Unsupervised Anomaly Detection for Wi-Fi Networks using RFFI

Xinyi Li
Faculty of Computer Science
Dalhousie University
Halifax, Canada
xn394804@dal.ca

Samer Lahoud
Faculty of Computer Science
Dalhousie University
Halifax, Canada
sml@dal.ca

Nur Zincir-Heywood
Faculty of Computer Science
Dalhousie University
Halifax, Canada
zincir@cs.dal.ca

***Abstract*—Radio Frequency Fingerprinting Identification (RFFI) offers a promising physical-layer approach to detect device impersonation in Wi-Fi networks by exploiting unique hardware-induced signal distortions. In this work, we develop an unsupervised anomaly detection framework based on autoencoders trained exclusively on legitimate device signals. We evaluate both raw IQ samples and manually extracted features, conducting detailed feature analysis to identify the most discriminative hardware impairments. Our results across three diverse datasets, synthetic, emulated, and real-world Wi-Fi signals, demonstrate that extracted features consistently outperform raw IQ inputs, achieving up to 97.36% impersonation detection accuracy and an average F1-score of 0.94 using a reduced feature set. These findings validate the effectiveness of our feature selection and highlight the suitability of unsupervised learning for real-world Wi-Fi security scenarios.**

***Index Terms*—Impersonation Attack Detection, Radio Frequency Fingerprinting Identification (RFFI), Unsupervised Learning**

I. Introduction

Wi-Fi, based on the IEEE 802.11 standard, is widely deployed in personal, business, and public networks, supporting high-speed data transmission and flexible network access. Given its ubiquity, Wi-Fi networks are vulnerable to attacks, making security a critical concern. Although some protocol-level protections such as WPA3 encryption and 802.1x mutual authentication exist, MAC-layer spoofing attacks remain possible [1]–[3]. Specifically, in impersonation attacks, an attacker hijacks the MAC address of a legitimate device to gain unauthorized access. To address this limitation, Radio Frequency Fingerprinting Identification (RFFI) has gained significant attention. RFFI leverages inherent hardware impairments in devices from radio signals, which is represented by In-phase and Quadrature (IQ) data, a complex-valued sequence capturing both amplitude and phase. Some possible impairments include carrier frequency offset (CFO), IQ imbalance and phase noise, which arise during manufacturing and are extremely difficult to replicate [4]. The impairments are protocol-independent and unique even across devices from the same production line [4], making them effective fingerprints for device identification. Methods that utilize RFFI can be broadly categorized as engineered feature-based, which manually extract fingerprints from IQ signals, and deep learning-based, which automatically learn device fingerprints from raw signals [4].

In this work, we propose an unsupervised autoencoder-based anomaly detection method under a centralized learning setting to capture fingerprints of legitimate devices based on RFFI. Supervised learning is difficult in dynamic Wi-Fi networks due to frequent device changes and scarce labels for impersonation attacks. Unsupervised learning is therefore suitable, as it does not require labeled attack data. Our method detects impersonation attacks using manually extracted fingerprint features. We also conduct feature analysis to show that using extracted features, rather than raw IQ data alone, improves device identification and classification. We validate our impersonation detection approach on three datasets: Signal-level Synthetic Dataset, Protocol-level Emulated Dataset, and WiSig Real-world Dataset [5]. While our experiments focus on Wi-Fi 802.11a/g, the proposed unsupervised approach is general and can be applied to other wireless protocols.

The key contributions of this paper are:

- We identify and analyze the most significant features and hardware impairments for RFFI, providing insights into the contributions of different hardware impairments.
- We propose an autoencoder-based anomaly detection model that captures device fingerprints and detects impersonation attacks by only using signals for legitimate devices without labels. We use two types of input data for comparison: one is raw IQ signals, and the other is extracted features. Our method is evaluated on three datasets to demonstrate its generalizability.
- We perform comprehensive comparative evaluations across multiple datasets and different machine learning algorithms, including XGBoost, K-Nearest Neighbours (KNN) and autoencoder, highlighting the performance differences and the effectiveness of our feature extraction approach in various settings.

The remainder of the paper is organized as follows. Section II reviews related work. Section III presents the methodology, including feature extraction and anomaly detection. Section IV details experiments and results. Section V concludes the paper.

