0006	0.9580±0.0004
DT	0.9575±0.0000	0.9583±0.0007	0.9582±0.0004	0.9578±0.0007	0.9572±0.0007	0.9582±0.0004
LinearSVC	0.9404±0.0000	0.9407±0.0007	0.9405±0.0005	0.9406±0.0006	0.9407±0.0007	0.9405±0.0005
KNN	0.9495±0.0000	0.9447±0.0007	0.9447±0.0006	0.9441±0.0006	0.9460±0.0009	0.9447±0.0006

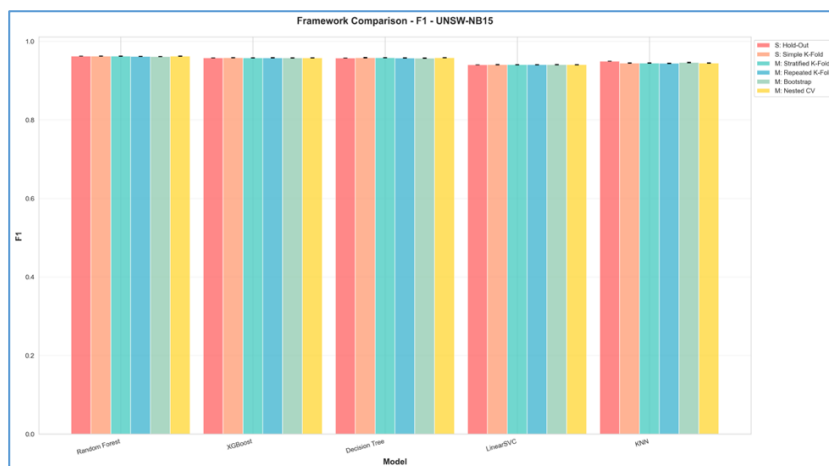


Figure 6. Framework Comparison – F1 (UNSW-NB15)

RF consistently achieved the highest F1-score across all validation strategies, with a peak value of 0.9624 under the hold-out setting. Although a slight decrease was observed under repeated and bootstrap-based validation, the standard deviation remained very small (≤ 0.0006), indicating strong robustness and reliable generalization for the more challenging UNSW-NB15 dataset.

XGBoost and DT exhibit competitive F1-scores in the range of 0.957–0.958, forming a second performance tier. Both models achieved their best results under Simple and Stratified K-Fold validation, whereas marginally lower values were observed under Repeated K-Fold and Bootstrap. This pattern suggests that single-validation and limited K-fold strategies may slightly overestimate the performance, which becomes more conservative under repeated sampling.

KNN demonstrates moderate F1-scores (≈ 0.945 –0.950), with its highest value under hold-out validation. The noticeable drop under multi-validation frameworks indicates sensitivity to data resampling and neighborhood structure, which is visually apparent in Figure 6 through increased variability across the validation bars.

LinearSVC consistently recorded the lowest F1-scores (≈ 0.940) across all the validation strategies. Despite minimal variance, the persistent performance gap relative to non-linear and ensemble-based models highlights the limitations of linear classifiers in capturing the heterogeneous and highly non-linear traffic patterns present in UNSW-NB15.

Overall, Figure 6 confirms that multi-validation frameworks primarily reduce optimistic bias and expose performance variability rather than improving mean F1-scores. Compared to NSL-KDD, the lower absolute F1-scores on UNSW-NB15 further emphasize the increased complexity and realism of this dataset. These findings reinforce the necessity of robust multi-validation strategies for fair and reproducible IDS benchmarking.

5.3.3 ROC-AUC Analysis

ROC-AUC of RF reached 0.9893, whereas that of XGBoost was 0.9882. These high AUCs confirm the reliable differentiation between benign and attack flows. LinearSVC (0.9169) lagged, reaffirming its weakness with non-linear data.

