I.J. Wireless and Microwave Technologies, 2026, 4, 95-110
Published Online on August 8, 2026 by MECS Press (http://www.mecs-press.org/)
DOI: 10.5815/ijwmt.2026.04.07

Modified Multi-Stage Ensemble Feature Selection (MMSE-FS) for Network Intrusion Detection

Faruq A. Al-Omari*
College of Engineering and Technology, American University in the Emirates, International Academic City, Dubai, UAE
Computer Engineering Department, Yarmouk University, Irbid, Jordan
E-mail: faruq.alomari@aue.ae, fomari@yu.edu.jo
ORCID iD: https://orcid.org/0000-0001-7321-3929
*Corresponding Author

Alaa Y. Mhesin
Network Engineering and Security Dept., Jordan University of Science and Technology, Irbid, Jordan
E-mail: aymhesin19@cit.just.edu.jo
ORCID iD: https://orcid.org/0009-0006-0041-9495

Mohammad M. Al-Shurman
Network Engineering and Security Dept., Jordan University of Science and Technology, Irbid, Jordan
E-mail: alshurman@just.edu.jo
ORCID iD: https://orcid.org/0000-0001-8764-7947

Received: June 5, 2026; Revised: June 24, 2026; Accepted: July 16, 2026; Published: August 8, 2026

Abstract: Intrusion Detection Systems (IDS) are essential for protecting modern networks against unauthorized access and evolving cyber threats. A persistent challenge in IDS design is the high dimensionality of network traffic data, which complicates the identification of the most relevant features for effective detection. This study introduces a modified multi-stage ensemble feature selection (MMSE-FS) framework that incorporates algorithmic adaptations of Random Forest (RF), Principal Component Analysis (PCA), and KBest methods. These enhanced variants are integrated through an intelligent ensemble voting mechanism, followed by a refinement stage that further strengthens feature relevance and discriminative capability.

To validate the proposed framework, experiments were conducted on the UNSW-NB15 benchmark dataset, reducing 49 initial features to 18 critical ones. The dataset was partitioned into 70% training and 30% testing subsets, and classification performance was evaluated using five machine learning classifiers (DT, RF, GB, KNN, and LR). Key hyperparameters of the proposed MMSE-FS framework ($\alpha = 0.75$, $\lambda = 1.0$, and $B = 50$ bootstrap repetitions) were determined through 5-fold cross-validation on the training partition and subsequently fixed for all experiments. The proposed framework achieved detection accuracies ranging from 99.03% to 99.83% for binary classification and from 94.20% to 96.60% for multi-class classification.

Compared with conventional feature selection methods, the proposed MMSE-FS framework substantially reduced the feature space while maintaining high detection performance across both binary and multi-class intrusion detection tasks. The reported results were obtained using the UNSW-NB15 dataset following the adopted preprocessing strategy, which excluded extremely underrepresented attack classes.

Index Terms: Intrusion detection systems (IDS), Feature selection framework, Ensemble learning, Dimensionality reduction, Network security

1. Introduction

In the digital age, “intrusion” refers to unauthorized access, tampering, or disruption of data, networks, and computer systems. Such activities range from spam and phishing to large-scale attacks like malware, denial-of-service (DoS), and data breaches [1–3]. With cybercrime costs escalating globally and organizations facing growing regulatory and security demands, the effectiveness of intrusion detection systems (IDS) has become critical to protecting digital infrastructure.

IDS identify anomalous traffic or behavior patterns to flag potential threats [4–6]. They are generally classified into host-based (HIDS) and network-based (NIDS). While HIDS focus on individual devices, NIDS analyze traffic flows

across entire networks [7–8]. NIDS are especially crucial in large-scale environments, relying on flow-based [9–10], time-based [11], rule-based [6], and correlation-based [12] methods. However, extracting the most relevant features from high-dimensional traffic data remains a persistent challenge.

High-dimensional data often contain redundant or irrelevant attributes that degrade detection accuracy and increase computational costs [13,15,19]. To mitigate these challenges, numerous feature selection techniques have been proposed, including hybrid IG–RF approaches [14,21], recursive feature elimination [15], correlation-based methods [20], and XGBoost-based feature selection [16]. Nevertheless, many existing approaches continue to face challenges in balancing dimensionality reduction with predictive accuracy, particularly when dealing with class-imbalanced intrusion detection datasets [16,18,19].

This paper proposes a modified multi-stage ensemble feature selection (MMSE-FS) framework for intrusion detection. The framework incorporates algorithmically adapted variants of Random Forest (RF), Principal Component Analysis (PCA), and KBest to address limitations associated with feature stability, class imbalance, and discriminative capability. These adapted variants are integrated through an ensemble-based voting mechanism, followed by a refinement stage that further reduces feature redundancy and improves feature relevance. The proposed MMSE-FS framework is designed to provide a systematic feature selection approach for high-dimensional intrusion detection datasets and is evaluated on the UNSW-NB15 benchmark dataset for both binary and multi-class classification tasks. The experimental results demonstrate that the proposed MMSE-FS framework reduces the feature space while maintaining high detection performance compared with the approaches considered in this study [16–18].

The remainder of this paper is structured as follows: Section II reviews related work; Section III presents the methodology; Section IV reports results and evaluation; and Section V concludes with directions for future research.

2. Related Work

Benchmark datasets such as KDD Cup ’99, NSL-KDD, CICIDS-2017, and UNSW-NB15 have been widely used to evaluate network intrusion detection systems (NIDS). Among these, UNSW-NB15 is considered one of the most comprehensive due to its representation of modern traffic and attack scenarios. Moustafa and Slay [13] presented a comprehensive statistical analysis of the UNSW-NB15 dataset, examining its feature characteristics, correlations, and complexity relative to the KDD99 dataset. Their study demonstrated that UNSW-NB15 provides a more challenging and representative benchmark for modern intrusion detection systems, thereby emphasizing the importance of effective feature selection for reducing dimensionality while preserving discriminative information.

Feature selection has since become central to improving IDS efficiency and accuracy. Zong et al. [14] proposed a two-stage system (IG-TS) that combined Information Gain with SMOTE, reducing the feature set to 23 and achieving 85.78% accuracy on UNSW-NB15. Meftah and Souhail [15] applied recursive feature elimination with random forests, obtaining a multi-class accuracy of 86.04% using 21 features. Kasongo et al. [16] employed XGBoost for ranking, isolating 19 critical features and improving decision tree accuracy by 1.9%. Roy and Singh [17] adopted Reduced Error Pruning Tree (REPTree), reducing 44 features to 20 and achieving 84.1% multi-class accuracy. Eunice et al. [18] integrated RF-based selection with deep neural networks, reaching 82.1% binary accuracy, but without evaluating multi-class performance.

Correlation-based techniques have also shown promise. Prasad et al. [20] used Pearson coefficients to eliminate redundant features, reducing the set to 15 and achieving 95.2% multi-class accuracy with decision trees. Yin et al. [21] introduced a hybrid IG–RF approach with recursive feature elimination, selecting 23 features and attaining 84.24% accuracy.

Beyond traditional methods, recent studies have continued to investigate the impact of advanced feature selection and classification strategies. Ahmed et al. [36] evaluated several feature selection techniques, including Chi-square, ANOVA, RFE, and Extra-Trees, across multiple machine learning classifiers on the CICIDS2017 dataset, demonstrating that appropriate feature selection can substantially improve detection performance while reducing feature dimensionality. Turukmane and Devendiran [37] proposed M-MultiSVM, integrating ASmoT for balancing and modified SVD for extraction, achieving 97.54% accuracy on UNSW-NB15. More et al. [38] tested multiple classifiers and showed the robustness of random forests, reporting 97.80% F1-score and 98.63% accuracy. Jouhari et al. [39] combined CNN-BiLSTM with X^2 feature selection, achieving 97.90% binary and 97.09% multi-class accuracy, illustrating the potential of deep learning hybrids.

