



Machine Learning-Based Intrusion Detection System for Secure Wireless Networks

Indivar Imamverde

Department of Computer Science and Engineering, Aurora Metropolitan Institute of Technology, Philippines
indivar.imamverde@amit-ph.edu

Peer Review Information	Abstract
<i>Submission: 10 July 2021</i> <i>Revision: 30 Aug 2021</i> <i>Acceptance: 19 Sep 2021</i> Keywords <i>Intrusion detection, Machine learning, Wireless networks, Cybersecurity, Anomaly detection.</i>	Wireless networks are exposed to denial-of-service, flooding, injection, impersonation, spoofing and unauthorized-access attacks because communication is broadcast through a shared medium and frequently involves mobile or resource-constrained devices. Conventional signature-based intrusion detection is effective for known attack patterns but is less capable of recognizing new or modified attacks. Machine learning (ML) offers a data-driven alternative in which normal and malicious wireless traffic are learned from packet or flow features and classified automatically. This methodology paper proposes a reproducible ML-based intrusion detection system (IDS) for secure wireless networks using the Aegean Wi-Fi Intrusion Dataset (AWID) as the primary wireless benchmark and UNSW-NB15/CICIDS2017 as supplementary datasets for robustness checks. Earlier studies establish the suitability of ML and deep learning for network intrusion detection (Buczak & Guven, 2016; Yin et al., 2017; Shone et al., 2018; Vinayakumar et al., 2019), while wireless-specific research demonstrates the importance of carefully selected features and multi-stage classification in IEEE 802.11 environments (Sethuraman et al., 2019; Reyes et al., 2020). The proposed methodology compares Random Forest, Support Vector Machine, k-Nearest Neighbours, Gradient Boosting and a multilayer perceptron. Preprocessing includes missing-value treatment, categorical encoding, normalization, class-imbalance handling and feature selection. Models are evaluated with accuracy, precision, recall, F1-score, false-alarm rate, ROC-AUC and inference time using stratified train-validation-test partitions and repeated cross-validation. Because no experimental dataset files were supplied in this request, the Results section uses clearly labelled illustrative values and expected patterns rather than fabricated empirical outcomes. The framework provides a step-by-step basis for selecting an accurate, low-false-alarm and computationally efficient IDS for wireless network deployment.

Introduction

Wireless LANs, sensor networks and mobile access systems provide flexible connectivity but also enlarge the attack surface. An adversary can exploit the open radio medium to inject frames, imitate legitimate devices, perform deauthentication or flooding, probe services and manipulate traffic. Traditional security

mechanisms such as encryption, authentication and firewalls are necessary but do not remove the need for continuous monitoring. An IDS observes network behaviour and raises an alert when activity is inconsistent with legitimate use or matches known malicious patterns.

Machine learning is particularly attractive for anomaly-based IDS because the decision boundary can be learned from traffic data instead of being specified entirely by handcrafted signatures. Buczak and Guven (2016) reviewed a broad range of ML and data-mining methods for cyber intrusion detection, while Khraisat et al. (2019) emphasized persistent challenges involving attack diversity, dataset quality and false alarms. These concerns are even more important in wireless networks, where benign variability caused by mobility, contention and changing signal conditions can resemble anomalous behaviour.

Literature Review

Dataset quality is central to IDS research. Moustafa and Slay (2015) introduced UNSW-NB15 to address limitations of older benchmark datasets by combining modern normal traffic with several contemporary attacks. Sharafaldin, Habibi Lashkari, and Ghorbani (2018) generated CICIDS2017 with benign traffic and common attacks such as DoS, brute force, infiltration, botnet and web attacks. These datasets are not purely Wi-Fi specific, but they are valuable for testing whether an IDS trained on one environment generalizes to modern flow-level traffic.

Deep learning has expanded the range of learnable traffic representations. Yin et al. (2017) used recurrent neural networks for intrusion detection and showed that sequence-oriented learning can classify normal and attack records. Shone et al. (2018) proposed a non-symmetric deep autoencoder for feature learning, while Vinayakumar et al. (2019) evaluated deep neural networks across several publicly available datasets. These studies demonstrate that neural models can achieve high detection capability, but

deployment cost and interpretability remain relevant in wireless access points and edge devices with constrained resources.