II. Related Work

Anomaly detection methods can use either signal-level or protocol-level information. Among them, RFFI, as a signal-

978-3-903176-75-1 ©2025 IFIP

level method, is particularly promising, as it identifies devices based on unique hardware impairments in wireless signals. Existing approaches fall into two categories: raw IQ-based and extracted feature-based. Raw IQ methods directly apply deep learning to IQ data. For example, the authors in [6] predefined IQ imbalance and DC offset impairments, achieving 80–95% accuracy by using Convolutional Neural Network (CNN). The work [7] also used CNNs on simulated impairments, reporting 89–99% accuracy. However, reliance on simulations or artificial impairments limits real-world deployment. Another line of research focuses on features extracted from hardware fingerprints rather than raw IQ data, divided into supervised and unsupervised approaches. In supervised learning, the work [8] extracts the Envelope Power Spectrum (EPS) based on CFO, robust across time, channel, and location. Neural networks can automatically extract fingerprints, as in [9], which uses a CNN to extract fingerprints from distorted and undistorted signals, and a U-Net to reconstruct the distorted signal. Contrastive learning is used in [10], [11], with data augmentation and federated frameworks in the former and a modified ResNet-18 encoder in the latter. Additionally, [12] proposes static distance and GAN-based models for different receivers. All these methods [8]–[12] are validated on seen devices through classification tasks. Unsupervised fingerprinting is less explored, but work [13] shows that a complex-valued CNN trained on multiple devices can extract fingerprints for unseen devices, verified via DBSCAN clustering.

Although supervised learning can achieve good results, it struggles in real Wi-Fi networks due to dynamic device mobility, making labels hard to maintain [14]. Impersonation attacks mimic legitimate devices, reducing Received Signal Strength (RSS)-based detection effectiveness [15], and the scarcity of attack samples causes class imbalance and poor generalization [16]. Unsupervised learning avoids these issues by not requiring attack labels. Motivated by this, we propose an autoencoder-based anomaly detection using extracted Wi-Fi features to detect impersonation attacks without labeled data.

III. Autoencoder-based Unsupervised Learning Method for Anomaly Detection

This section presents an overview of our proposed method for impersonation attack detection using an unsupervised autoencoder-based anomaly detection model.

A. System Architecture

Our approach leverages RFFI to detect impersonation attacks using autoencoders. Autoencoders are unsupervised methods, learning patterns from legitimate device signals without labeled attack data by reconstructing inputs and minimizing reconstruction error. As shown in Fig. 1, two alternative workflows can be used. One uses a CNN-based autoencoder to process raw IQ signals, handling the real and imaginary parts directly, while the other uses a fully connected autoencoder on extracted features, including time-domain, frequency-domain, and hardware impairment features. The chosen model is trained on legitimate signals and tested with both legitimate and attack samples. Impersonation attacks are detected by comparing reconstruction errors with a threshold. The threshold is set as the average reconstruction error of the training data multiplied by a factor between 1 and 2 to reduce false positives caused by minor variations in legitimate signals. In prior autoencoder-based anomaly detection studies [17]–[19], thresholds are typically adjusted rather than taken directly from the average, highlighting the importance of proper threshold selection.

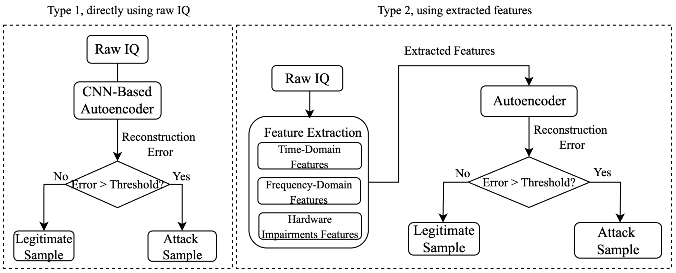

Fig. 1. System Architecture.

B. Feature Extraction

To effectively characterize the unique fingerprint of Wi-Fi signals and improve impersonation attack detection, we manually extract 24 features from the raw IQ data, categorized into three groups: time-domain, frequency-domain, and hardware impairment-related features. Although grouped, these features are interrelated and interact with each other.