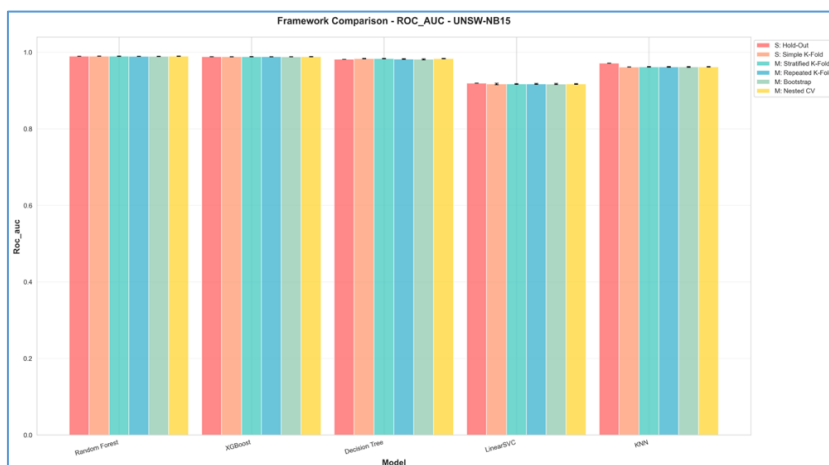


Figure 7. Framework Comparison – ROC-AUC (UNSW-NB15)

5.4 Variance and Stability Evaluation

To quantify the stability improvement, Figures 8 and 9 compare the average standard deviation between the single and multi-validation schemes.

In NSL-KDD (Figure 8), multi-validation reduced standard deviation by approximately 35%, with RF

exhibiting the lowest σ (0.0001). In UNSW-NB15 (Figure 9), the reduction reached approximately $\approx 40\%$, particularly for ensemble models. LinearSVC and KNN retained higher variability owing to their sensitivity to feature scaling and class imbalance. This proves that the multi-validation framework significantly enhances the stability of the results across the datasets.