More recently, intrusion detection research has expanded beyond conventional feature selection and deep learning models to incorporate graph neural networks (GNNs), large language models (LLMs), and advanced deep learning-based feature selection frameworks. Farrukh et al. [41] proposed XG-NID, a dual-modality intrusion detection framework that integrates heterogeneous graph neural networks with large language models to capture complex structural relationships and enhance decision interpretability. Similarly, Eljialy et al. [42] introduced a deep learning framework that combines multiple feature selection methods to improve intrusion detection performance. Although these approaches demonstrate the growing potential of graph-based and deep learning techniques, they generally involve substantially greater computational complexity and training requirements. Consequently, computationally efficient and interpretable feature selection frameworks remain highly relevant for practical network intrusion detection systems.

Overall, prior studies demonstrate that effective feature selection can significantly enhance IDS performance by reducing dimensionality and computational overhead. Reported results vary from 93.5% to 98.3% accuracy for binary classification [4,12,22] and 87.44% to 93% for multi-class classification [23–25]. However, challenges remain in jointly addressing high-dimensionality, class imbalance, and computational efficiency. Despite these advances, relatively limited attention has been given to simultaneously addressing feature stability, class imbalance, correlation-aware importance estimation, and objective feature subset determination within a unified feature selection framework. The proposed MMSE-FS framework addresses these challenges by integrating stability-regularized Random Forest feature importance, discriminative component selection in PCA, class-aware adaptive KBest ranking, and consensus-based ensemble voting into a single multi-stage feature selection pipeline.

3. Methodology

This study presents a systematic multi-stage framework for feature reduction in network intrusion detection. The approach is designed to minimize dimensionality while preserving discriminative power, thereby improving both efficiency and accuracy of IDS models. The methodology comprises three key phases: data preparation, modified feature selection, and model evaluation.

3.1. Dataset and Preprocessing

The UNSW-NB15 dataset [26], a widely used benchmark for NIDS research, was selected for evaluation due to its realistic simulation of modern traffic and nine attack categories (fuzzers, analysis, backdoors, DoS, exploits, generic, reconnaissance, shellcode, and worms). It includes 49 features, of which 42 are traffic attributes, in addition to metadata and class labels.

Following prior studies [21,27], we removed two redundant attributes (Attack Name and Attack Reference), leaving 47 features. Minority classes (Shellcode and Worms), comprising less than 0.002% of all records, were excluded to mitigate extreme imbalance and misclassification risk. The resulting dataset included eight categories of traffic, supporting both binary and multi-class classification tasks, Fig. 1.

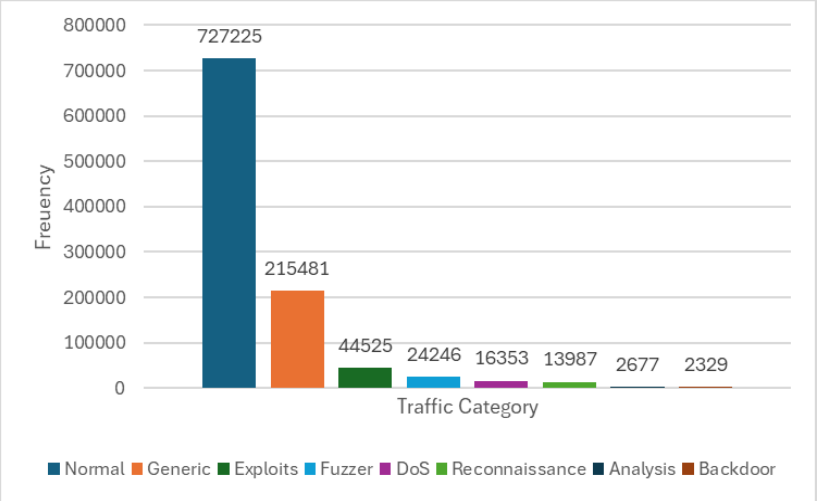


Fig. 1. Class distribution of normal traffic and attack categories in the UNSW-NB15 dataset.

Categorical attributes (service, proto, state) were numerically encoded using a label encoder, and the dataset was partitioned into training and testing subsets (70%/30% split), preserving original label distributions. To ensure consistent scale across features, MinMax normalization was applied, as:

$$x_{scaled} = \frac{x - min(x)}{max(x) - min(x) + \epsilon} \quad (1)$$

where ϵ is a small constant to avoid division by zero. This transformation mapped all features to the range [0,1], improving comparability and stabilizing classifier training [28].

The distribution of features by category after preprocessing is summarized in Table 1.

Table 1. Number of Features per Category after Preprocessing

Feature Category	Number of Features
Flow-based	5
Basic	13
Content-based	8
Time-based	9
General purpose	5
Connection-based	7
Total	47

3.2. Proposed Multi-Stage Ensemble Feature Selection (MMSE-FS) Framework

The proposed methodology is a modified multi-stage ensemble feature selection (MMSE-FS) framework that incorporates algorithmic adaptations to Random Forest (RF), Principal Component Analysis (PCA), and KBest feature selection. Rather than applying these techniques in their conventional forms, each method is adapted to address specific challenges associated with intrusion detection, including feature stability, class imbalance, and discriminative capability. The adapted methods are subsequently integrated through an ensemble-based voting mechanism, followed by a refinement stage to reduce feature redundancy and improve the quality of the final feature subset, as illustrated in Fig. 2.

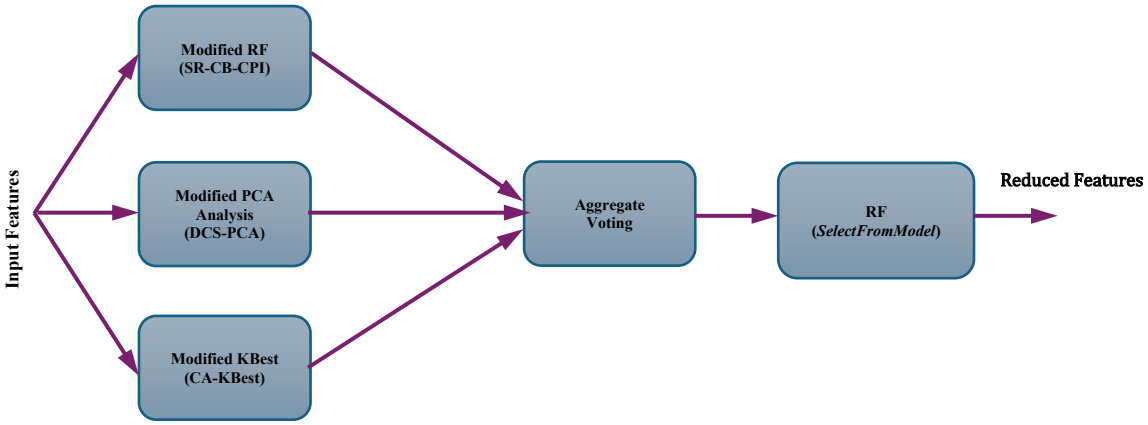


Fig. 2. Overview of the proposed multi-stage ensemble feature selection framework.

As shown in Fig. 2, the proposed MMSE-FS framework integrates modified variants of Random Forest (RF), Principal Component Analysis (PCA), and KBest. Each algorithm operates independently to generate candidate feature subsets, which are then combined through an ensemble voting mechanism. Features appearing in at least two algorithms are retained. Finally, an RF-based refinement stage (*SelectFromModel*) eliminates redundancy and ensures only highly discriminative and stable features are preserved.

3.3. Modified Feature Selection Algorithms

To address the limitations of conventional feature selection methods in intrusion detection, this study introduces algorithmic adaptations to Random Forest (RF), Principal Component Analysis (PCA), and KBest feature selection. Each adapted method targets a specific limitation of its conventional counterpart, namely feature ranking stability, sensitivity to class imbalance, and discriminative feature selection. The following subsections describe the rationale, implementation, and expected role of each adapted method within the proposed multi-stage framework.

3.4. Modified Random Forest (SR-CB-CPI)

Random Forest (RF) is a widely used ensemble learning algorithm that constructs multiple decision trees on bootstrapped subsets of data, aggregating their predictions to improve accuracy and reduce overfitting [29–35]. In its standard form, RF evaluates feature importance through impurity-based measures such as Gini index or entropy reduction. While effective, these measures are biased toward high-cardinality features, unstable across bootstraps, and insensitive to class imbalance. These limitations reduce their suitability for intrusion detection tasks.