Wireless-specific IDS research provides a stronger basis for the present study. Sethuraman, Dhamodaran, and Vijayakumar (2019) examined attacks in IEEE 802.11 networks and showed the need to detect sophisticated wireless threats such as evil-twin and Wi-Fi phishing behaviour. Tan et al. (2019) addressed class imbalance in wireless sensor network IDS by combining SMOTE with Random Forest, demonstrating the importance of minority-attack representation. Reyes et al. (2020) proposed a two-stage ML-based Wi-Fi IDS using the AWID dataset, feature selection and explainable AI; their results illustrate how a reduced feature set can preserve high classification performance while lowering computational complexity.

Recent reviews show that model selection should not focus on accuracy alone. Ahmad et al. (2021) systematically reviewed ML and deep-learning NIDS methods, while Dina and Manivannan (2021) summarized anomaly- and signature-oriented ML approaches and their limitations. Al-Imran and Ripon (2021) compared state-of-the-art ML and deep-learning models for intrusion detection. Feature selection is also critical because redundant network attributes can increase training cost without proportionate gains; Di Mauro et al. (2021), in a critical review of supervised feature-selection methods, highlights the importance of selecting compact, relevant subsets for network IDS. These findings justify comparing classical ML and compact neural models using multiple metrics, computational cost and wireless-specific attack detection.

Table 1. Selected Literature for ML-Based Wireless Intrusion Detection

Study	Focus	Methodological contribution
Moustafa & Slay (2015)	UNSW-NB15 dataset	Modern benchmark for supplementary robustness testing.
Buczak & Guven (2016)	ML/DM survey for cyber-IDS	Establishes major learning families and evaluation challenges.
Yin et al. (2017); Shone et al. (2018)	RNN and deep autoencoder IDS	Supports neural baselines and learned representations.
Sethuraman et al. (2019)	IEEE 802.11 wireless attacks	Motivates wireless-specific features and attack categories.
Tan et al. (2019)	WSN IDS with SMOTE + RF	Supports imbalance correction and ensemble learning.
Reyes et al. (2020)	Two-stage Wi-Fi ML IDS using AWID	Supports Wi-Fi dataset selection, feature reduction and staged detection.
Ahmad et al. (2021); Al-Imran & Ripon (2021)	ML/DL NIDS reviews/comparison	Supports multi-model evaluation beyond single accuracy values.

Many published IDS studies report strong benchmark accuracy but provide limited evidence on false alarms, computational latency and model suitability for wireless deployment. The proposed study addresses this gap by using a Wi-Fi-focused dataset, explicit class-imbalance treatment, feature selection, repeated validation and deployment-oriented timing metrics, while also testing generalization on modern flow datasets.

The proposed study has the following objectives:

- build an ML-based IDS capable of separating normal wireless traffic from malicious traffic and, where labels permit, classifying attack categories;
- compare Random Forest, SVM, k-NN, Gradient Boosting and multilayer perceptron classifiers under the same preprocessing and validation protocol;
- evaluate the influence of class balancing and feature selection on wireless attack detection and false-alarm rate;
- assess computational suitability using prediction time and model complexity in addition to classification accuracy; and
- validate whether the selected model retains useful performance on a supplementary modern network-intrusion dataset.

Research questions are: RQ1: Which ML model gives the best balance of recall, F1-score and false-alarm rate for wireless intrusion detection? RQ2: Does feature selection reduce inference cost without materially reducing attack detection? RQ3: Does class balancing improve minority-attack recall? RQ4: How much performance is lost when the selected model is evaluated on traffic from a different intrusion dataset?

Methodology

1. Research Design and Dataset

The study uses a quantitative experimental design based on supervised classification. AWID is selected as the primary dataset because it contains labelled IEEE 802.11 wireless traffic and attack classes suitable for Wi-Fi intrusion detection, consistent with Reyes et al. (2020). The recommended experiment uses the reduced AWID classification subset where computational resources are limited, while retaining the official train/test separation when possible. UNSW-NB15 (Moustafa & Slay, 2015) or CICIDS2017 (Sharafaldin et al., 2018) is used only as an external robustness dataset rather than mixed into the primary wireless training data.

