Building Trust in IQ-Based RF Deep Learning Models for Wireless Applications

Sage Trudeau
University of Texas at Austin
Austin, TX, USA
MIT Lincoln Laboratory
Boston, MA, USA
strudeau@utexas.edu

Kaushik Chowdhury
University of Texas at Austin
Austin, TX, USA
kaushik@utexas.edu

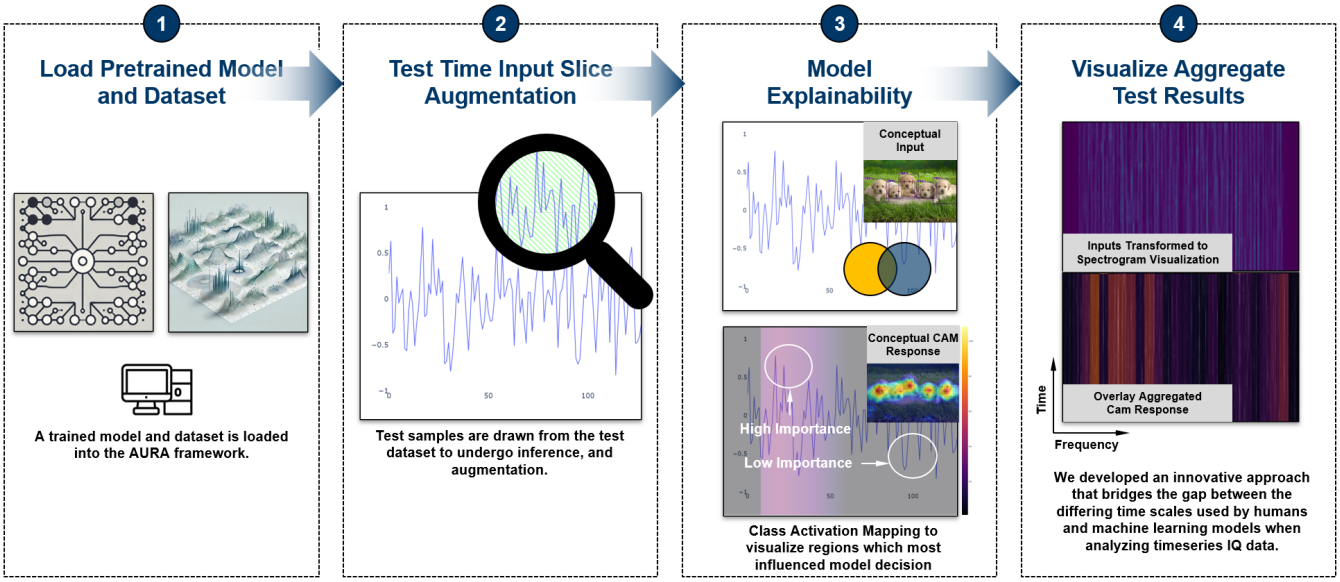

Figure 1: AURA framework helps to unveil what an RF ML model has learned. A model under evaluation is loaded, run through automated augmented input testing, and then quantified and visualized. This framework helps to determine which aspects of the wireless channel are accounted for, and which are not by the models learned features.

Abstract

The dynamic nature of wireless environments presents significant challenges for machine learning (ML) models in real-world radio frequency applications, where impairments such as noise, fading, and frequency shifts disrupt performance. To address these challenges and build trust in ML models, we present Augmented Input Resilience Analysis (AURA), a test framework designed for IQ-based RF models to rigorously assess ML model performance by simulating RF impairments and identifying critical vulnerabilities. AURA systematically applies test-time augmentations to provide a detailed examination of model strengths and weaknesses. Key contributions include (1) Score-CAM adapted to 1D IQ in time and a frequency-selective variant to localize spectral features, and (2) embedding similarity evaluation to quantify distribution shifts caused by impairments. By integrating these methods, AURA enhances interpretability, promoting trust in ML decision-making. We demonstrate AURA's utility in exposing critical vulnerabilities in well-cited models, such as over-reliance on power-based features, including instances where random noise is misclassified as a legitimate signal with 99.7% accuracy. AURA also evaluates remediation strategies, such as noise classes, which reduce misclassifications to less than 1% in the noise augmentation case. This framework aims to advance the design of trustworthy and resilient AI-driven systems for future RF ML technologies.