To overcome these issues, this study introduces a Stability-Regularized, Class-Balanced Conditional Permutation Importance (SR-CB-CPI) approach as a modification of the conventional RF feature ranking process. The adaptation incorporates four key enhancements:

Class-Balanced Training: The RF model is trained with class weights to mitigate skew in the UNSW-NB15 dataset, ensuring that minority attack categories contribute proportionally to the feature evaluation process. The out-of-bag (OOB) samples generated during bootstrap aggregation are used to compute the baseline per-class F1 scores, providing an unbiased estimate of predictive performance prior to feature permutation.

Conditional Permutation Importance: Instead of naive permutation, each feature is permuted within correlated feature groups (e.g., time-based or flow-based subsets). This preserves underlying traffic correlations and prevents artificially inflated or suppressed importance values.

Per-Class Contribution Scoring: For each permutation, the decrease in F1-score is measured separately across classes, providing a fine-grained estimate of how each feature supports minority as well as majority classes. Feature importance is then aggregated using inverse-prevalence weighting to emphasize features that contribute to rare attacks:

$$\mu_j = \sum_c \omega_c . \Delta F1_c(j), \quad \omega_c \propto \frac{1}{\pi_c^\alpha}, \quad 0.5 \leq \alpha \leq 1 \tag{2}$$

where π_c is the prevalence of class c and α controls the weighting strength. where π_c is the prevalence of class c, and α controls the strength of inverse-prevalence weighting. The parameter was selected through 5-fold cross-validation on the training partition by evaluating candidate values within the predefined range of 0.5–1.0 using the macro-F1 score as the optimization criterion. A value of α = 0.75 was adopted for all experiments as it provided the best balance between minority-class sensitivity and overall classification performance.

Stability Regularization: To improve robustness, the procedure is repeated across multiple bootstraps (B) of the dataset. To improve robustness, the procedure is repeated across multiple bootstraps (B) of the dataset. The number of bootstrap repetitions was selected through 5-fold cross-validation on the training partition by evaluating candidate values with respect to the macro-F1 score. A value of B = 50 was adopted, providing stable feature importance estimates while maintaining computational efficiency. The mean importance μ_j is penalized by its standard deviation σ_j, yielding the final stability-adjusted score:

$$S_j = \mu_j - \lambda . \sigma_j \tag{3}$$

where λ is the stability regularization factor controlling the penalty applied to variability in feature importance across bootstrap iterations. The parameter was optimized through 5-fold cross-validation on the training partition using macro-F1 as the optimization criterion. A value of λ=1.0 was adopted, providing effective suppression of unstable features while preserving consistently informative ones.

The final RF importance score Sj is sorted in descending order, and an elbow detection method is applied to determine the threshold for retaining features. Specifically, the elbow point is identified as the location where the rate of decrease in the stability-adjusted importance scores changes most noticeably, indicating the transition from highly informative to marginally contributing features. This data-driven criterion provides an objective threshold for feature selection, balancing predictive performance with feature subset compactness while avoiding arbitrary cut-off values. This modification ensures that the selected subset is class-aware, correlation-resilient, and stable, thereby overcoming the key shortcomings of conventional RF importance in the IDS context. The following pseudocode illustrates the modified RF algorithm:

Feature Selection Using Modified RF

```
Procedure SR_CB_CPI_RF (X, y, B=50, λ=1, α=0.75):
Train RF with class weights → get OOB baseline F1 per class
For each feature j:
   deltas ← [ ]
   Repeat B times:
      Conditionally permute feature j (preserve correlations)
      Compute per-class ΔF1 = F1_base − F1_perm
      Weighted macro drop: μ_j = Σ_c (w_c * ΔF1_c), w_c ∝ 1/π_c^α
      Append μ_j to deltas
   End Repeat
  S_j = mean(deltas) − λ * std(deltas)
End For
Sort features in descending order of Sj
Determine elbow threshold from the ranked importance curve
Select features with S_j ≥ elbow threshold
Return selected feature set
```

3.5. Modified Principal Component Analysis (DCS-PCA)

Principal Component Analysis (PCA) reduces dimensionality by projecting data into orthogonal components that capture maximum variance [31]. However, conventional PCA may retain components with high variance but low discriminative value for classification, particularly in imbalanced datasets such as UNSW-NB15.

To overcome this, we introduce a Discriminative Component Selection PCA (DCS-PCA). In this adaptation, each principal component (PC) is evaluated using two criteria: (i) explained variance ratio (EVR), and (ii) discriminative score (DS) with respect to class labels, measured using the ANOVA F-statistic. A combined ranking score is then computed as:

$$C_k = \alpha . EVR_k + (1 - \alpha) . NormDS_k, \quad 0 \leq \alpha \leq 1 \tag{4}$$

where a balances variance preservation and class discrimination. The parameter was selected through 5-fold cross-validation on the training partition by evaluating candidate values within the range $0 \leq \alpha \leq 1.0$ using the macro-F1 score as the optimization criterion. A value of $\alpha = 0.75$ was adopted for all experiments as it provided the most effective trade-off between preserving informative variance and maximizing class separability.

PCs are ranked by Ck, and the elbow method is applied to determine the optimal subset. Specifically, the elbow point is identified as the point where the rate of improvement in the combined ranking scores decreases markedly, indicating diminishing returns from retaining additional principal components. This data-driven criterion provides an objective threshold that balances variance preservation with discriminative capability while avoiding arbitrary component selection. This ensures that the retained components both capture informative variance and contribute effectively to class discrimination, thereby improving robustness against class imbalance and feature redundancy. The modified DCS-PCA procedure is summarized in the following pseudocode:

Feature Selection Using Modified PCA

```
Procedure DCS_PCA(X, y, α = 0.75):
  Center and scale X
  Compute PCA → PCs, explained variance ratios (EVR)
  For each PC k:
      Compute discriminative score DS_k (e.g., ANOVA F vs. labels y)
     Normalize DS_k → NormDS_k
    Combined score C_k = α·EVR_k + (1−α)·NormDS_k
  Rank PCs by C_k
 Select top PCs using elbow threshold
Return selected PCs and transformed data
```

Following the selection of the principal components using the proposed DCS-PCA criterion, the retained components were mapped back to the original feature space through the PCA loading matrix. Specifically, the contribution of each original feature was quantified by aggregating the absolute loading coefficients across the retained principal components. Features with higher aggregated absolute loadings were considered to have a greater contribution to the retained principal components and were therefore assigned higher importance scores for subsequent ranking and selection. This mapping preserves the modified component selection strategy while enabling the identification of the most influential original features for the subsequent ensemble voting stage.

3.6. Modified KBest (CA-KBest)

KBest is a filter-based method that selects features by ranking them according to statistical tests such as χ^2 or ANOVA F-values [31]. Although efficient, its conventional form treats features independently and is often biased toward attributes aligned with majority classes, overlooking those critical for minority categories.

To address this, we propose a Class-Aware Adaptive KBest (CA-KBest). In this modification, discriminative scores are computed for each feature across all classes individually. Scores are then aggregated using inverse-prevalence weights to emphasize minority classes:

$$S_j = \sum_c \omega_c . DS_c(j), \qquad \omega_c \propto \frac{1}{\pi_c^{\alpha}} \tag{5}$$

where $DS_c(j)$ is the class-specific discriminative score, π_c is the class prevalence, and α controls the strength of inverse-prevalence weighting. Consistent with the other stages of the proposed framework, the parameter was selected through 5-fold cross-validation on the training partition by evaluating candidate values using the macro-F1 score as the optimization criterion. A value of $\alpha = 0.75$ was adopted for all experiments. Features are ranked by their weighted scores, and the optimal subset size is determined adaptively through elbow detection or validation performance rather than by a fixed *k*. When elbow detection is employed, the threshold is identified as the point where further increases in the number of selected features yield only marginal improvements in the weighted discriminative scores, providing an objective balance between feature relevance and subset compactness.