The time-domain features capture statistical properties of the in-phase (I) and quadrature (Q) components. Specifically, we compute the mean, standard deviation, skewness, and kurtosis for both real and imaginary parts of the IQ samples. While mean and standard deviation capture general trends, skewness and kurtosis provide information about asymmetry and heavy-tailed distributions in the signal, often induced by device-specific hardware impairments such as phase noise. Frequency-domain features capture spectral characteristics of the transmitted signal. We perform a Fast Fourier Transform (FFT) on the IQ samples and compute spectral mean, spectral standard deviation, spectral flatness, spectral entropy, and spectral slope. For hardware impairment-related features, we include IQ imbalance ratio, CFO variance, phase noise skewness and kurtosis, error vector magnitude, autocorrelation of both real and imaginary parts, and energy variation. These features help account for device-specific impairments that influence the RF fingerprint.

C. Model Design

The proposed approach uses two distinct autoencoder architectures to process different types of data. One autoencoder is designed for processing raw IQ data using convolutional layers, while the other operates on extracted features using fully connected layers. The detailed structures are shown in Fig. 2. For the both autoencoders, since the input length is not fixed, the number of intermediate nodes needs to be adjusted accordingly. The figure shows the case with an input length of 80 for CNN-based autoencoders and 24 for fully connected autoencoders.

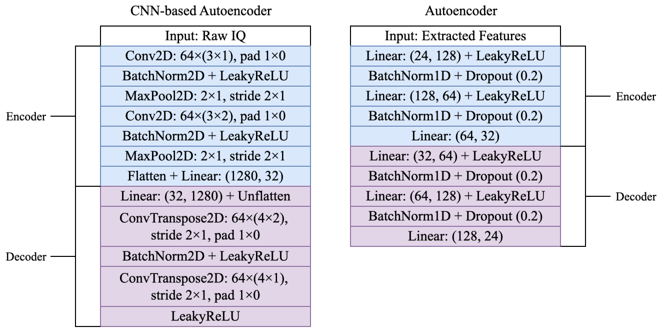
CNN-based Autoencoder
Input: Raw IQ
Conv2D: 64×(3×1), pad 1×0
BatchNorm2D + LeakyReLU
MaxPool2D: 2×1, stride 2×1
Conv2D: 64×(3×2), pad 1×0
BatchNorm2D + LeakyReLU
MaxPool2D: 2×1, stride 2×1
Flatten + Linear: (1280, 32)
Linear: (32, 1280) + Unflatten
ConvTranspose2D: 64×(4×2), stride 2×1, pad 1×0
BatchNorm2D + LeakyReLU
ConvTranspose2D: 64×(4×1), stride 2×1, pad 1×0
LeakyReLU
Encoder
Decoder
Autoencoder
Input: Extracted Features
Linear: (24, 128) + LeakyReLU
BatchNorm1D + Dropout (0.2)
Linear: (128, 64) + LeakyReLU
BatchNorm1D + Dropout (0.2)
Linear: (64, 32)
Linear: (32, 64) + LeakyReLU
BatchNorm1D + Dropout (0.2)
Linear: (64, 128) + LeakyReLU
BatchNorm1D + Dropout (0.2)
Linear: (128, 24)
Encoder
Decoder

Fig. 2. Model Architecture.

IV. Experiments and Results

We evaluate our approach through feature analysis and anomaly detection. Feature analysis is based on Signal-level Synthetic Data and WiSig Real-world Data. The first examines how hardware impairments affect device separability, while the second identifies important features in real-world conditions. Anomaly detection then leverages these findings to improve performance.

A. Data Generation

We use three datasets to evaluate our method: Signal-level Synthetic Data, Protocol-level Emulated Data, and WiSig Real-world Data [5]. The first two were generated in our lab, while WiSig is publicly available.

Signal-level Synthetic Data allows controlled analysis of hardware impairments. Signals are modulated using 16-QAM and processed through orthogonal frequency-division multiplexing (OFDM) with a cyclic prefix. Device-specific CFO, IQ imbalance, phase noise, and non-linear distortion are applied, followed by an additive white Gaussian noise (AWGN) channel and Rayleigh fading to simulate multipath effects. Protocol-level Emulated Data introduces realistic WiFi protocol behavior using GNU Radio while preserving hardware impairments. Bitstreams with configurable MAC/IP addresses are encoded, modulated via OFDM, and transmitted through a simulated wireless channel. WiSig Real-world Data [5] reflects practical conditions with multiple transmitters and receivers. We use the *SingleDay* subset, which is pre-processed, balanced, and contains only preamble signals. Note that the number of complex numbers in the original IQ sample differs across datasets: 80 per sample for Signal-level, 3840 per sample for Protocol-level, and 256 per sample for WiSig.