MobiHoc '25, Houston, TX, USA

ACM ISBN 979-8-4007-1353-8/25/10
https://doi.org/10.1145/3704413.3764466

CCS Concepts

• **Computing methodologies** → **Neural networks**; • **Hardware** → **Signal processing systems**.

Keywords

explainable artificial intelligence, data augmentation, radio frequency fingerprinting, waveform classification

ACM Reference Format:
Sage Trudeau and Kaushik Chowdhury. 2025. Building Trust in IQ-Based RF Deep Learning Models for Wireless Applications. In *Proceedings of The 26th International Symposium on Theory, Algorithmic Foundations, and Protocol Design for Mobile Networks and Mobile Computing (MOBIHOC'25).* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3704413.3764466

1 Introduction

Emerging wireless standards such as Wi-Fi 7 and 3GPP Release 18 increasingly consider machine learning (ML) for enhanced interference handling, spectrum allocation, and security measures in unlicensed and shared-spectrum scenarios [1, 15]. While these ML-driven approaches promise major gains for next-generation mobile and Wi-Fi networks, their real-world impact hinges on effectively handling the intrinsic variability of radio frequency (RF) environments. Indeed, channel conditions such as multipath fading, noise, and interference often cause distribution shifts that break models trained only under ideal or controlled conditions.

Two practical use cases illustrate these challenges. First, waveform classification in dynamic spectrum sharing settings must detect incumbent vs unlicensed despite unseen wireless channel scenarios [10]. Second, RF fingerprinting for device authentication risks overfitting to superficial traits such as device transmit power that do not generalize to realistic deployments [12]. Both issues are frequently uncovered *after* lab-stage trials, signaling the need for a framework that can systematically stress-test ML models throughout their development cycle in order to guide more robust designs.

To address this, we introduce Augmented Input Resilience Analysis (AURA), a comprehensive framework tailored for RF ML experimental studies on mobile and next-generation wireless networks. AURA provides an automated, open-source test harness that simulates real-world signal impairments via test-time augmentations, pinpoints model fragilities through embedding-space similarity, and explains learned signal features via time- and frequency-domain class activation mapping (CAM). By offering a cohesive suite of diagnostic methods, AURA not only reveals vulnerabilities like over-reliance on power-based cues but also proposes concrete remediation strategies to align ML performance with the harsh conditions of operational networks. Through this approach, researchers can build more reliable, trustworthy RF ML solutions that are ready for the uncertainties of actual wireless environments.

Our main **systems contributions** and research use cases enabled by AURA are as follows:

- **AURA: an open test harness for IQ-based RF models**: ([18]) Given a pretrained model and test set, AURA applies stochastic *test-time* augmentations that emulate OTA impairments and produces quantitative and visual diagnostics.
- **Class Activation Mapping in Time and Frequency Domains**: We adapt Score-CAM to (i) time-domain IQ and (ii) a new frequency-selective variant to localize spectral features driving decisions with open source code release ([18]).
- **Alignment of Model Outputs with Signal-Processing Intuition**: By aggregating sequential samples and aligning them directly with standard spectrograms or time-domain plots, AURA bridges ML inferences with domain expertise, expediting the debug process and trust-building.
- **Analysis of Distribution Shifts via Embedding Similarity**: We quantify how channel impairments shift model embeddings, isolating both benign shifts from desirable learned hardware features and harmful ones which cause misclassifications.

All experiments and methods in this paper target IQ-based models that consume raw 1D time-domain IQ samples. We do not convert inputs to spectrograms or image-like tensors for training, and our findings therefore pertain to architectures operating directly on IQ sequences. We do utilize spectrograms as a visualization tool to assist with signal processing intuition such as in Figure 9. In all cases, the results are calculated in the time domain and then transformed to spectrogram representation afterwards to enhance interpretation. Future efforts aim to include models additionally trained on hand-crafted features or 2D spectrogram inputs.