This adaptation ensures that the selected features are both statistically discriminative and balanced across classes, thereby improving robustness in highly imbalanced intrusion detection datasets.

Feature Selection Using Modified KBest

```
Procedure CA_KBest(X, y, α = 0.75, metric, selection):
Compute class prevalences π_c and weights w_c ∝ 1 / π_c^α ; normalize Σ_c w_c = 1
For each feature j:
      For each class c:
    y_bin ← 1 if y==c else 0
    DS_cj ← DiscriminativeScore(X[:,j], y_bin, metric)   # χ² / ANOVA F / MI
Normalize {DS_cj}_c across classes
S_j ← Σ_c (w_c · DS_cj)                  # class-aware aggregate
Rank features by S_j (desc)
Determine elbow threshold τ from the ranked scores
F ← {j | S_j ≥ τ}
Return F (selected features) and S (scores)
```

3.7. Ensemble Voting and Final Refinement

After obtaining candidate subsets from the three adapted selectors, namely SR-CB-CPI RF, DCS-PCA, and CA-KBest, an aggregated approach is followed using a consensus voting scheme to improve stability and reduce method-specific bias. Let F^{RF}, F^{PCA}, and F^{KB} denote the selected feature sets and s^{RF}, s^{PCA}, and s^{KB} their corresponding score vectors.

Consensus selection. A feature fj is retained if it is selected by at least two methods:

$$F^{vote} = \{j \mid 1[j \in F^{RF}] + 1[j \in F^{PCA}] + 1[j \in F^{KB}] \geq 2\} \quad (6)$$

For ties at the decision boundary, we break ties by a normalized score sum

$$\tilde{s_j} = \hat{s}_j^{RF} + \hat{s}_j^{PCA} + \hat{s}_j^{KB} \quad (7)$$

where each $\hat{s}$ is min–max normalized to [0,1].

Final refinement. The consensus subset Fvote is passed to an RF SelectFromModel step trained with class weights on the training split only. We retain features with importance ≥ the mean importance to remove residual redundancy and enforce stability.

Ensemble Voting & Final Refinement

```
Procedure EnsembleRefine(F_RF, s_RF, F_PCA, s_PCA, F_KB, s_KB, X_train, y_train):
    F_vote ← { j | 1[j∈F_RF] + 1[j∈F_PCA] + 1[j∈F_KB] ≥ 2 }
    If tie at boundary:
        ŝ_RF, ŝ_PCA, ŝ_KB ← min–max normalize scores
        rank ties by ŝ_sum(j) = ŝ_RF[j]+ŝ_PCA[j]+ŝ_KB[j]
    X_sub ← X_train[:, F_vote]
    RF_ref ← TrainRandomForest(X_sub, y_train, class_weights=balanced)
    imp ← RF_ref.feature_importances_
    τ ← mean(imp)  # or elbow on sorted(imp)
    F_final ← { F_vote[j] | imp[j] ≥ τ }
    return F_final
```

Complexity note. Voting is O(m). The refinement RF adds O(T. n. m_{try}. log n), where m is the number of original candidates features after preprocessing (m=47 in our case), n is the number of training samples, T is the number of trees (T=100 in our setup), and m_{try} is defined as:

$$m_{try} = min(m', \max\_features) \quad (8)$$

Where m' is the feature count after consensus voting (i.e., $|F^{vote}|$ before the final RF refinement, typically much smaller than m keeping the entire stage computationally modest [40].

3.8. Classification Algorithms for Evaluation

To assess the effectiveness of the proposed feature selection framework, we trained five widely used classifiers representing complementary learning paradigms: a single Decision Tree (DT), a Random Forest (RF), a Gradient Boosting (GB) classifier, a K-Nearest Neighbors (KNN) classifier, and Logistic Regression (LR). Using classifiers with diverse

learning characteristics enables a more comprehensive evaluation of whether the selected features generalize across different classification models rather than favoring a particular learning paradigm. All models were trained exclusively on the training split using the final refined feature subset, while the test split was reserved for final performance evaluation.

Decision Tree (DT). A CART-style tree is employed with Gini impurity, enabling transparent, interpretable decision boundaries. To reduce variance and overfitting, tree growth is regularized (minimum samples per split/leaf and maximum depth tuned on the training split). DT provides a strong baseline for gauging the discriminative quality of the reduced feature set.

Random Forest (RF). Independent of the RF stages used for selection/refinement, we train a separate RF classifier for prediction. The model aggregates T trees (default $T = 100$), uses feature subsampling at each split$\sqrt{d}$ (with d the number of selected features), and applies class_weight = "balanced" for the multi-class task to mitigate residual imbalance. This evaluates performance under a robust, low-bias ensemble.

Gradient Boosting (GB). We use a gradient boosting classifier with decision stumps/shallow trees as base learners. Learning rate, number of estimators, and maximum depth are tuned on the training split (defaults: learning rate 0.1, estimators 200, depth 3). GB complements bagging by emphasizing hard-to-classify samples and tests whether the selected features support additive, margin-maximizing ensembles.

K-Nearest Neighbors (KNN). KNN provides a non-parametric check on the geometric separability of the selected features. Inputs are MinMax-scaled; k and the distance weighting scheme are tuned on the training split (defaults: Euclidean distance, weights="distance"). Because KNN is sensitive to scale and redundant attributes, it is an informative stress test for the feature subset.

Logistic Regression (LR). Logistic Regression was included as a linear baseline classifier to assess the separability of the selected feature subset under a linear decision boundary. The model was trained using L2 regularization with hyperparameters tuned on the training split. Although Logistic Regression is computationally efficient and highly interpretable, its performance may be limited when the underlying class boundaries are nonlinear. Including LR provides an additional reference for evaluating the effectiveness of the proposed feature selection framework across both linear and nonlinear classifiers.

Binary vs. Multi-Class Setups. All five classifiers were trained for both binary (attack vs. normal) and multi-class (all studied attack categories plus normal) classification tasks. For the multi-class task, tree-based models handle the problem natively; KNN uses the standard majority (or distance-weighted) rule, and Logistic Regression employs a one-vs-rest strategy as implemented in scikit-learn.

3.9. Evaluation Protocol and Metrics

Protocol. All preprocessing, feature selection, and model tuning were performed exclusively on the training split. The dataset is split 70/30 (train/test) with stratification preserved for both the binary and multi-class setups. PCA fitting, RF importance estimation, KBest scoring, ensemble voting, and the RF-based refinement step are all conducted within the training data; the test split is used once for final reporting. Confusion matrices are provided to analyze error modes.

Hyperparameter selection. To ensure a consistent and unbiased evaluation, the key hyperparameters of the proposed MMSE-FS framework were selected using 5-fold cross-validation on the training partition only. Candidate values were evaluated using macro-F1 as the selection criterion to account for class imbalance. The selected values were then fixed for all subsequent feature selection and classifier evaluation experiments. Specifically, $\alpha = 0.75$ was adopted for inverse-prevalence weighting and DCS-PCA component balancing, $\lambda = 1.0$ was adopted for stability regularization in SR-CB-CPI RF, and $B = 50$ bootstrap repetitions were adopted for permutation-importance stability estimation. The independent test partition was not used during hyperparameter selection.

Evaluation Metrics. Given class imbalance, we emphasize macro-averaged measures (each class contributes equally) and report standard point and threshold-free metrics:

Accuracy is defined as the proportion of true results (both true positives and true negatives) among the total number of cases examined. It is a measure of the correctness of a classification model. That is:

$$Accuracy = \frac{Number\ of\ Correct\ Predictions}{Total\ Number\ of\ Predictions} = \frac{TP + TN}{TP + TN + FP + FN} \tag{9}$$

Where:

True Positive (TP): The number of instances correctly predicted as positive.

True Negative (TN): The number of instances correctly predicted as negative.

False Positive (FP): The number of instances incorrectly predicted as positive (also known as a Type I error).

False Negative (FN): The number of instances incorrectly predicted as negative (also known as a Type II error).