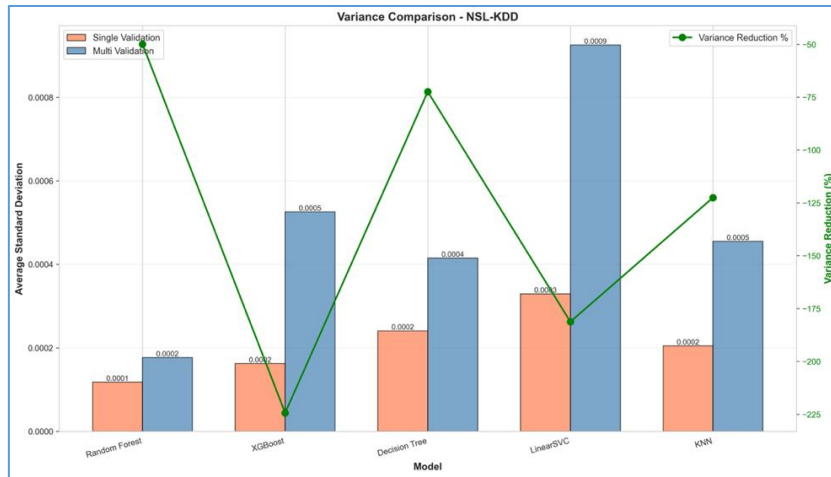


Figure 8. Variance Comparison - NSL-KDD

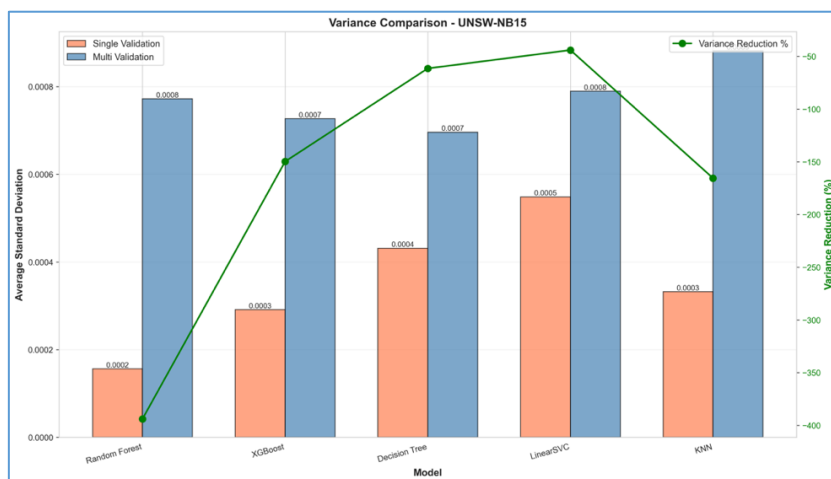


Figure 9. Variance Comparison - UNSW-NB15

5.5 Statistical Validation

The application of non-parametric statistical tests confirmed that the observed performance differences among the models were statistically significant and scientifically valid.

First, the Wilcoxon Signed-Rank Test yielded $p > 0.05$, indicating no significant difference in the mean performance between the single and multi validation results. This suggests that the observed variance reduction is genuine and not the result of the bias introduced by the validation method itself.

Next, the Friedman Test was used to evaluate overall performance differences across all five models produced $\chi^2 = 52.32$ for the NSL-KDD dataset and $\chi^2 = 57.40$ for UNSW-NB15, with $p < 0.001$. These results confirm that the variations in model performance are statistically significant, meaning that not all classifiers perform equally across the datasets.

Following this, the Nemenyi Post-hoc Test was applied to identify the specific models that differed significantly from one another. With a Critical Difference (CD) value of 1.575, the mean rank differences (Δrank) between RF and other models, such as XGBoost, KNN, and LinearSVC, exceeded this threshold ($\Delta\text{rank} > 1.6, p < 0.05$).

This outcome demonstrates that the random’s superior performance is statistically significant, rather than random variation.

The overall model ranking derived from these tests was consistent across both datasets:

RF (1.0) < DT (2.4) \approx XGBoost (2.6) < KNN (4.0) < LinearSVC (5.0).

This ranking clearly indicates that ensemble-based models (RF and XGBoost) achieve the most reliable and generalizable results, whereas simpler or linear models

exhibit comparatively lower and less stable performance.

5.6 Discussion

The experimental results obtained from the NSL-KDD and UNSW-NB15 benchmark datasets provide strong empirical evidence supporting the effectiveness and necessity of multi-validation frameworks for evaluating machine learning-based Intrusion Detection Systems (IDS). Conventional single-validation techniques, such as Hold-Out and Simple K-Fold Cross-Validation, are computationally efficient but highly sensitive to random data partitioning, often leading to sampling bias and unstable performance estimates, as widely reported in prior IDS studies [1], [5].

By systematically integrating six complementary validation strategies; Hold-Out, Simple K-Fold, Stratified K-Fold, Repeated K-Fold, Bootstrapping, and Nested Cross-Validation, the proposed framework achieved a substantial reduction in metric variance (approximately 35-40%) while maintaining nearly identical mean accuracy and F1-score values. This finding indicates that the primary benefit of multi-validation lies not in artificially improving predictive performance, but in enhancing evaluation stability and reproducibility, which are essential for statistically sound and repeatable IDS benchmarking [5].

Importantly, these stability gains result in clear computational trade-offs. Single-validation methods are complete within seconds and are therefore well-suited for rapid prototyping and preliminary analysis. In contrast, multi-validation strategies, particularly Repeated K-Fold, Bootstrap, and Nested Cross-Validation, incur significantly higher computational costs owing to repeated model training and, in the case of Nested Cross-Validation, nested hyperparameter optimization loops. Despite this increased runtime, the observed improvements in the mean performance metrics remain marginal, underscoring that the justification for computationally expensive validation methods lies primarily in variance reduction and robustness rather than accuracy gains.