For all datasets, we split the data 8:1:1 for training, validation, and testing for impersonation detection. Anomalies appear only in the test set. In the Singal-level and Protocol-level dataset, anomalies reuse legitimate addresses with distinct hardware impairments and random payloads. While in the WiSig dataset, anomalies come from devices excluded from training, as all the devices use the same address which fits into impersonation scenario. Table I summarizes the number of samples used in our experiment, which is a subset. The whole Signal-level Synthetic Dataset and Protocol-level Emulated Dataset are available at the GitHub repository: https://github.com/lxy3300/RFFI-dataset-for-anomaly-detection.

TABLE I
Summary of Datasets

Dataset	Train / Val / Test	Attack Records (Test)	Legitimate Records (Test)
Signal-level Synthetic	4,000 / 500 / 500	250	250
Protocol-level Emulated	2,400 / 300 / 300	150	150
WiSig Real-world	11,200 / 1,400 / 1,400	700	700

B. Feature Analysis

1) Separable Hardware Impairments: To study the effect of hardware impairments, we generated Signal-level Synthetic Dataset for eight devices, varying four impairments (CFO, IQ imbalance, phase noise, and non-linear distortion). We used XGBoost and KNN for classification, both of which have demonstrated strong performance in prior work [20]–[24]. A total of 800 packets were generated, 100 per device.

Results in Table II show that only IQ imbalance variation significantly improves device separability. XGBoost achieves 92.5% accuracy with an F1-score of 0.925, while KNN reaches 65.6% accuracy with an F1-score of 0.661. Variation in CFO, phase noise, or non-linear distortion alone does not yield high accuracy. Feature importance analysis based on XGBoost indicates that the standard deviation of the real part of the signal, the spectral slope, and the IQ imbalance ratio are the dominant features. The standard deviation of the real part captures fluctuations in the in-phase component that arise from device-specific noise levels and nonlinearities in the RF front-end. The spectral slope reflects the decay of the power spectrum across frequency, which is influenced by oscillator phase noise. The IQ imbalance ratio directly measures amplitude and phase mismatches between the I and Q branches of the transceiver. These results confirm that hardware-induced impairments, particularly IQ imbalance, are the most critical factors for distinguishing devices.

TABLE II
Classification results for Signal-level Synthetic Data

Method	Settings	Accuracy	Precision	Recall	F1-score
XGBoost	CFO Variation	0.6312	0.6352	0.6313	0.6307
	IQ Imbalance Variation	**0.9250**	0.9276	0.9250	**0.9248**
	Phase Noise Variation	0.5000	0.5016	0.5000	0.4996
	Non-Linear Distortion Variation	0.6750	0.6751	0.6750	0.6711
KNN	CFO Variation	0.4437	0.4436	0.4195	0.4257
	IQ Imbalance Variation	**0.6562**	0.6711	0.6684	**0.6606**
	Phase Noise Variation	0.3812	0.3886	0.3785	0.3656
	Non-Linear Distortion Variation	0.5687	0.5282	0.5418	0.5284

2) Feature analysis based on WiSig Real-world Dataset: We then analyzed data from eight WiSig transmitters (100 packets each) received by four receivers. Classification with XGBoost and KNN was performed using all 24 features and a reduced subset of the top 12 features based on XGBoost.

Table III shows that XGBoost maintains high performance, achieving an F1-score of 0.9811 with either feature set. KNN benefits more from feature selection, with accuracy improving from 56.9% to 74.4% and F1-score increasing from 0.576 to 0.732. Analysis of feature contributions on real-world data shows that amplitude-related features (standard deviations of the I/Q components) and spectral features (spectral mean, flatness, and entropy) dominate. Additional higher-order statistics such as kurtosis of the real part and sparsity also contribute, reflecting diverse hardware impairments. These results highlight that in real environments, the impact of hardware impairments is more complex and less reproducible, which strengthens their value for device identification.

These results demonstrate that removing less informative features based on feature analysis substantially improves classification performance, particularly for simpler models like KNN, while confirming the importance of IQ imbalance and amplitude and spectral metrics for device separability.