Precision is the proportion of true positive (*TP*) predictions among all positive predictions made by the model. It measures the accuracy of the positive predictions. That is:

$$APrecision = \frac{TP}{TP + FP} \tag{10}$$

Recall, also known as *sensitivity* or true positive rate, is the proportion of true positive predictions among all actual positive instances. It measures the model's ability to identify all relevant instances. That is:

$$Recall = \frac{TP}{TP + TN} \tag{11}$$

The *F1 score* is the harmonic mean of precision and recall. It provides a single metric that balances both the precision and recall of a model, making it useful when both false positives and false negatives are considered. That is:

$$F1\ score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{12}$$

Averaging scheme. For multi-class evaluation, we report both micro-F1 and macro-F1. Micro-F1 is computed by aggregating true positives, false positives, and false negatives over all classes before deriving precision, recall, and F1, and thus is influenced by the dominant classes. Macro-F1 is the unweighted mean of per-class F1 scores, treating all classes equally and offering a class-balanced view under imbalance. In the binary setting, micro-averaged precision, recall, and F1 coincide with accuracy; accordingly, we report accuracy together with the positive-(attack)-class precision, recall, and F1, and additionally include macro-*F1* (the mean of the positive- and negative-class F1) as a class-balanced summary.

4. Results and Discussion

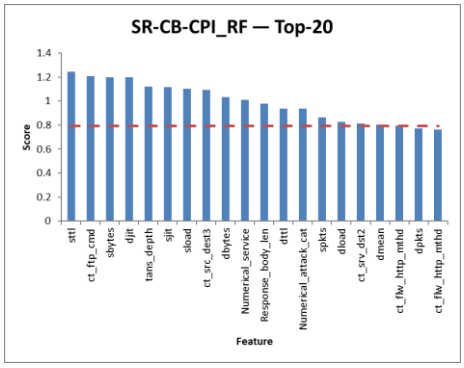
All experiments were conducted in Python 3.8 using scikit-learn, NumPy, Pandas, and Matplotlib on a Windows 11 Pro machine (Intel® Core™ i5-8265U, 16 GB RAM). Models were trained on the training split only; the test split was used once for final reporting. Preprocessing (encoding, scaling), modified feature selection (Sections 3.2–3.4), and hyperparameter tuning (Section 3.5) followed the protocol in Section 3.6.

4.1. Feature Selection Outcomes

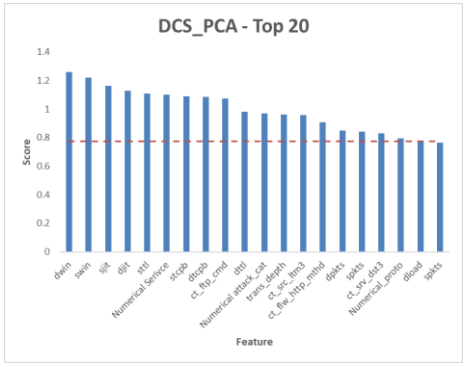
Applying the modified selectors to the preprocessed UNSW-NB15 data yielded compact, largely consistent subsets. The SR-CB-CPI RF ranked features via stability-regularized, class-balanced conditional permutation. DCS-PCA retained the most discriminative principal components, which were subsequently mapped to the original feature space using the procedure described in Section III.5. CA-KBest provided class-aware, imbalance-adjusted scores. Individually, the three selectors proposed 19, 19, and 21 features, respectively.

Using consensus voting (≥ 2 of 3) produced an intermediate set of 32 features, which the RF *SelectFromModel* refinement reduced to a final subset of 18 features (Fig. 3). Table 2 summarizes the distribution across categories (Basic 7, Time-based 5, Content-based 4, General-purpose 1, Connection-basedd 1). These results confirm that the framework attains substantial dimensionality reduction while preserving discriminative coverage across traffic descriptors.

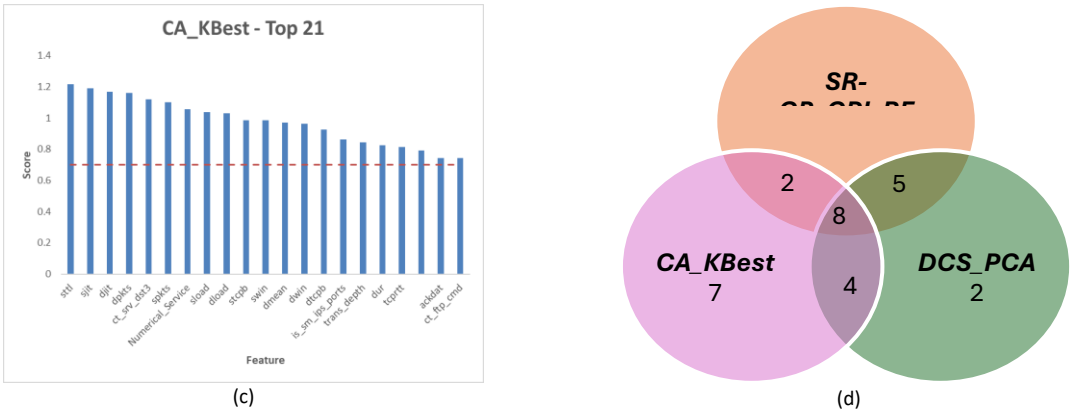
The results of applying RF and PCA and subsequent thresholding highlight the most critical features selected by these algorithms. These intermediate results, up to this stage, are depicted in Fig. 3.



(a)



(b)



(c) (d)

Fig. 3. Outputs of the modified selectors in the training phase, (a) SR-CB-CPI RF, (b) DCS-PCA, (c) CA-KBest, and (d) their overlap.

As illustrated in Fig. 3., the three selectors agree on a core set of features, yet each also yields some method-specific choices reflecting their complementary criteria. Using a ≥2-of-3 consensus rule yields an intermediate set of 32 features, as summarized in Fig. 3(d).

We then applied an RF-based SelectFromModel refinement (class-weighted; training split only) to remove residual redundancy. Features with importance at or above the mean were retained, producing a final subset of 18 features, reducing dimensionality from 47 while preserving discriminative coverage across categories. Table 2 reports the distribution of the retained features by category.

Table 2: Breakdown of the final feature set (18) by category

Feature Category	Basic	Time-based	Content-based	General Purpose	Connection -basedd
Number of Features	7	5	4	1	1

The framework in Fig. 2. provides a modular, dataset-agnostic pipeline for dimensionality reduction and feature prioritization. Although evaluated on UNSW-NB15, the procedures (modified RF (SR-CB-CPI), DCS-PCA, CA-KBest, consensus voting, and RF refinement) can be re-fit on other network traces, provided standard preprocessing (cleaning, encoding, scaling) is applied and class imbalance is addressed. Method-specific hyperparameters such as λ and B for SR-CB-CPI RF, and α for DCS-PCA/CA-KBest, are estimated on the new training split, after which the same consensus and refinement steps yield a compact subset. As summarized in Table 2, the resulting features maintain discriminative coverage with minimal redundancy, supporting its applicability to datasets with different traffic profiles and attack mixes, provided appropriate preprocessing and parameter tuning are performed.

Next, classification performance is evaluated on this compact subset for both binary and multi-class detection.

4.2. Binary Classification Results

Binary intrusion detection (normal vs. attack) was evaluated on the held-out test split using the final 18-feature subset. Accuracy, Precision, Recall, and F1-score were computed, with F1-score emphasized due to class imbalance. Results are summarized in Table 3.

Table 3. Binary classification results using the final 18-feature subset

Metric \ ML Model	GB	DT	LR	RF	KNN
Precision	99.9701%	99.8570%	95.4438%	99.9917%	98.7130%
Recall	99.5627%	99.8863%	99.4738%	99.4995%	99.9111%
Accuracy	99.6755%	99.8217%	96.3356%	99.6465%	99.0333%
F1-score	99.7660%	99.8717%	97.4171%	99.7450%	99.3084%

As can be seen from table 3, tree-based methods provided the strongest performance, KNN was competitive, and Logistic Regression trailed, indicating that the reduced feature space supports non-linear decision boundaries more effectively than linear ones.