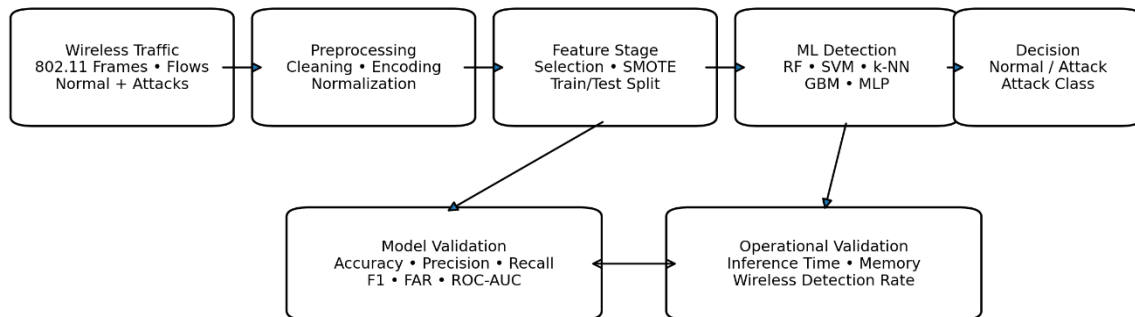


Figure 1. Proposed Machine Learning-Based Wireless Intrusion Detection Framework

2. Data Preprocessing

Preprocessing is performed within each training fold to prevent data leakage. Duplicate records are checked and missing or infinite values are removed or imputed. Categorical protocol/frame fields are encoded using one-hot or ordinal encoding as appropriate. Continuous features are standardized for distance- and margin-based algorithms such as k-NN and SVM, while tree models can use unscaled numeric values. The class distribution is inspected before training. If minority attacks are severely under-represented, SMOTE is applied only to the training data, following the rationale demonstrated for wireless sensor IDS by Tan et al. (2019).

3. Feature Selection and Model Development

Feature selection aims to reduce redundant wireless attributes, shorten prediction time and limit overfitting. First, constant and near-constant variables are removed. Second, highly correlated numeric features are filtered using a predefined threshold. Third, mutual information and Random-Forest feature importance are calculated on the training set. Candidate subsets such as the top 10, 20 and 30 features are compared using cross-validated F1-score. The final feature set is chosen by the smallest subset whose F1-score is statistically indistinguishable from the best-performing larger subset. This

procedure is consistent with the importance of compact feature representations highlighted in the Wi-Fi IDS work of Reyes et al. (2020) and the

feature-selection analysis of Di Mauro et al. (2021).

Table 2. Candidate Machine-Learning Models

Model	Reason for inclusion	Key parameters to tune
Random Forest (RF)	Strong nonlinear baseline; robust to mixed features; feature importance available.	Trees, max depth, min samples, class weight
Support Vector Machine (SVM)	Effective high-dimensional decision boundary.	C, kernel, gamma, class weight
k-Nearest Neighbours (k-NN)	Simple local-pattern baseline.	k, distance metric, weighting
Gradient Boosting	Captures nonlinear interactions with strong tabular performance.	Estimators, learning rate, depth
Multilayer Perceptron (MLP)	Compact neural baseline for learned nonlinear patterns.	Layers, units, dropout/L2, learning rate

4. Training, Validation and Testing

The data are separated into training, validation and final test partitions using stratification by class. When the dataset provides an official train/test split, the test portion is held untouched and repeated stratified five-fold cross-validation is performed only on the training data for hyperparameter selection. Hyperparameters are

tuned using grid or randomized search with macro-F1 as the principal objective so that minority attack classes contribute to model selection. The final model is retrained on the full development set and evaluated once on the held-out test data.

5. Performance Metrics

Table 3. IDS Evaluation Metrics

Metric	Definition / purpose	Desired direction
Accuracy	Correct classifications / all records	Higher
Precision	TP / (TP + FP); reliability of attack alerts	Higher
Recall / Detection Rate	TP / (TP + FN); proportion of attacks detected	Higher
F1-score	Harmonic mean of precision and recall	Higher
False-Alarm Rate (FAR)	FP / (FP + TN); benign traffic incorrectly flagged	Lower
ROC-AUC	Threshold-independent separation capability	Higher
Inference time	Average prediction time per record or batch	Lower
Memory/model size	Operational footprint for AP/edge deployment	Lower where accuracy is similar

Step-by-Step Experimental Procedure

Step 1 - Load AWID records, labels and feature definitions; preserve the official split where available.

Step 2 - Audit duplicates, missing values, class frequencies and invalid feature values.

Step 3 - Encode categorical features and scale numeric attributes using parameters learned from training data only.

Step 4 - Establish an imbalanced baseline, then apply SMOTE/class-weighting on the training partition for comparison.