2 Related Work

2.1 Robustness, Data Augmentation in RF ML

Modern ML architectures can achieve high accuracy on idealized datasets but often struggle with real-world RF impairments [8]. To address distribution shifts caused by noise, fading, or frequency offsets, researchers have explored various domain adaptation strategies such as aligning learned features across frequency bands and data augmentation methods that simulate realistic channel conditions [21]. Although these techniques can mitigate overfitting, few frameworks systematically pinpoint which impairments degrade model performance the most and how best to remedy them—gaps that motivate our proposed approach.

2.2 Interpretability for RF Applications

As ML-based systems become integral to wireless communications, there is growing interest in understanding how and why models arrive at their decisions. Duggal *et al.* [3] introduce saliency maps for modulation classification, revealing time-frequency regions that strongly influence the network's output. Other methods, such as SHAP [9], quantify the contribution of individual input features to a model's predictions, whereas Grad-CAM [13] ranks the importance of spatial regions in 2D input data. While saliency methods are mature for 2D images, many RF models operate on 1D IQ sequences, limiting direct reuse. To bring CAM techniques to IQ-based RF models, we adapt Score-CAM [20] to 1D inputs, and we introduce a frequency-selective variant. These yield intuitive *time- and frequency-localized* visualizations that help RF experts relate model focus to signal-processing expectations and spot failure conditions.

3 AURA: An Overview

AURA evaluates the ML model's resilience by subjecting it to a comprehensive series of augmented input tests, using the original test dataset, and then visualizing the model's performance under this variety of scenarios. In the following subsections, we will detail the steps to our process flow outlined in Figure 1.

3.1 Load Pretrained Model and Dataset

Firstly, a trained model is loaded into the AURA framework to undergo evaluation by including both model parameters along with its original test dataset. A helper function is provided with examples in our Github repository to load new models swiftly.

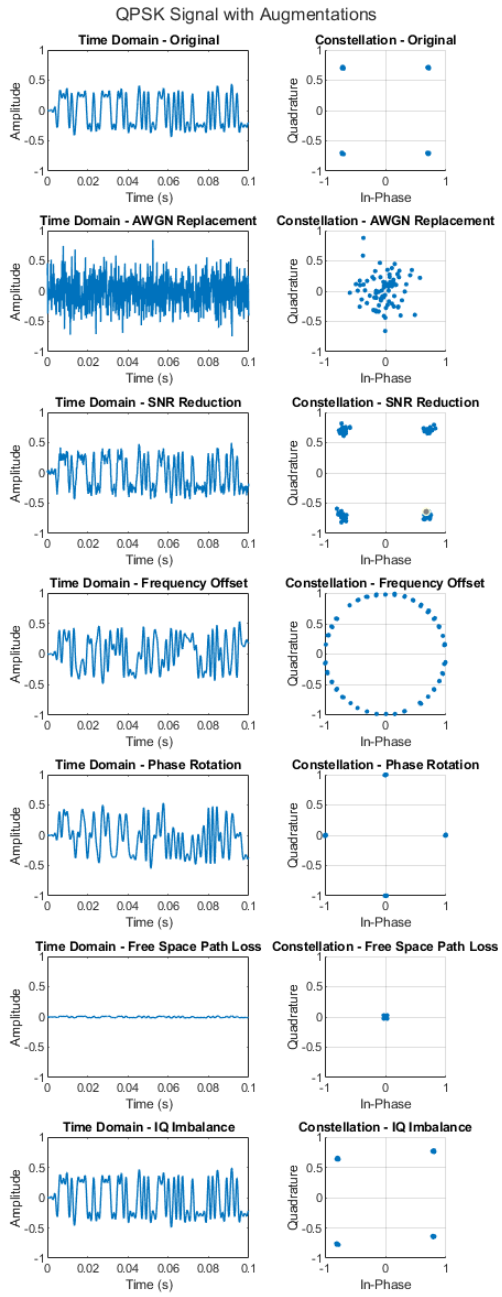

Figure 2: Augmentations used in AURA: SNR reduction, CFO, phase, FSPL, IQ imbalance; values shown are illustrative, configuration ranges can be seen in Section 5.1

3.2 Test Time Input Slice Augmentation

Secondly, test samples are drawn from the test dataset to undergo inference and augmentation. The original sample is run to calculate ground truth, and then N augmented versions of that sample are compared according to the experiment test plan parameters. These augmentations encompass a spectrum of real-world RF challenges, including Additive White Gaussian Noise (AWGN) replacement, SNR reduction, frequency shifts, phase flips, phase rotations, free space path loss, and IQ imbalance. A visualization of these augmentations can be seen in Figure 2. The model's original test dataset is used with the addition of these augmentations to determine which aspects of the wireless channel that model under test struggles on. Additionally, AURA incorporates a flexible augmentation function, allowing for user-defined modifications or extensions via MATLAB's signal processing toolkit [17]. The details of the augmentations can be found in the example walkthrough.