Fig. 4. presents the confusion matrices for the binary classifiers (rows: true class; columns: predicted). Across models, true positives and true negatives dominate, with false negatives notably rare, critical for IDS, while false positives are

limited. Overall error rates are sparse, consistent with the imbalance-aware feature selection and indicating that the 18-feature subset preserves near-complete separability between attack and normal traffic.

Fig. 4. Confusion matrices for binary intrusion detection using different machine learning models.

4.3. Multi-Class Classification Results

Multi-class intrusion detection was evaluated on the held-out test split using the final 18-feature subset. Evaluation metrics follow Section 3.6, with macro-F1 emphasized under imbalance. Table 4 reports model scores. Tree-based methods (DT, RF, GB) lead; KNN is competitive; LR lags, indicating that non-linear boundaries better exploit the reduced feature space.

Table 4. Multi-class classification results using the final 18-feature subset

Metric \ ML Model	GB	DT	LR	RF	KNN
Precision	96.20%	96.75%	80.42%	96.35%	94.13%
Recall	96.17%	96.60%	84.65%	95.22%	94.20%
Accuracy	96.17%	96.60%	84.65%	95.22%	94.20%
F1 score	95.65%	96.56%	81.42%	94.60%	94.12%

Fig. 5. presents the confusion matrix for the DT model as a representative case. Most errors occur between attack classes, with rare mislabeling of attacks as normal, operationally preferable for IDS. Under-represented classes (e.g., Analysis, Backdoor, DoS) show lower recall, consistent with class imbalance; high recall is maintained for normal and frequent attack types (e.g., Generic).

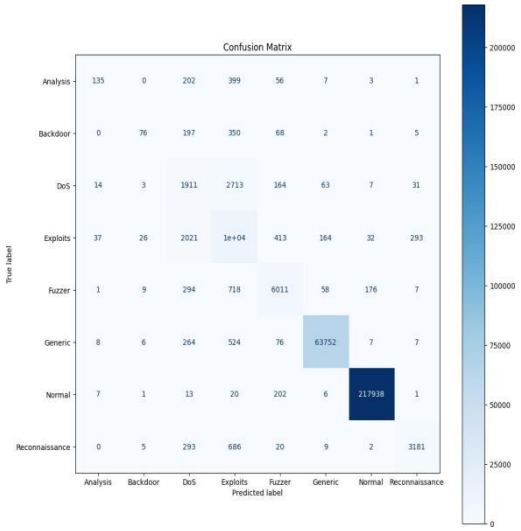


Fig. 5. Confusion matrix for multi-class intrusion detection using the Decision Tree with the final 18-feature subset.

Fig. 5. shows that residual errors occur mainly among attack categories, with attacks rarely mislabeled as normal, an operationally preferable profile for IDS. Normal exhibits near-perfect recall (high support, distinct patterns), and Generic performs strongly, while Analysis, Backdoor, and DoS show lower recall, reflecting limited representation and greater similarity to other attacks. Overall, the 18-feature subset sustains high multi-class accuracy while concentrating mistakes within the attack space. This pattern indicates that benign–malicious separation is well preserved, whereas fine-grained attack discrimination remains constrained by class imbalance; further gains are expected from class-aware reweighting or resampling, calibrated/class-specific thresholds, and targeted augmentation for underrepresented classes.

4.4. Comparison with Prior Work

Comparisons on UNSW-NB15 are summarized in Tables 5 (binary) and 6 (multi-class). Using the compact 18-feature subset produced by the MMSE-FS framework, the proposed approach achieves performance that is comparable to, and in some cases exceeds, previously reported results despite using fewer selected features. Within the experimental protocol adopted in this study, the results suggest that the proposed feature selection and ensemble refinement strategy effectively preserves discriminative information while substantially reducing the feature space.

Table 5. Comparison with prior work - Binary intrusion detection on UNSW-NB15

Research Method	Approach	No. of Selected Features	F1 score (%)	Accuracy (%)
Proposed Method	MMSE-FS	18	99.83	99.83
Meftah et al. [15]	SVM	-	-	82.11
Kocher and Kumar [19]	RF	23	100.00	99.64
Aleesa et al. [34]	ANN	48	-	99.26
More et al. [38]	RF	-	97.80	98.63
Jouhari et al. [39]	CNN-BiLSTM	-	-	97.90

Multi-class comparisons are summarized in Table 6.

Table 6. Comparison with prior work – Multi-class intrusion detection on UNSW-NB15

Research Method	Approach	No. of Selected Features	F1 score (%)	Accuracy (%)
Proposed Method	MMSE-FS Framework	18	96.56	96.60
Kasongo and Sun [16]	FFDNN	19	77.28	77.50
Roy and Singh [17]	MLP	20	-	84.10
Eunice et al. [18]	DNN	20	-	82.10
Yin et al. [21]	MLP	23	82.85	84.24
Turukmane & Devendiran [37]	M-MultiSVM with ONgO Optimization	-	-	97.54
Jouhari et al. [39]	CNN-BiLSTM	-	-	97.09

For the multi-class task, the compact feature subset achieves competitive Accuracy and F1-score relative to several previously reported studies on UNSW-NB15 [16–18,21], while approaching the performance of more complex models reported in [37,39]. These findings indicate that effective intrusion detection can be achieved using a substantially reduced feature set without relying on computationally intensive deep learning architectures.

Direct quantitative comparisons with previously published studies should be interpreted with caution because differences in dataset preprocessing, treatment of minority classes, feature engineering, resampling strategies, train-test partitioning, evaluation protocols, and reported performance metrics can substantially influence the resulting performance. Consequently, the comparisons presented in Tables 5 and 6 are intended to provide qualitative contextual benchmarking rather than statistically controlled performance comparisons. Within the common UNSW-NB15 benchmark, the proposed MMSE-FS framework demonstrates that competitive intrusion detection performance can be achieved while substantially reducing the feature space to only 18 selected features.

Overall, the MMSE-FS framework reduced the feature space from 47 to 18 while preserving strong discriminative power. On the held-out test split, binary detection reached ≈99.8% F1/Accuracy, and multi-class detection achieved ≈96.6% with errors concentrated among attack categories rather than benign traffic. Compared with prior work, comparable or superior performance is obtained using substantially fewer features, with remaining limitations tied to underrepresented classes and cross-study protocol differences. Future work includes cross-dataset validation and algorithmic refinements to improve robustness and adaptability.

5. Conclusion

A Modified Multi-Stage Ensemble Feature Selection (MMSE-FS) framework was presented for network intrusion detection. The approach integrates adapted variants of Random Forest (SR-CB-CPI, class-balanced and stability-regularized conditional permutation), DCS-PCA (variance–discriminability driven component selection), and CA-KBest (class-aware, imbalance-adjusted ranking), followed by ≥2-of-3 consensus and an RF SelectFromModel refinement. Applied to UNSW-NB15, the framework reduced the feature space from 49 to 18 while preserving strong discriminative power. On the held-out test split, binary detection achieved approximately 99.8% F1-score and accuracy for binary classification, while multi-class classification achieved approximately 96.6% accuracy. Confusion analyses indicated that residual multi-class errors occur mainly among attack categories rather than benign traffic, which is operationally favorable.

The proposed MMSE-FS framework provides a modular feature selection pipeline that can be applied to intrusion detection datasets following appropriate preprocessing and parameter tuning. The present study evaluated the framework using the UNSW-NB15 dataset, where it demonstrated effective dimensionality reduction while maintaining strong classification performance. However, the current implementation relies on the exclusion of extremely underrepresented attack classes during preprocessing, which may limit its applicability to datasets containing severe class imbalance. Future work will investigate the individual contribution of the modified feature selection components through a systematic ablation analysis, alternative strategies for handling extreme class imbalance while preserving all attack categories, and further evaluation on additional benchmark intrusion detection datasets.

All the Declarations and Statements

Author Contributions Statement

Faruq Al-Omari – the corresponding author, conceptualized the algorithmic framework, provided guidance on the computational methodology, and verified the accuracy of the code and data analysis. He played a key role in refining the research approach, overseeing the technical implementation, and ensuring the integrity of the findings. Additionally, he contributed to structuring and reviewing the manuscript to enhance its clarity and coherence.