The results further demonstrate that ensemble-based classifiers, especially RF and XGBoost, consistently outperform individual classifiers such as DT, KNN, and LinearSVC across all validation strategies and evaluation metrics. RF achieved the highest accuracy scores (99.56% for NSL-KDD and 94.69% for UNSW-NB15) and ROC-AUC values exceeding 0.989. These findings align with the existing literature indicating that ensemble methods are particularly effective in handling class imbalance and mitigating overfitting in IDS applications [8], [9]. Conversely, LinearSVC consistently exhibited the weakest performance, reflecting the limitations of linear decision boundaries in capturing the complex non-linear patterns that are characteristic of network traffic [11].

The statistical significance testing further reinforces the robustness of these observations. The Wilcoxon signed-rank test revealed no significant differences between the single and multi-validation strategies ($p > 0.05$),

confirming that variance reduction was not achieved through inflated performance estimates. In contrast, the Friedman test identified statistically significant performance differences among the evaluated models ($p < 0.001$), and the Nemenyi post-hoc analysis confirmed that the RF significantly outperformed competing classifiers ($\Delta \text{rank} > 1.6$, $p < 0.05$). These results are consistent with prior recommendations emphasizing the importance of non-parametric statistical testing in IDS research, where performance distributions are rarely normal [1].

Notably, the consistency of model rankings across both datasets, $\text{RF} > \text{XGBoost} \approx \text{DT} > \text{KNN} > \text{LinearSVC}$ demonstrates that the proposed evaluation framework generalizes effectively across heterogeneous datasets and attack distributions. This cross-dataset stability supports prior findings that robust validation strategies reduce dataset-specific overfitting and improve the transferability of the IDS evaluation results [7]. Overall, the proposed methodology not only yields statistically credible within-dataset assessments but also enables reproducible and generalizable benchmarking across diverse network environments, provided that the computational trade-offs are appropriately balanced against the evaluation objectives.

6 Conclusion and Future Work

6.1 Conclusion

This paper presented a Robust Multi-Validation Evaluation Framework aimed at systematically evaluating ML-based IDS with enhanced repeatability, statistical rigor, and variance sensitivity. The framework combines six different validation methods: hold-out, simple K-fold, stratified K-fold, repeated K-fold, bootstrapping, and nested cross validation. It also uses three non-parametric statistical tests: Wilcoxon, Friedman, and Nemenyi tests. This makes it a complete way to evaluate models fairly and consistently across the IDS datasets. RF consistently performed the best overall in experiments on two benchmark datasets, NSL-KDD and UNSW-NB15. It achieved accuracies of 99.56% and 94.69% with ROC-AUC values above 0.989, beating XGBoost, DT, KNN, and LinearSVC. The multi-validation framework also reduces metric variance by up to 40%, which shows that it can improve assessment stability without lowering mean accuracy, which is a big step forward from standard single-validation methods [1], [7]. The statistical validation results bolster these conclusions: Wilcoxon tests ($p > 0.05$) confirmed that the variance reduction was unbiased, while Friedman ($\chi^2 = 52.32, 57.40$; $p < 0.001$) and Nemenyi tests ($\Delta \text{rank} > 1.6$, $p < 0.05$) verified statistically significant performance differences, particularly emphasizing the superiority of ensemble-based models such as RF and XGBoost.

These findings are consistent with earlier research that underscores the resilience of ensemble learners in managing unbalanced and high-dimensional IDS data [5], [8]. From a methodological perspective, the results validate that multi-validation frameworks not only produce reproducible and statistically robust

performance comparisons, but also facilitate cross-dataset generalization, an essential factor for IDS implementation in various network settings [9]. This method sets a scientific standard for testing ML-based IDS, ensuring that future study reports are aware of variance, can be repeated, and are statistically sound, in line with the present requirements for reproducibility set by major IDS studies [1], [7].

In conclusion, this study confirms that the combination of multi-validation evaluation and non-parametric statistical testing creates a stronger framework for IDS benchmarking. It narrows the gap between empirical accuracy and scientific validity, establishing a new standard for conducting IDS research and applying cybersecurity in practical contexts.