3.3 Model Explainability

The aura framework utilizes two primary methodologies to help explain the model's decisions under evaluation, highlighting which distortions are strengths and weaknesses for that model's architecture. The first compares broadly the embedding similarity between the original and augmented test case through qualitative and quantitative metrics standard in literature. We compare the feature space embedding of the original and augmented inputs across multiple iterations to examine stability in the output label. The intuition here is whether it remains unchanged, becomes randomly confused, or systematically shifts to another class in its predictions. The second explainability method aims to visually highlight for a particular erratic classification which aspects of the signal caused the model to misclassify. Here, we adapt a technique from image processing, known as class activation mapping [22], to the RF domain.

3.3.1 Embedding Similarity. In our evaluation framework, we leverage embedding similarity scores to assess the robustness of RF machine learning models under various augmented input conditions. By examining the feature space outputs at the layer immediately before the classification layer, we can gauge the impact of input distortions on the model's internal representations. The combined use of Affinity, Distance Metrics, and t-distributed Stochastic Neighbor Embedding (tSNE) Visualization [19] offers a broad high-level view of how data augmentation affects model performance and robustness. As introduced by Gontijo-Lopes et al.[5], Affinity measures the shift in data distribution relative to the model's decision boundary, quantifying how an augmentation alters the data perceived by the model. It is defined as the difference in model accuracy on clean data versus augmented data, reflecting the model's sensitivity to augmentation-induced changes. This metric helps us understand the degree to which the augmented data remains within the model's effective classification boundary.

Analysis of these shifts due to particular augmentations and their implications can be found in Section 6.2.

3.3.2 Class Activation Mapping. Building on Score-CAM [20], we adapt class activation mapping to RF models that operate on 1D raw IQ and introduce a frequency-selective variant. Figure 3 illustrates the concept on images: regions most influential for the

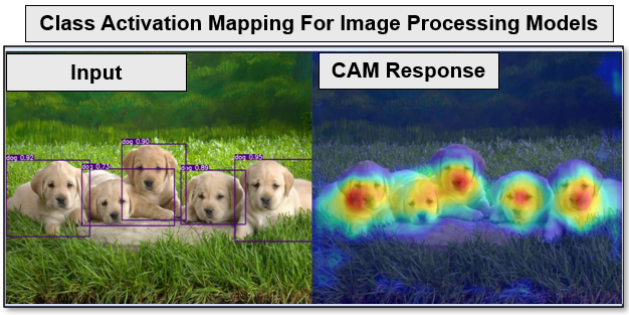

Figure 3: Class Activation mapping visualizes original input (left), and overlayed model perspective (right). This technique highlights which pixels in the input the model most highly values towards the output of "dog".

predicted class ("dog") are highlighted; spurious highlights such as background grass would indicate shortcut features. In AURA, time- and frequency-domain maps localize the temporal/spectral indices that drive each decision, revealing how signal distortions affect model focus. Implementation details and modifications appear in Section 5.2.

3.4 Interactive GUI for Model Analysis

To visualize the test results of our ML classifier for radio frequency signals, we developed an innovative approach that bridges the gap between the differing time scales used by humans and machine learning models when analyzing time-series IQ data. While our models operate on tiny slices of perhaps hundreds of samples—representing mere fractions of a second in the fast-paced RF domain—humans require more extensive context to interpret signals effectively. Traditional signal processing visualizations like spectrograms, time-domain plots, power spectral density (PSD), and constellation diagrams provide this broader context on a larger time scale, enabling researchers to orient themselves to the type of modulation or communication in the signal.