Alaa Mhesin – was responsible for implementing the coding aspects of the research, conducting the primary data analysis, and interpreting the results in consultation with her supervisors. She actively participated in discussions regarding the findings and contributed significantly to drafting and revising the manuscript.

Mohammad Shurman – contributed to the validation and verification of the data analysis and results. He was involved in critical discussions to interpret the findings, ensuring the robustness of the conclusions. Furthermore, he provided insights into refining the theoretical aspects of the study and actively contributed to the writing and final revision of the manuscript.

All authors have read and agreed to the published version of the manuscript.

Conflict of Interest Statement

The authors declare no conflicts of interest.

Funding Declaration

The authors declare that no funds, grants, or other financial support were received for conducting this research.

Data Availability Statement
The datasets generated and/or analyzed during the current study are available in the UNSW-NB15 Dataset repository, https://research.unsw.edu.au/projects/unsw-nb15-dataset

Ethical Declarations
The authors should declare the following: Statements of ethical approval for studies involving human subjects and/or animals.

Acknowledgments
N/A

Declaration of Generative AI in Scholarly Writing
Generative AI was used solely to improve the language and readability of this manuscript. All scientific content was reviewed, verified, and approved by the authors, who assume full responsibility for the final manuscript.

Abbreviations
The following abbreviations are used in this manuscript:

IDS – Intrusion Detection Systems
MMSE-FS – Modified Multi-Stage Ensemble – Feature Selection
RF – Random Forest
PCA – Principal Component Analysis
DT – Decision Trees
GB – Gradient Boosting
KNN – K-Nearest Neighbor
LR – Logistic Regression
GNN – Graph Neural Networks
LLM – Large Language Models
DoS – Denial of Service
SR-CB-CPI-RF – Stability Regularized, Class Balanced, Conditional Permutation Importance – Random Forest
OOB – Out of Bag
DCS-PCA – Discriminative Component Selection – Principal Component Analysis
CA-KBest – Class Aware K Best
AI - Artificial Intelligence
NLP - Natural Language Processing
DL - Deep Learning

Appendix A/B/C…, with appendix title
N/A

References

[1] Mukherjee, B., Heberlein, L. T., & Levitt, K. N. (1994). Network intrusion detection. *IEEE Network*, 8(3), 26-41. https://api.semanticscholar.org/CorpusID:263738512
[2] Stavroulakis, P., & Stamp, M. (Eds.). (2010). Handbook of Information and Communication Security. Springer. DOI: 10.1007/978-3-642-04117-4
[3] Modi, C. N., Patel, D. R., Patel, A., & Rajarajan, M. (2012). Integrating signature apriori based network intrusion detection system (NIDS) in cloud computing. *Procedia Technology*, 6, 905-912. DOI: http://dx.doi.org/10.1016/j.protcy.2012.10.110
[4] Kanimozhi, V., & Jacob, P. (2019). UNSW-NB15 dataset feature selection and network intrusion detection using deep learning. *International Journal of Recent Technology and Engineering*, 7(5). DOI: 10.35940/ijrte.D5002.078519
[5] Parimala, G., & Kayalvizhi, R. (2021). An effective intrusion detection system for securing IoT using feature selection and deep learning. In 2021 *International Conference on Computer Communication and Informatics (ICCCI)*, 1-4 IEEE. DOI: 10.1109/ICCCI50826.2021.9402562
[6] Ayo, F. E., Folorunso, S. O., Abayomi-Alli, A. A., Adekunle, A. O., & Awotunde, J. B. (2020). Network intrusion detection based on deep learning model optimized with rule-based hybrid feature selection. *Information Security Journal: A Global Perspective*, 29(6), 267-283. DOI: 10.1080/19393555.2020.1767240

[7] Verma, J., Bhandari, A., & Singh, G. (2022). Feature selection algorithm characterization for NIDS using machine and deep learning. In 2022 *IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS)*, 1-7 IEEE. DOI: 10.1109/IEMTRONICS55184.2022.9795709

[8] Pranto, M. B., Ratul, M. H. A., Rahman, M. M., Diya, I. J., & Zahir, Z.-B. (2022). Performance of machine learning techniques in anomaly detection with basic feature selection strategy-a network intrusion detection system. *Journal of Advances in Information Technology*, 13(1). DOI: 10.12720/jait.13.1.36-44

[9] Dey, S. K., & Rahman, M. M. (2018). Flow based anomaly detection in software defined networking: A deep learning approach with feature selection method. In *2018 4th International Conference on Electrical Engineering and Information & Communication Technology (iCEEiCT)*, 630-635. IEEE. DOI: 10.1109/CEEICT.2018.8628122

[10] Farhan, R. I., Maolood, A. T., & Hassan, N. (2021). Hybrid feature selection approach to improve the deep neural network on new flow-based dataset for NIDS. *Wasit Journal of Computer and Mathematics Science*, 66-83. DOI: 10.31185/wjcm.2021.1.1.6

[11] Li, X., Chen, W., Zhang, Q., & Wu, L. (2020). Building auto-encoder intrusion detection system based on random forest feature selection. *Computers & Security*, 95, 101851. DOI: 10.1016/j.cose.2020.101851

[12] Hammad, M., El-Medany, W., & Ismail, Y. (2020). Intrusion detection system using feature selection with clustering and classification machine learning algorithms on the UNSW-NB15 dataset. In *2020 International Conference on Innovation and Intelligence for Informatics, Computing and Technologies (3ICT)*, 1-6. IEEE. DOI: 10.1109/3ICT51146.2020.9311973

[13] Moustafa, N., & Slay, J. (2016). The evaluation of network anomaly detection systems: Statistical analysis of the UNSW-NB15 data set and the comparison with the KDD99 data set. *Information Security Journal: A Global Perspective*, 25(1-3), 18-31. DOI: 10.1080/19393555.2015.1125974

[14] Zong, W., Chow, Y.-W., & Susilo, W. (2018). A two-stage classifier approach for network intrusion detection. In *Information Security Practice and Experience: 14th International Conference, ISPEC 2018*, Tokyo, Japan, September 25-27, 329-340. Springer. DOI: 10.1007/978-3-319-99807-7_20

[15] Meftah, S., Rachidi, T., & Assem, N. (2019). Network based intrusion detection using the UNSW-NB15 dataset. *International Journal of Computing and Digital Systems*, 8(5), 478-487. DOI: 10.12785/ijcds/080505

[16] Kasongo, S. M., & Sun, Y. (2020). Performance analysis of intrusion detection systems using a feature selection method on the UNSW-NB15 dataset. *Journal of Big Data*, 7, 1-20. DOI: 10.1186/s40537-020-00379-6

[17] Roy, A., & Singh, K. J. (2021). Multi-classification of UNSW-NB15 dataset for network anomaly detection system. In *International Conference on Communication and Computational Technologies: ICCCT-2019*, 429-451. Springer. DOI: 10.1007/978-981-15-5341-7_38

[18] Eunice, A. D., Gao, Q., Zhu, M.-Y., Chen, Z., & Na, L. (2021). Network anomaly detection technology based on deep learning. In *2021 IEEE 3rd International Conference on Frontiers Technology of Information and Computer (ICFTIC)*, 6-9. IEEE. DOI: 10.1109/ICFTIC54222.2021.9644165

[19] Kocher, G., & Kumar, G. (2021). Analysis of machine learning algorithms with feature selection for intrusion detection using UNSW-NB15 dataset. *Available at SSRN 3784406*. DOI: 10.2139/ssrn.3784406

[20] Prasad, M., Gupta, R. K., & Tripathi, S. (2022). A multi-level correlation-based feature selection for intrusion detection. *Arabian Journal for Science and Engineering*, 47(8), 10719-10729. DOI: 10.1007/s13369-022-06601-4

[21] Yin, Y., Jang-Jaccard, J., Xu, W., Singh, A., Zhu, J., Sabrina, F., & Kwak, J. (2023). IGRF-RFE: A hybrid feature selection method for MLP-based network intrusion detection on UNSW-NB15 dataset. *Journal of Big Data*, 10(1), 1-26. DOI: 10.1186/s40537-023-00612-5

