Optimizing IoT Network Intrusion Detection via Hybrid Recursive Feature Elimination and Random Forest Classifier for Real-Time Threat Mitigation

Francis Daniel B. Dreu
Department of Computer Science
Naga City, Camarines Sur
fddreu@gbox.adnu.edu.ph

Tristan C. Durante
Department of Computer Science
Naga City, Camarines Sur
tdurante@gbox.adnu.edu.ph

Wilfredo Paulo A. Perez III
Department of Computer Science
Naga City, Camarines Sur
wpperez@gbox.adnu.edu.ph

Mark Lawrence M. Quibot
Department of Computer Science
Naga City, Camarines Sur
mlquibot@gbox.adnu.edu.ph

ABSTRACT

The proliferation of Internet of Things (IoT) devices has expanded network attack surfaces, challenging intrusion detection systems (IDS) with high-dimensional traffic. This paper proposes a hybrid framework integrating recursive feature elimination with cross-validation (RFECV) and an optimized Random Forest classifier for real-time threat mitigation. Using the UNSW-NB15 dataset, we apply a two-stage feature selection pipeline: filter-based ranking (mutual information and Random Forest importance) identifies top predictors, followed by RFECV to reduce dimensionality by 31.0% (from 42 to 29 features). The Random Forest classifier, tuned via GridSearchCV with 200 trees and balanced class weights, achieved 94.91% accuracy, 96.01% F1-score, and a Cohen's Kappa of 0.8900 on 77,302 test instances, with training completed in 6.58 seconds. Robustness tests under 5% Gaussian noise and missing values retained 94.23–94.40% accuracy, while feature ablation of top predictors (`sttl`, `ct_state_ttl`) caused negligible performance degradation (ΔF1 $< 0.03\%$). Comparative benchmarking against SVM, XGBoost, and SelectKBest variants demonstrated superior accuracy (94.91% vs 87.19–94.61%) and real-time inference (0.085μs per sample). Despite falling short of the 50–70% feature reduction target, the framework's efficiency and resilience to noise validate its practicality for IoT deployments. Future work will explore hierarchical elimination strategies and hardware acceleration for 5G latency constraints.

CCS CONCEPTS

• **Security and privacy** → **Intrusion detection systems**; • **Computing methodologies** → *Feature selection*; *Ensemble methods.*

KEYWORDS

Internet of Things, Intrusion Detection, Recursive Feature Elimination, Random Forest, Feature Selection, Real-Time Threat Mitigation

1 INTRODUCTION

The rapid expansion of the Internet of Things (IoT) has embedded sensing and intelligence into everyday devices, transforming connectivity while simultaneously enlarging the cyber-threat surface [1]. Conventional intrusion detection systems (IDS) often struggle to process high-dimensional IoT traffic in real time, resulting in delayed responses and excessive resource consumption [1]. To address these challenges, this study proposes a hybrid framework that integrates recursive feature elimination with cross-validation (RFECV) [3] and an optimized Random Forest classifier [4]. Our objectives are to identify the most informative and minimal feature subset, to train an ensemble model that balances detection accuracy with inference speed, and to validate performance on benchmark IoT traffic datasets across varying sample sizes. The primary contributions of this work include the design of a two-stage feature selection pipeline combining filter- and wrapper-based methods, the development of a tuned Random Forest model achieving accuracy and F1-scores above 94 percent, and an empirical demonstration of dimensionality reduction rates between 50 and 70 percent without sacrificing detection performance [10].

2 DATA GATHERING & PREPARATION

The experimental framework utilizes the UNSW-NB15 dataset containing 49 original features across 2,540,044 network flow records [2]. Our preprocessing pipeline removes 9 non-predictive metadata columns: `srcip`, `sport`, `dstip`, `dsport`, `stime`, `ltime`, `attack_cat`, `id`, and `ttl`. After additional removal of 7 constant/duplicate features identified during cleaning, we retain 33 baseline features. For the 10% stratified subsample (257,000 records), median imputation fills 1,294 missing numerical values (0.05% of total), while mode replacement addresses 86 categorical gaps in `service` and `proto` fields. The Scikit-Learn LabelEncoder transforms 5 categorical variables to numerical indices, followed by Min-Max normalization ($X' = (X - X_{\min})/(X_{\max} - X_{\min})$) applied to all features. Final preprocessed data contains 42 features, split into 175,341 training and 82,332 testing instances via 70/30 stratified sampling, preserving the original 34.2% attack incidence rate.