Our tool allows users to view signals through these traditional means and then zoom into specific sections of interest. This zoomed-in view remains synchronized to the model response and XAI techniques to reveal how the classifier responds to those segments and their augmented versions. Researchers can observe which aspects of the signal the model focuses on in both the time and frequency domains, gaining insights into the classifier's opinions, difficulties, or indifferences regarding particular signal slices.

By providing this contextual layering, our visualization approach enables researchers to align their signal-processing intuition directly alongside the model's decision-making process. This offers a new perspective for testing hypotheses and enhances explainability, building trust in our models and fostering a deeper understanding of their inner workings.

4 End to End Example Walkthrough

To illustrate AURA's practical application, we provide a detailed walkthrough using the ORACLE model—a one-dimensional convolutional neural network (1D CNN) designed for RF fingerprinting. This example aims to demonstrate each step of our proposed approach, highlighting how AURA can be employed to evaluate and interpret a pretrained RF machine learning model. By systematically applying augmented input tests and explainability techniques via AURA's embedding similarity analysis and CAM visualizations, we uncovered that the ORACLE model appeared to over-rely on power-related features. This walkthrough shows how identifying such insights can improve model robustness and guide future developments in the field. This example is illustrative rather than a universal benchmark, and it is scoped to IQ-based models under test-time augmentations without retraining to help researchers identify what features their models may have learned.

4.1 Example Model Under Evaluation

We evaluate the journal version of ORACLE [12], a 1D CNN for RF fingerprinting on raw IQ. ORACLE processes I/Q as two channels with stacked conv blocks and a softmax head; we use the authors' training setup and focus solely on *test-time* stress tests without retraining.

4.2 Augmented Input Testing

During the input slice augmentation phase, we applied a comprehensive set of signal distortions to the test dataset to evaluate the model's resilience under various real-world RF conditions. All augmentations are applied at *test time*; the model is not retrained in these experiments. All augmentation tests were enabled, including:

- **Additive White Gaussian Noise (AWGN) Replacement**: Noise was added with a relative power of 0dB relative to the average original signal power, simulating false inputs which matched the average power of the training data.
- **Carrier Frequency Offset (CFO)**: Random frequency shifts were introduced, uniformly selected between −500Hz and 500Hz, to mimic small frequency offsets between Tx and Rx.
- **Signal-to-Noise Ratio (SNR) Reduction**: The SNR was reduced by a range of 10 to 50 dB reduction.
- **Phase Rotation**: Phase angles were uniformly selected from -180° to 180° to simulate phase shifts caused by channel effects.
- **Free Space Path Loss (FSPL)**: Attenuation was applied based on distances randomly selected between 0 m and 10 m, using a path loss exponent (α) of 2, corresponding to open-air conditions.
- **IQ Imbalance**: A gain imbalance factor of 1.02 and a phase imbalance of 3° were introduced to simulate hardware imperfections commonly found in SDRs.

The dataset is sourced from bit-similar USRP X310 radios that transmit frames compliant with the IEEE 802.11a standard, which are generated using the MATLAB WLAN System toolbox and then have been transmitted and received over the air. These frames have random payloads while maintaining consistent address fields. Dataset details are provided with the model by the original authors [4]. The test dataset chosen for AURA utilized the first six devices with 1000 instances per class randomly selected from across all distances. Signal slices of 128-length samples were extracted randomly from the original test dataset without overlap. These augmentations were applied to the slices to create augmented test inputs, which were

then used to evaluate the pretrained model's performance under these varied conditions.

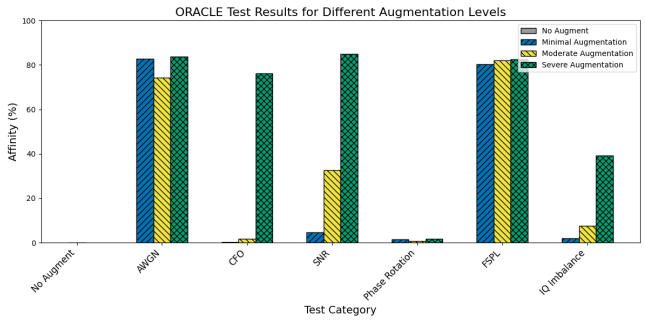

Figure 4: Affinity of ORACLE model under each AURA test category. The original model accuracy was 99.7%. The model is especially resilient to frequency offsets, phase rotations, and IQ Imbalance, while it struggles with AWGN noise replacement, severe SNR reduction, and FSPL.