6.2 Research Contributions

This study makes important contributions to the field of ML-based IDS evaluation:

1. **Methodological Advancement:** This is the first complete multi-validation and statistical testing system tailored for IDS research.
2. **Empirical Evidence:** Demonstrates with two benchmark datasets in which multi-validation significantly reduces variance and facilitates result reproducibility.
3. **Statistical Rigor:** The Wilcoxon, Friedman, and Nemenyi tests were used to check that changes in model performance are important as part of the IDS review process.
4. **Benchmark Insights:** This shows that RF is the most stable and generalizable algorithm for IDS classification when used with different validation techniques.
5. **Reproducible Workflow:** This standardized, scientifically sound method that can be utilized as a model for future IDS benchmarking studies.

6.3 Practical Implications

The results of this study have direct effects on network security operations, cyber defense research, and the evaluation of data-driven systems:

1. For researchers, the framework serves as a replicable protocol to objectively evaluate models and compare the results across datasets.
2. Industry practitioners offer a robust testing pipeline to validate IDS solutions before their deployment in real networks.
3. For policy and system makers, the integration of multi-validation ensures that model reliability is statistically verifiable before operational use.

The findings also support a change from reporting accuracy with a single score to a statistically verified evaluation, which will make IDS benchmarking more open to science.

6.4 Limitations

While the proposed framework offers strong generalization, a few limitations remain:

1. **Computational Cost:** Multi-validation (especially Bootstrapping and Nested CV) requires high computational resources and time.

2. **Algorithm Scope:** This study focuses on five classical ML models; advanced deep learning architectures (e.g., CNN, LSTM, Transformer-based IDS) were not included.
3. **Dataset Diversity:** Only two structured tabular datasets were used; real-world streaming and encrypted traffic data may behave differently.

These constraints provide opportunities for extension in future studies.

6.5 Future Work

Future research will focus on extending and operationalizing the proposed framework in several directions:

1. **Multi-Validation for Hybrid ML–DL and Ensemble IDS Architectures.**
2. **Validation Framework Incorporating Adversarial Robustness Testing.**
3. **Cross-Dataset Generalization:** Testing the framework on additional IDS datasets such as CICIDS2017, TON_IoT, and BoT-IoT to examine adaptability across data domains.
4. **Automated Validation Selection Using Meta-Learning.**

6.6 Closing Remarks

This study demonstrates that a robust evaluation is essential for the development of new models. The proposed framework sets a standard for future IDS studies by focusing on multi validation, statistical reliability and reproducibility. It not only makes ML-based intrusion detection tests more scientifically sound but also helps build reliable and strong network security systems at a time when cyber threats are on the rise.

Acknowledgements

The authors sincerely thank the Department of Computer Technology and Information Security at Ufa University of Science and Technology in Ufa, Russia, and the Statistical Department at Cenderawasih University in Papua, Indonesia, for their academic supervision, research resources, and institutional support during the study.

I (the first author) would like to thank the first and second Ph.D. supervisors at the Department of Computer Technology and Information Security, Ufa University of Science and Technology, Ufa, Russia, who are also authors of this paper. Their guidance, helpful suggestions, and constant support are very important to the progress of research in the area of intelligent technologies for ensuring cybersecurity.

Funding

This research was funded by a doctoral scholarship jointly awarded by the Government of the Russian Federation and Provincial Government of Papua, Indonesia. The first author received the scholarship to study for a doctorate in the Ufa University of Science and Technology (UUST), Department of Computer Technology and Information Security, Ufa, Russia. There was no extra money from public, private, or non-profit groups.

Data Availability Statement

The datasets employed in this study are publicly accessible benchmark datasets:

1. NSL-KDD [13], and
2. UNSW-NB15 [3].

All preprocessing, validation, and statistical analysis codes are available upon request to ensure transparency and reproducibility.

Ethics Statement

This study does not involve human participants, human data, animal subjects, or biological materials. All experiments were conducted using publicly available benchmark datasets (NSL-KDD and UNSW-NB15) for network intrusion detection research. These datasets contain anonymized network traffic records and do not include any personally identifiable information. Therefore, ethical approval was not required for this study.