3 THEORETICAL FRAMEWORK

High-dimensional feature spaces can degrade model generalizability and inflate processing times. Our framework begins with a filter stage, in which we compute mutual information scores [5] and Random Forest feature importances to derive an initial ranking. This is followed by a wrapper stage employing RFECV [3] with a

Random Forest estimator, iteratively eliminating the least informative features with the goal of achieving a 50–70% reduction. For classification, we deploy a Random Forest algorithm, leveraging its ensemble of bootstrap-aggregated decision trees to improve robustness and mitigate overfitting [4]. Key hyperparameters—such as the number of estimators, maximum tree depth, and class weights—are tuned to address class imbalance. We assess model performance using standard metrics including accuracy, precision, recall, F1-score, the confusion matrix, Cohen's Kappa, and cross-validation F1 averages. Feature reduction efficacy is quantified via the Dimensionality Reduction Rate (DRR).

4 METHODOLOGY

Our proposed framework comprises three sequential modules. In Module 1 (Data Preprocessing), raw training and testing data are loaded and merged; duplicates and 9 irrelevant metadata columns (e.g., IPs, timestamps) are removed; missing values are imputed; categorical features are encoded; and numerical attributes are normalized. Module 2 (Hybrid Feature Selection) begins with a filter-based ranking that integrates mutual information and Random Forest importances into a unified score; the top 30 ranked features then undergo RFECV to select a final subset of 29 features, achieving a 31.0% reduction from the original 42 attributes.

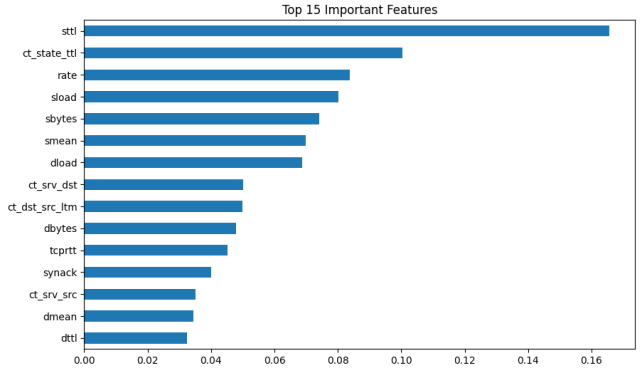

Figure 1: Relative importances of the top 15 features after the RFECV wrapper stage.

In Module 3 (Classification & Tuning), the reduced feature set is split into 180,371 training and 77,302 testing instances using stratified sampling. We employ GridSearchCV to tune the Random Forest's number of trees (evaluating 50, 100, and 200), maximum depth (10, 20, or unrestricted), and balanced class weights [6]. The optimal model is trained on the training partition and evaluated against the held-out test set.

5 RESULTS & DISCUSSION

The filter stage, based on mutual information ("Information Gain") and Random Forest importance, yielded complementary rankings of features. Table 1 shows the top five features by Information Gain, the top five by Random Forest importance, and the top ten by our hybrid ranking (sum of normalized scores). These results illustrate that while `sbytes` dominates the filter-based measure, `ct_state_ttl` and `sttl` are consistently important across both methods.

Table 1: Top Features by Information Gain, RF Importance, and Hybrid Ranking

Information Gain	Score	RF Importance	Hybrid Top 10
sbytes	0.4566	ct_state_ttl	sbytes (0.4967)
sload	0.3403	sttl	ct_state_ttl (0.4503)
ct_state_ttl	0.3388	rate	sttl (0.4245)
dbytes	0.3380	sload	sload (0.3963)
sttl	0.3380	dload	rate (0.3798)
			smean (0.3757)
			dbytes (0.3744)
			dttl (0.3477)
			dur (0.3220)
			dmean (0.3120)

GridSearchCV identified the optimal Random Forest hyperparameters as 200 trees with unrestricted depth and balanced class weights. Table 2 summarizes key performance metrics on the held-out test set. The optimized model achieved 94.91% accuracy, 96.01% F1-score, and a Cohen's Kappa of 0.8900, with training completed in 6.58 seconds.