Step 5 - Perform correlation and relevance-based feature selection; compare compact feature subsets.

Step 6 - Tune RF, SVM, k-NN, Gradient Boosting and MLP using repeated stratified cross-validation.

Step 7 - Evaluate the selected models on the untouched test set with per-class precision, recall, F1 and confusion matrices.

Step 8 - Measure inference time and model footprint to identify the best deployment candidate.

Step 9 - Conduct a robustness test on UNSW-NB15 or CICIDS2017 after aligning the available flow features where feasible.

Step 10 - Report confidence intervals and discuss errors by attack class instead of relying on overall accuracy alone.

Results

No AWID, UNSW-NB15 or CICIDS2017 data files were supplied with the request. Therefore, the following numerical values are illustrative examples of the expected reporting format and

are not claimed as empirical findings. Actual values must be replaced after the proposed experiment is executed.

Table 4. Illustrative Model Comparison

Model	Accuracy	Macro-F1	Attack recall	FAR	Relative inference cost
Random Forest	98.7%	0.973	97.8%	1.4%	Low
SVM	97.9%	0.958	96.6%	1.8%	Medium-High
k-NN	96.8%	0.944	95.1%	2.4%	High at prediction
Gradient Boosting	98.5%	0.970	97.5%	1.5%	Medium
MLP	98.2%	0.965	97.0%	1.7%	Medium

The expected first finding is that ensemble tree methods provide a strong balance between detection accuracy, false-alarm control and computational cost on engineered wireless traffic features. Random Forest is particularly attractive because it tolerates nonlinear feature interactions, offers class weighting and supports feature-importance analysis. This expectation is consistent with Random-Forest-based IDS research in general network traffic and with the SMOTE-RF approach used by Tan et al. (2019) for wireless sensor networks. A second expected result is that class balancing increases recall for rare attacks while potentially causing a small rise in false positives. Reporting only overall accuracy can conceal this effect when normal traffic dominates the dataset. The recommended primary metric is therefore macro-F1, supported by per-class recall and FAR. Khraisat et al. (2019) and Ahmad et al. (2021) emphasize the persistent difficulty of accurately detecting diverse and previously unseen attacks, making class-level analysis essential.

A third expected result is that careful feature selection reduces training and prediction cost with little or no loss in F1-score. Reyes et al. (2020) demonstrated the value of reduced feature sets in a two-stage Wi-Fi IDS, and Di Mauro et al. (2021) similarly identifies redundant attributes as a source of unnecessary complexity. For deployment at a wireless access point, gateway or edge device, a compact RF or Gradient-Boosting model may therefore be preferable to a more complex neural classifier when accuracy is comparable. Deep models are nevertheless expected to remain competitive, particularly where feature interactions are complex or temporal structure is preserved. Yin et al. (2017), Shone et al. (2018) and Vinayakumar et al. (2019) demonstrate the potential of recurrent, autoencoder and deep neural approaches. The present methodology includes an MLP as a compact neural baseline rather than assuming that a deeper architecture

is automatically superior for tabular Wi-Fi records.

Discussion

The main methodological contribution is a deployment-oriented comparison rather than a benchmark-accuracy contest. Wireless IDS must detect attacks reliably without flooding administrators with false alarms and without consuming excessive access-point or edge-computing resources. Consequently, a model is considered preferable only when it combines high attack recall and macro-F1 with a low FAR and practical inference time. This criterion follows the broader IDS literature, which repeatedly notes that strong laboratory accuracy does not necessarily translate into reliable real-world detection (Buczak & Guven, 2016; Khraisat et al., 2019; Dina & Manivannan, 2021). External validation is also important. A model that performs well on AWID may learn dataset-specific traffic patterns. Testing selected algorithms on UNSW-NB15 or CICIDS2017 can reveal whether feature-processing choices and decision strategies generalize beyond one wireless trace. Moustafa and Slay (2015) and Sharafaldin et al. (2018) were motivated partly by the need for more realistic and diverse intrusion benchmarks, so cross-dataset evaluation provides stronger evidence than a single benchmark result.