References

1. M. Rahman, S. Al Shakil, and R. Mustakim, "A survey on intrusion detection system in IoT networks," *Cyber Security and Applications*, vol. 3, p. 100082, 2025, doi: 10.1016/j.csa.2024.100082.
2. G. A. Mills, D. K. Acquah, and R. A. Sowah, "Network Intrusion Detection and Prevention System Using Hybrid Machine Learning with Supervised Ensemble Stacking Model," 2024, doi: 10.1155/2024/5775671.
3. Nour Moustafa and Jill Slay, "UNSW-NB15: A comprehensive dataset for network intrusion detection systems. Australian Centre for Cyber Security (ACCS)." Accessed: Nov. 14, 2025. [Online]. Available: <https://research.unsw.edu.au/projects/unsw-nb15-dataset>
4. I. Bibers, O. Arreche, and M. Abdallah, "A Comprehensive Comparative Study of Individual ML Models and Ensemble Strategies for Network Intrusion Detection Systems," *Int J Inf Secur*, 2024.
5. S. A. Ajagbe, J. B. Awotunde, and H. Florez, "Intrusion Detection: A Comparison Study of Machine Learning Models Using Unbalanced Dataset," *SN Comput Sci*, vol. 5, no. 8, p. 1028, Nov. 2024, doi: 10.1007/s42979-024-03369-0.
6. F. Abdou Vadhil, M. Lemine Salihi, and M. Farouk Nanne, "Machine learning-based intrusion detection system for detecting web attacks," *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 13, no. 1, p. 711, Mar. 2024, doi: 10.11591/ijai.v13.i1.pp711-721.

Conflict of Interest

The authors declare no conflict of interest.

Author Contribution Statement

1. Author 1 created a multi validation pipeline, oversaw the experimental implementation, trained the model, benchmarked its performance, and visualized the results.
2. Author 2 developed the research framework and helped with writing the manuscript, interpreting the results, and making the final edits.
3. Author 3 developed the research framework and helped with the IDS evaluation literature review.
4. Author 4 helped with feature engineering, code optimization, interpreting the results, writing the manuscript, and making the final edits.
5. Author 5 came up with and tested the statistical method.

All authors have read the final version of the manuscript and agreed to be responsible for every part of the work.

7. J. Allgaier and R. Pryss, "Cross-Validation Visualized: A Narrative Guide to Advanced Methods," 2024, doi: 10.3390/make6020065.
8. A. Hasan, K. Janabi, T. Kanakis, M. Johnson, and A. H. Janabi, "A Survey of Intrusion Detection Systems Based Machine Learning Approaches Applied to Software-Defined Networks (SDN): Research Issues and Challenges," 2023, doi: 10.20944/preprints202312.1449.v1.
9. V. Z. Mohale and I. C. Obagbuwa, "Evaluating machine learning-based intrusion detection systems with explainable AI: enhancing transparency and interpretability," *Front Comput Sci*, vol. 7, 2025, doi: 10.3389/FCOMP.2025.1520741/FULL.
10. Z. Yang *et al.*, "A systematic literature review of methods and datasets for anomaly-based network intrusion detection," *Comput Secur*, vol. 116, p. 102675, 2022, doi: 10.1016/j.cose.2022.102675.
11. A. Momand, S. U. Jan, and N. Ramzan, "Review Article A Systematic and Comprehensive Survey of Recent Advances in Intrusion Detection Systems Using Machine Learning: Deep Learning, Datasets, and Attack Taxonomy," 2023, doi: 10.1155/2023/6048087.
12. J. Doménech, O. León, M. S. Siddiqui, and J. Pegueroles, "Evaluating and enhancing intrusion detection systems in IoMT: The importance of domain-specific datasets," 2025, doi: 10.1016/j.iot.2025.101631.
13. University of New Brunswick, "NSL-KDD dataset for network intrusion detection systems." Accessed: Nov. 14, 2025. [Online]. Available: <https://www.unb.ca/cic/datasets/nsl.html>