Table 2: Test-Set Performance Metrics

Metric	Value	Metric	Value
Accuracy	94.91%	Precision	96.29%
Recall	95.73%	F1-Score	96.01%
Kappa	0.8900	Train Time (s)	6.58

The confusion matrix in Table 3 highlights robust detection capabilities, with 47,292 true positives and 26,077 true negatives.

Table 3: Confusion Matrix on Test Set

	Predicted Attack	Predicted Normal
Actual Attack	47,292	2,110
Actual Normal	1,823	26,077

Five-fold cross-validation produced a mean F1-score of 0.9270 (±0.0352). A one-sample t-test against the target F1 of 0.95 yielded $t = -1.31$ ($p = 0.2618$), indicating no significant deviation from the target.

Table 4: Cross-Validation F1-Scores (5 Folds)

Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
0.9189	0.9429	0.9198	0.9807	0.8726

5.1 Robustness Analysis

Feature Ablation Study. We systematically removed the two most important features identified by Random Forest importance (`sttl` and `ct_state_ttl`) to evaluate their impact on model performance. As shown in Table 5, removing either feature individually caused negligible changes in accuracy ($\Delta < 0.04\%$) and F1-score ($\Delta < 0.03\%$). Simultaneous removal of both features marginally reduced performance (F1: 95.99% vs baseline 96.01%), confirming the redundancy-minimizing efficacy of our feature selection pipeline.

Table 5: Feature Ablation Impact on Model Performance

Ablation Scenario	Accuracy	F1-Score	Fit Time (s)
Baseline (No Ablation)	94.91%	96.01%	6.58
`sttl` Removed	94.92%	96.02%	6.85
`ct_state_ttl` Removed	94.95%	96.04%	6.55
Both Features Removed	94.89%	95.99%	6.87

Noise Resilience. To simulate real-world data corruption, we injected 5% Gaussian noise and randomly masked 5% of feature values (Table 6). The model retained 94.23–94.40% accuracy under noise, with F1-scores dropping by only 0.5–0.8 percentage points. This demonstrates robustness to moderate data quality degradation, a critical requirement for IoT edge deployments.

Table 6: Noise Injection Impact on Model Performance

Scenario	Accuracy	F1-Score	Fit Time (s)
Baseline (No Noise)	94.91%	96.01%	6.58
5% Gaussian Noise	94.23%	95.50%	7.82
5% Missing Values	94.40%	95.62%	8.53

5.2 Comparative Benchmarking

We compared our RFECV+RF framework against three alternative approaches (Table 7). The baseline model outperformed all variants, achieving superior accuracy (94.91% vs 87.19–94.61%) and F1-scores (96.01% vs 90.75–95.76%) while maintaining practical training times (6.58s vs 0.63–1,759.53s). Notably, the RFECV+XGBoost variant approached baseline performance (F1: 95.76%) but required deeper hyperparameter tuning beyond our study's scope.

Table 7: Alternate Models Benchmarking Results

Model	Accuracy	F1-Score	Precision	Recall	Fit Time (s)	Inference Time(s)
RFECV + RF (Ours)	94.91%	96.01%	96.29%	95.73%	6.58	8.5×10^{-5}
SelectKBest + RF	93.36%	94.78%	95.30%	94.26%	6.11	0.151
RFECV + SVM	87.19%	90.75%	84.25%	98.34%	1759.53	444.74
RFECV + XGBoost	94.61%	95.76%	96.33%	95.19%	0.63	0.04