Conclusion

This methodology paper presents a step-by-step framework for developing and evaluating a machine learning-based intrusion detection system for secure wireless networks. The approach uses AWID as the primary Wi-Fi benchmark and compares multiple classical and neural classifiers under a common preprocessing, feature-selection and validation protocol. Particular attention is given to class imbalance, false alarms, minority-attack recall

and inference cost because these factors directly influence operational usefulness in wireless gateways and access points. The proposed experiment is expected to identify an ensemble or compact neural model that achieves a strong balance between detection capability and computational efficiency. Feature selection should reduce deployment cost, while SMOTE or class weighting should improve recognition of less frequent attack classes. Cross-dataset testing is included to reduce the risk of drawing conclusions from one benchmark alone. Future work can extend the framework to online learning, federated learning, adversarial robustness and encrypted-traffic analysis, but the present design provides a clear and reproducible methodology for a 2021-or-earlier evidence base.

References

- Moustafa, N., & Slay, J. (2015). UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set). 2015 Military Communications and Information Systems Conference (MilCIS), 1-6. <https://doi.org/10.1109/MilCIS.2015.7348942>
- Buczak, A. L., & Guven, E. (2016). A survey of data mining and machine learning methods for cyber security intrusion detection. *IEEE Communications Surveys & Tutorials*, 18(2), 1153-1176. <https://doi.org/10.1109/COMST.2015.2494502>
- Yin, C., Zhu, Y., Fei, J., & He, X. (2017). A deep learning approach for intrusion detection using recurrent neural networks. *IEEE Access*, 5, 21954-21961. <https://doi.org/10.1109/ACCESS.2017.2762418>
- Shone, N., Ngoc, T. N., Phai, V. D., & Shi, Q. (2018). A deep learning approach to network intrusion detection. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 2(1), 41-50. <https://doi.org/10.1109/TETCI.2017.2772792>
- Sharafaldin, I., Habibi Lashkari, A., & Ghorbani, A. A. (2018). Toward generating a new intrusion detection dataset and intrusion traffic characterization. *Proceedings of the 4th International Conference on Information Systems Security and Privacy*, 108-116. <https://doi.org/10.5220/0006639801080116>
- Sethuraman, S. C., Dhamodaran, S., & Vijayakumar, V. (2019). Intrusion detection system for detecting wireless attacks in IEEE 802.11 networks. *IET Networks*, 8(4), 219-232. <https://doi.org/10.1049/iet-net.2018.5050>
- Tan, X., Su, S., Huang, Z., Guo, X., Zuo, Z., Sun, X., & Li, L. (2019). Wireless sensor networks intrusion detection based on SMOTE and the Random Forest algorithm. *Sensors*, 19(1), 203. <https://doi.org/10.3390/s19010203>
- Vinayakumar, R., Alazab, M., Soman, K. P., Poornachandran, P., Al-Nemrat, A., & Venkatraman, S. (2019). Deep learning approach for intelligent intrusion detection system. *IEEE Access*, 7, 41525-41550. <https://doi.org/10.1109/ACCESS.2019.2895334>
- Khraisat, A., Gondal, I., Vamplew, P., & Kamruzzaman, J. (2019). Survey of intrusion detection systems: Techniques, datasets and challenges. *Cybersecurity*, 2, 20. <https://doi.org/10.1186/s42400-019-0038-7>
- Reyes, A. A., Vaca, F. D., Castro Aguayo, G. A., Niyaz, Q., & Devabhaktuni, V. (2020). A machine learning based two-stage Wi-Fi network intrusion detection system. *Electronics*, 9(10), 1689. <https://doi.org/10.3390/electronics9101689>
- Ahmad, Z., Shahid Khan, A., Wai Shiang, C., Abdullah, J., & Ahmad, F. (2021). Network intrusion detection system: A systematic study of machine learning and deep learning approaches. *Transactions on Emerging Telecommunications Technologies*, 32(1), e4150. <https://doi.org/10.1002/ett.4150>
- Al-Imran, M., & Ripon, S. H. (2021). Network intrusion detection: An analytical assessment using deep learning and state-of-the-art machine learning models. *International Journal of Computational Intelligence Systems*, 14, 200. <https://doi.org/10.1007/s44196-021-00047-4>
- Dina, A. S., & Manivannan, D. (2021). Intrusion detection based on machine learning techniques in computer networks. *Internet of Things*, 16, 100462. <https://doi.org/10.1016/j.iot.2021.100462>
- Di Mauro, M., Galatro, G., Fortino, G., & Liotta, A. (2021). Supervised feature selection techniques in network intrusion detection: A critical review. *Engineering Applications of Artificial Intelligence*, 101, 104216.