TABLE III
CLASSIFICATION RESULTS FOR WISIG DATA

Method	Settings	Accuracy	Precision	Recall	F1-score
XGBoost	Full Feature Set	0.9812	0.9821	0.9812	**0.9811**
	Reduced Feature Set	0.9812	0.9821	0.9812	**0.9811**
KNN	Full Feature Set	0.5687	0.5817	0.5792	0.5755
	Reduced Feature Set	0.7438	0.7371	0.7317	**0.7318**

C. *Impersonation Attack Detection*

We evaluate our method on three datasets, training on legitimate samples and testing on both legitimate and attack samples under a centralized setting. For each experiment, training is performed on legitimate samples and testing is performed on both legitimate and attack samples, with hyperparameter settings shown in Table IV. All parameters, including the threshold factor, are selected based on Hyperband optimization [25] results combined with manual adjustments from experimental tuning.

TABLE IV
TRAINING HYPERPARAMETERS FOR DIFFERENT DATASETS

Dataset	Training Data	Epoch	LR	Batch	Thresh
Signal-level Synthetic Data	Raw IQ (80)	10	0.0005	32	1.5
	Ext. Features (Full)	10	0.0005	32	1.5
	Ext. Features (Reduced)	50	0.0004	128	1.5
Protocol-level Emulated Data	Raw IQ (3840)	30	0.0001	32	1.5
	Ext. Features (Full)	50	0.0005	64	2
	Ext. Features (Reduced)	10	0.0005	128	2
WiSig Real-world Data	Raw IQ (256)	10	0.0002	32	1
	Ext. Features (Full)	100	0.0001	128	1.5
	Ext. Features (Reduced)	30	0.0005	64	2

In all datasets, raw IQ samples are used as input to the CNN-based autoencoder, while extracted features serve as input to the autoencoder model. We evaluate both the full 24-dimensional feature set and the top 12 features selected via feature importance. The top features vary across different datasets. All testing results are analyzed based on precision, accuracy, and F1-score that are summarized in Table V.

The results indicate a consistent trend where extracted features outperform raw IQ data across all datasets. For Signal-level Synthetic Data, training on reduced features achieved 94.64% accuracy and an F1-score of 0.9463, higher than the 85.32% accuracy and 0.8712 F1-score using raw IQ data, showing that feature extraction captures critical device characteristics. In Protocol-level Emulated Data, raw IQ data failed completely, achieving 0% accuracy and an F1-score of 0, whereas reduced features enabled robust detection. For WiSig Real-world Data, raw IQ data yielded 56.52% accuracy and 0.5114 F1-score, while reduced features improved performance to 92.51% accuracy and 0.9286 F1-score, confirming their effectiveness in real-world conditions. These improvements arise because raw IQ data contains noise and redundant information, while extracted features highlight key patterns.

Comparing full and reduced feature sets, the reduced set generally performs better. In Signal-level Synthetic Data, legal accuracy rose from 83.65% to 94.64% and F1-score from 0.9079 to 0.9436, while impersonation detection remained high. Protocol-level Emulated Data showed slight improvements, with F1-score increasing from 0.9513 to 0.9578. In WiSig Real-world Data, legal accuracy improved from 78.8% to 87.7%, total accuracy from 87.85% to 92.51%, and F1-score from 0.8887 to 0.9286. Across all datasets, reduced features allowed impersonation detection accuracy to reach 97.36%, demonstrating the method's robustness.

When using raw IQ data alone, Signal-level Synthetic and WiSig datasets achieve reasonable results, but Protocol-level Emulated Data performs poorly due to signal complexity. This highlights the difficulty of extracting fingerprints from complex raw signals and the advantage of using extracted features.

D. *Limitations and Discussion*

While our study demonstrates the feasibility of unsupervised anomaly detection based on device-specific hardware impairments, several limitations remain. First, the stability of hardware impairments over long time scales is not considered, as our experiments focus on short-term scenarios. Second, handling new legitimate devices currently requires retraining, and exploring incremental training is an important future direction. Finally, the small size of our datasets may lead to overly optimistic performance metrics, with precision and recall approaching 100%. Collecting large-scale datasets remains challenging, but our current setup allows careful analysis of hardware-specific features and demonstrates the feasibility of the approach.