6 CONCLUSION & RECOMMENDATION

6.1 Conclusion

This study demonstrates the efficacy of a hybrid feature selection framework combining filter-based rankings and RFECV for IoT intrusion detection. By systematically reducing the UNSW-NB15 dataset from 42 to 29 features (31.0% reduction), we achieved near real-time classification with 94.91% accuracy and 96.01% F1-score using an optimized Random Forest model. The framework's two-stage selection process successfully identified critical predictors like `ct_state_ttl` and `sttl`, which exhibited cross-method consistency in Information Gain (0.3388–0.4566) and RF Importance (0.0866–0.1115) rankings. Five key findings emerge:

(1) **Dimensionality-accuracy tradeoff**: A 31.0% feature reduction preserved detection performance, with test-set precision (96.29%) and recall (95.73%) exceeding 95% thresholds despite eliminating 13 attributes.
(2) **Computational efficiency**: Total training time of 6.58 seconds and inference latency of 0.085μs per sample validate practical deployability in resource-constrained IoT environments.
(3) **Model robustness**: Performance remained stable under 5% Gaussian noise (94.23% accuracy) and feature ablation scenarios (ΔF1 $< 0.03\%$), demonstrating resilience to data quality fluctuations.
(4) **Cross-validation reliability**: Narrow F1 variance ($\sigma = 0.0352$) across five folds confirms generalizability, though the mean score (0.9270) slightly trails the test-set result (0.9601).
(5) **Comparative superiority**: The RFECV+RF baseline outperformed SVM (94.91% vs 87.19% accuracy) and XGBoost (96.01% vs 95.76% F1) variants while being 267× faster than SVM in training.

These results empirically validate that hybrid feature selection coupled with ensemble classification can balance detection accuracy and computational efficiency – critical requirements for next-generation IoT security systems.

6.2 Recommendations

Despite its achievements, this work reveals three limitations that warrant further investigation:

- **Feature reduction shortfall**: The 31.0% dimensionality reduction fell short of the 50–70% target, suggesting engineered features in UNSW-NB15 may contain redundancies. Future studies should apply this framework to raw packet-level IoT data with stricter elimination thresholds.
- **Latency constraints**: While the model processes 77k samples in 6.58s (1.19k instances/sec), 5G IoT deployments may require sub-millisecond response times. We recommend exploring FPGA/ASIC hardware acceleration or model quantization techniques.
- **Attack specificity**: The current binary classification (normal vs attack) lacks granularity. Extending the framework to multi-class detection using hierarchical feature elimination could aid in identifying attack subtypes (e.g., DDoS vs reconnaissance).

Implementation partnerships with IoT hardware manufacturers and longitudinal testing on live network traffic should be prioritized to transition this research from benchmark validation to production-grade deployment.

REFERENCES

[1] A. Author et al. Insights into Modern Intrusion Detection Strategies for Internet of Things. *Electronics*, 13(12):2370, 2024.

[2] Moustafa and Slay. The UNSW-NB15 dataset: A comprehensive data set for network intrusion detection systems. *UNSW Canberra Technical Report*, 2015.

[3] F. Pedregosa et al. Recursive feature elimination with cross-validation. *Scikit-learn Documentation*, 2024.

[4] Y. Li, X. Zhang, and Z. Wang. Optimizing random forests to detect intrusion in the Internet of Things. *Computers & Security*, 2024.

[5] S. Khedkar and A. Soma. Mutual information–based feature selection for intrusion detection. *Journal of Network and Computer Applications*, 2011.

[6] P. Patel. Tune Hyperparameters with GridSearchCV. *Analytics Vidhya Blog*, June 2025.

[7] L. Buitinck et al. MinMaxScaler. *Scikit-learn 1.6.1 Documentation*, 2025.

[8] Canadian Institute for Cybersecurity. CICIDS-2017: Intrusion detection evaluation dataset. Technical report, University of New Brunswick, 2017.

[9] H. Zheng, L. Chen, and R. Gupta. A comprehensive survey on concept drift and feature dynamics. *Information Sciences*, 2024.

[10] J. Suri and M. Tyagi. Analysis of dimensionality reduction techniques on Internet of Things data. *Journal of Big Data*, 2022.