[22] Hemanth, D., et al. (2021). Intrusion detection system using convolutional neural network on UNSW NB15 dataset. *Advances in Parallel Computing Technologies and Applications*, 40, 1. DOI: 10.3233/APC210001

[23] Husain, A., Salem, A., Jim, C., & Dimitoglou, G. (2019). Development of an efficient network intrusion detection model using extreme gradient boosting (XGBoost) on the UNSW-NB15 dataset. In *2019 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)*, 1-7 . IEEE. DOI: 10.1109/ISSPIT47144.2019.9001844

[24] Alabrah, A. (2022). A novel study: GAN-based minority class balancing and machine-learning-based network intruder detection using chi-square feature selection. *Applied Sciences*, 12(22), 11662. DOI: 10.3390/app122211662

[25] Abd, S. N., Alsajri, M., & Ibraheem, H. R. (2020). Rao-SVM machine learning algorithm for intrusion detection system. *Iraqi Journal for Computer Science and Mathematics*, 1(1), 23-27. DOI: 10.52866/ijcsm.2020.01.01.003

[26] University of New South Wales. (2015). UNSW-NB15 dataset. DOI: 10.4225/35/55e4d5bfee1e5

[27] Ahmad, M., Riaz, Q., Zeeshan, M., Tahir, H., Haider, S. A., & Khan, M. S. (2021). Intrusion detection in internet of things using supervised machine learning based on application and transport layer features using UNSW-NB15 dataset. *EURASIP Journal on Wireless Communications and Networking*, 2021(1), 1-23. DOI: 10.1186/s13638-021-01893-8

[28] Alshaher, H. (2021). *Studying the effects of feature scaling in machine learning* PhD Dissertation, North Carolina Agricultural and Technical State University. Available at ProQuest: Link.

[29] Al-Sarem, M., Saeed, F., Alkhammash, E. H., & Alghamdi, N. S. (2021). An aggregated mutual information based feature selection with machine learning methods for enhancing IoT botnet attack detection. *Sensors*, 22(1), 185. DOI: 10.3390/s22010185

[30] Chen, R.-C., Dewi, C., Huang, S.-W., & Caraka, R. E. (2020). Selecting critical features for data classification based on machine learning methods. *Journal of Big Data*, 7(1), 52. DOI: 10.1186/s40537-020-00327-4

[31] Hasan, B. M. S., & Abdulazeez, A. M. (2021). A review of principal component analysis algorithm for dimensionality reduction. *Journal of Soft Computing and Data Mining*, 2(1), 20-30. DOI: 10.30880/jscdm.2021.02.01.003

[32] Fan, W., Liu, K., Liu, H., Wang, P., Ge, Y., & Fu, Y. (2020). AutoFS: Automated feature selection via diversity-aware interactive reinforcement learning. In *2020 IEEE International Conference on Data Mining (ICDM)*, 1008-1013. IEEE. DOI: 10.1109/ICDM50108.2020.00123

[33] Krishnaveni, S., Sivamohan, S., Sridhar, S. and Prabhakaran, S., 2022. Network intrusion detection based on ensemble classification and feature selection method for cloud computing. *Concurrency and Computation: Practice and Experience*, *34*(11), p.e6838. https://doi.org/10.1002/cpe.6838

[34] Aleesa, A., Younis, M., Mohammed, A. A., & Sahar, N. (2021). Deep-intrusion detection system with enhanced UNSW-NB15 dataset based on deep learning techniques. *Journal of Engineering Science and Technology*, 16(1), 711-727. Available at: Link
[35] Kasongo, S. M., & Sun, Y. (2019). A deep learning method with filter based feature engineering for wireless intrusion detection system. *IEEE Access*, 7, 38597-38607. DOI: 10.1109/ACCESS.2019.2905633
[36] Ahmed, A., El-Aasser, M., Ghantous, M. (2025). Analyzing the Effect of Feature Selection Algorithms on ML Classifiers. In: Abdelgawad, A., Jamil, A., Hameed, A.A. (eds) *Intelligent Systems, Blockchain, and Communication Technologies. ISBCom 2024*. Lecture Notes in Networks and Systems, vol 1268. Springer, Cham. https://doi.org/10.1007/978-3-031-82377-0_35
[37] Turukmane, Anil V., and Ramkumar Devendiran (2024). "M-MultiSVM: An efficient feature selection assisted network intrusion detection system using machine learning." *Computers & Security* 137: 103587. https://doi.org/10.1016/j.cose.2023.103587
[38] More, S., Idrissi, M., Mahmoud, H., & Asyhari, A. T. (2024). Enhanced Intrusion Detection Systems Performance with UNSW-NB15 Data Analysis. *Algorithms*, 17(2), 64. https://doi.org/10.3390/a17020064
[39] Jouhari, M., Benaddi, H., & Ibrahimi, K. (2024, July). Efficient Intrusion Detection: Combining X 2 Feature Selection with CNN-BiLSTM on the UNSW-NB15 Dataset. In *2024 11th International Conference on Wireless Networks and Mobile Communications (WINCOM)* (pp. 1-6). IEEE. https://doi.org/10.1109/WINCOM62286.2024.10658099
[40] Louppe, G. (2014). Understanding Random Forests: From Theory to Practice. arXiv:1407.7502 arXiv:1407.7502.
[41] Farrukh, Yasir Ali, Syed Wali, Irfan Khan, and Nathaniel D. Bastian (2025). Xg-nid: Dual-modality network intrusion detection using a heterogeneous graph neural network and large language model. *Expert Systems with Applications*, 287:128089. https://doi.org/10.1016/j.eswa.2025.128089
[42] Eljialy EM, Uddin MY, Ahmad S. (2024). Novel Framework for an Intrusion Detection System Using Multiple Feature Selection Methods Based on Deep Learning. *Tsinghua Science and Technology*, 29(4):948–958. https://doi.org/10.26599/TST.2023.9010032.

Authors' Profiles

Prof. Faruq Al-Omari received the B.Sc. and M.Sc. degrees in Computer Engineering from Jordan University of Science and Technology, Jordan, in 1990 and 1992, respectively, and the Ph.D. degree in Electrical and Computer Engineering from The University of Texas at Arlington, USA, in 1998. He is currently the Dean of the College of Engineering and Technology at the American University in the Emirates, Dubai, UAE. He has more than 28 years of academic and executive leadership experience in higher education. His research interests include artificial intelligence, machine learning, computer vision, medical image analysis, cybersecurity, educational technology, smart learning environments, and digital transformation in higher education.

Alaa Mhesin received the B.Sc. degree in Computer Engineering from Yarmouk University, Irbid, Jordan, and the M.Sc. degree in Security and Network Engineering from Jordan University of Science and Technology, Irbid, Jordan. She is currently a part-time Lecturer with the Department of Computer Engineering, Yarmouk University, Jordan. Her master's research focused on cybersecurity, and her research interests include cybersecurity, network security, information security, and artificial intelligence applications in cyber defense.

Mohammad M. Al-Shurman received the B.Sc. degree in electrical engineering from the Jordan University of Science and Technology, Irbid, Jordan, in 2000, and the M.Sc. and Ph.D. degrees in computer engineering from the University of Alabama in Huntsville, Huntsville, AL, USA, in 2003 and 2006, respectively. He is currently a Faculty Member with the Department of Network Engineering and Security at the Jordan University of Science and Technology, Irbid, Jordan. His research interests include computer networks, wireless and mobile ad hoc networks, network security, and cybersecurity. He has authored and co-authored numerous peer-reviewed publications in these areas.

How to cite this paper

Faruq A. Al-Omari, Alaa Y. Mhesin, Mohammad M. Al-Shurman, "Modified Multi-Stage Ensemble Feature Selection (MMSE-FS) for Network Intrusion Detection", International Journal of Wireless and Microwave Technologies(IJWMT), Vol.16, No.4, pp. 95-110, 2026. DOI:10.5815/ijwmt.2026.04.07