V. CONCLUSION

To address the limitations of supervised learning, we design an unsupervised anomaly detection method that uses extracted features from Wi-Fi signals and an autoencoder to identify impersonation attacks without requiring labeled data. Before implementing the model, we analyzed the importance of RF features under different hardware impairment settings. The analysis showed that impairments such as IQ imbalance and spectral features play a crucial role in distinguishing devices,

TABLE V
PERFORMANCE COMPARISON OF CENTRALIZED LEARNING

Dataset	Training Data	Legal Acc.	Impers. Acc.	Total Acc.	Precision	Recall	F1-score
Signal-level Synthetic Data	Raw IQ (80)	0.7178	0.9887	0.8532	0.7795	0.9887	0.8712
	Ext. Features (Full)	0.8365	0.9670	0.9018	0.8589	0.9670	0.9079
	Ext. Features (Reduced)	0.9452	0.9476	0.9464	0.9489	0.9476	**0.9463**
Protocol-level Emulated Data	Raw IQ (3840)	0.75	0.0	0.375	0.0	0.0	0.0
	Ext. Features (Full)	0.9067	0.9917	0.9492	0.9142	0.9917	0.9513
	Ext. Features (Reduced)	0.9117	1.0	0.9558	0.919	1.0	**0.9578**
WiSig Real-world Data	Raw IQ (256)	0.675	0.4555	0.5652	0.5834	0.4555	0.5114
	Ext. Features (Full)	0.788	0.969	0.8785	0.8217	0.969	0.8887
	Ext. Features (Reduced)	0.8770	0.9732	0.9251	0.8883	0.9732	**0.9286**

guiding the anomaly detection model to achieve better performance. We then implemented a centralized autoencoder-based anomaly detection model, trained only on legitimate device data. Across all datasets, the model successfully identified impersonation attacks, achieving an average impersonation detection accuracy of 97% and an F1-score of 0.93 using a reduced feature set. These results confirm both the effectiveness of our feature extraction process and the superiority of the proposed unsupervised approach for detecting impersonation attacks compared to using raw IQ data.

REFERENCES

[1] N. Dalal, N. Akhtar, A. Gupta, N. Karamchandani, G. S. Kasbekar, and J. Parekh, "A wireless intrusion detection system for 802.11 wpa3 networks," in *2022 14th International Conference on COMmunication Systems & NETworkS (COMSNETS)*, 2022, pp. 384–392.

[2] R. Banakh, A. Piskozub, and I. Opirskyy, "Detection of mac spoofing attacks in ieee 802.11 networks using signal strength from attackers' devices," in *Advances in Computer Science for Engineering and Education*, Z. Hu, S. Petoukhov, I. Dychka, and M. He, Eds. Cham: Springer International Publishing, 2019, pp. 468–477.

[3] Y. Sheng, K. Tan, G. Chen, D. Kotz, and A. Campbell, "Detecting 802.11 mac layer spoofing using received signal strength," in *IEEE INFOCOM 2008 - The 27th Conference on Computer Communications*, 2008, pp. 1768–1776.

[4] J. Zhang, G. Shen, W. Saad, and K. Chowdhury, "Radio frequency fingerprint identification for device authentication in the internet of things," *IEEE Communications Magazine*, vol. 61, no. 10, pp. 110–115, 2023.

[5] S. Hanna, S. Karunaratne, and D. Cabric, "Wisig: A large-scale wifi signal dataset for receiver and channel agnostic rf fingerprinting," *IEEE Access*, vol. 10, pp. 22 808–22 818, 2022.

[6] K. Sankhe, M. Belgiovine, F. Zhou, L. Angioloni, F. Restuccia, S. D'Oro, T. Melodia, S. Ioannidis, and K. Chowdhury, "No radio left behind: Radio fingerprinting through deep learning of physical-layer hardware impairments," *IEEE Transactions on Cognitive Communications and Networking*, vol. 6, no. 1, pp. 165–178, 2020.

[7] J. Zhang, R. Woods, M. Sandell, M. Valkama, A. Marshall, and J. Cavallaro, "Radio frequency fingerprint identification for narrowband systems, modelling and classification," *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 3974–3987, 2021.

[8] A. Elmaghbub and B. Hamdaoui, "Distinguishable iq feature representation for domain-adaptation learning of wifi device fingerprints," *IEEE Transactions on Machine Learning in Communications and Networking*, vol. 2, pp. 1404–1423, 2024.

[9] J. A. Gutierrez del Arroyo, B. J. Borghetti, and M. A. Temple, "Fingerprint extraction through distortion reconstruction (fedr): A cnn-based approach to rf fingerprinting," *IEEE Transactions on Information Forensics and Security*, vol. 19, pp. 9258–9269, 2024.

[10] G. Shen, J. Zhang, X. Wang, and S. Mao, "Federated radio frequency fingerprint identification powered by unsupervised contrastive learning," *IEEE Transactions on Information Forensics and Security*, vol. 19, pp. 9204–9215, 2024.

[11] J. Chen, W.-K. Wong, and B. Hamdaoui, "Unsupervised contrastive learning for robust rf device fingerprinting under time-domain shift," in *ICC 2024 - IEEE International Conference on Communications*, 2024, pp. 3567–3572.

[12] T. Zhao, S. Sarkar, E. Krijestorac, and D. Cabric, "Gan-rxa: A practical scalable solution to receiver-agnostic transmitter fingerprinting," *IEEE Transactions on Cognitive Communications and Networking*, vol. 10, no. 2, pp. 403–416, 2024.

[13] P. Lavoie, D. Naboulsi, and F. Gagnon, "Clustering of radio emitter characteristics with complex-valued cnns," in *2024 IEEE Canadian Conference on Electrical and Computer Engineering (CCECE)*, 2024, pp. 283–288.

[14] S. Tan, Y. Ren, J. Yang, and Y. Chen, "Commodity wifi sensing in ten years: Status, challenges, and opportunities," *IEEE Internet of Things Journal*, vol. 9, no. 18, pp. 17 832–17 843, 2022.

[15] B. Alotaibi and K. Elleithy, "A new mac address spoofing detection technique based on random forests," *Sensors*, vol. 16, no. 3, 2016. [Online]. Available: https://www.mdpi.com/1424-8220/16/3/281

[16] Z.-H. Zhou and X.-Y. Liu, "Training cost-sensitive neural networks with methods addressing the class imbalance problem," *IEEE Transactions on Knowledge and Data Engineering*, vol. 18, no. 1, pp. 63–77, 2006.

[17] J. An and S. Cho, "Variational autoencoder based anomaly detection using reconstruction probability," in *Special Lecture on IE*, vol. 2, no. 1, 2015, pp. 1–18.

[18] H. Torabi *et al.*, "Practical autoencoder based anomaly detection by using vector reconstruction error," *Cybersecurity*, vol. 5, no. 1, pp. 1–15, 2022.

[19] J. U. Ko, K. Na, J.-S. Oh, J. Kim, and B. D. Youn, "A new auto-encoder-based dynamic threshold to reduce false alarm rate for anomaly detection of steam turbines," *Expert Systems with Applications*, vol. 189, p. 116094, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0957417421014287

[20] X. Xu, H. Zhao, H. Liu, and H. Sun, "Lstm-gan-xgboost based anomaly detection algorithm for time series data," in *2020 11th International Conference on Prognostics and System Health Management (PHM-2020 Jinan)*, 2020, pp. 334–339.

[21] H. Jiang, Z. He, G. Ye, and H. Zhang, "Network intrusion detection based on pso-xgboost model," *IEEE Access*, vol. 8, pp. 58 392–58 401, 2020.

[22] A. Husain, A. Salem, C. Jim, and G. Dimitoglou, "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)*, 2019, pp. 1–7.

[23] M. Xie, J. Hu, S. Han, and H.-H. Chen, "Scalable hypergrid k-nn-based online anomaly detection in wireless sensor networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 8, pp. 1661–1670, 2013.

[24] R. Zhu, X. Ji, D. Yu, Z. Tan, L. Zhao, J. Li, and X. Xia, "Knn-based approximate outlier detection algorithm over iot streaming data," *IEEE Access*, vol. 8, pp. 42 749–42 759, 2020.

[25] L. Li, K. Jamieson, G. DeSalvo, A. Rostamizadeh, and A. Talwalkar, "Hyperband: A novel bandit-based approach to hyperparameter optimization," *Journal of Machine Learning Research*, vol. 18, no. 185, pp. 1–52, 2018. [Online]. Available: http://jmlr.org/papers/v18/16-558.html