4.3 AURA Test Results

Figure 4 summarizes the ORACLE model's reactions to different data augmentations. Overall, the model performs admirably, considering the variety of distribution shifts caused by each data augmentation. The model is incredibly resilient to frequency offsets, phase rotations, and IQ Imbalance, while it struggles with AWGN noise replacement, severe SNR reduction, and FSPL channel distortion. It is significant that the model can handle small and moderate frequency offsets and is indifferent to any phase rotation, as these are both common from changes in the relative positions between the transmitter and receiver. SNR behaves as expected, with model accuracy decreasing as noise levels increase. Finally, IQ Imbalance is also seen as an exciting result as it supports the original author's hypothesis that IQ imbalance is a feature learned by the model to distinguish between identical hardware. For slight imbalances, the model handles with ease, and then once those distortions become severe, the radios lose some of their distinguishing hardware characteristics, and the model, as expected, begins to struggle.

It is observed that the commonality between the challenging cases is signal power. The model reacts strongly to augmentations that distort the signal level of the input slice. To investigate that further, the next section utilizes the CAM technique to determine the features the ORACLE model has learned.

4.4 Investigate Learned Model Features

We utilize the CAM technique to investigate this apparent reliance on power-related features, with a targeted test that matches AWGN power to a specific class to probe for shortcut learning. Here, we look at a slice in time that is made up of noise with no transmission, and then an 802.11a packet begins. We overlay the CAM response onto the signal input to verify that our results match our intuition – that in areas of noise, there should be no response, and in areas of signal, there should be a highlight in the CAM. This can be seen in Figure 5.

We begin with the simplest case representing the most common scenario in the real world where the noise power is substantially lower than the signal power of the device we wish to use to detect. We use device three here of our six devices arbitrarily. In this first scenario, the model is seen to have low accuracy in the region of AWGN and high accuracy in the region that has not been replaced. The cam response also matches this outcome in that it shows activations in the region, which led to a classification of device 3, and no activation in regions replaced by the low-power AWGN, which correctly does not detect device 3. With results looking good, we move on to a more challenging case.

In our augmentation testing, we noticed that the accuracy was slightly higher in the AWGN moderate case when the power of the noise matched the power of the signal, so we visualize that case in Figure 5 (c) and (d) to see what is different in terms of the activation of feature maps via the CAM. In this case, we see that the model ultimately still has low accuracy, which is good. Still, some internal features are activating without ultimately leading to a classification output. This on its own is not necessarily an issue as long as the model ultimately does not misclassify, but it warrants curiosity. From this response, it was hypothesized that the model had learned features related to the specific power levels of the signal. To test this, a third scenario was run in which rather than just matching the average power of the signal for the noise replacement, we matched the specific power levels of device 0 in an attempt to fool the classifier. In doing so, we uncovered that when a signal is replaced with a AWGN slice matching a class power (class zero in this case), the classifier will misclassify that noise as that target class. Results can be seen in Figure 5 (e). We further verify this behavior over 100 different randomly selected test inputs in which we change the signal power of the original slice to match a random input of class 0 power. This results in the original class to be misclassified as class 0. Results can be seen in Figure 6.

This is a concerning condition as we can conclude that by matching the power of a particular device in the training dataset and creating a random input with that power, we can cause our classifier to misclassify. The features learned by this model are too dependent on power and are likely to experience issues in real-world scenarios when power is subject to easily change due to distance or other sources of attenuation.

4.5 Mitigating AWGN Misclassification with a Null Class

To address the misclassification behavior observed in the ORACLE model under AWGN augmentation, we added a seventh null class during training. This class, labeled "noise," consists of pure noise inputs with varying power levels between the signal SNR and a -10 dB reference. The null class was introduced as an additional class during training, providing the model with examples of spectrum sections devoid of meaningful signals.

This approach proved highly effective in mitigating the AWGN misclassification issue. As shown in Figure 7, the confusion matrix for the mimic scenario—where inputs are replaced with AWGN matching the power of class 0—now demonstrates the model's ability to correctly classify these inputs as class 7 ("noise"). This can be seen both in the unaugmented case, and in the mimic case.

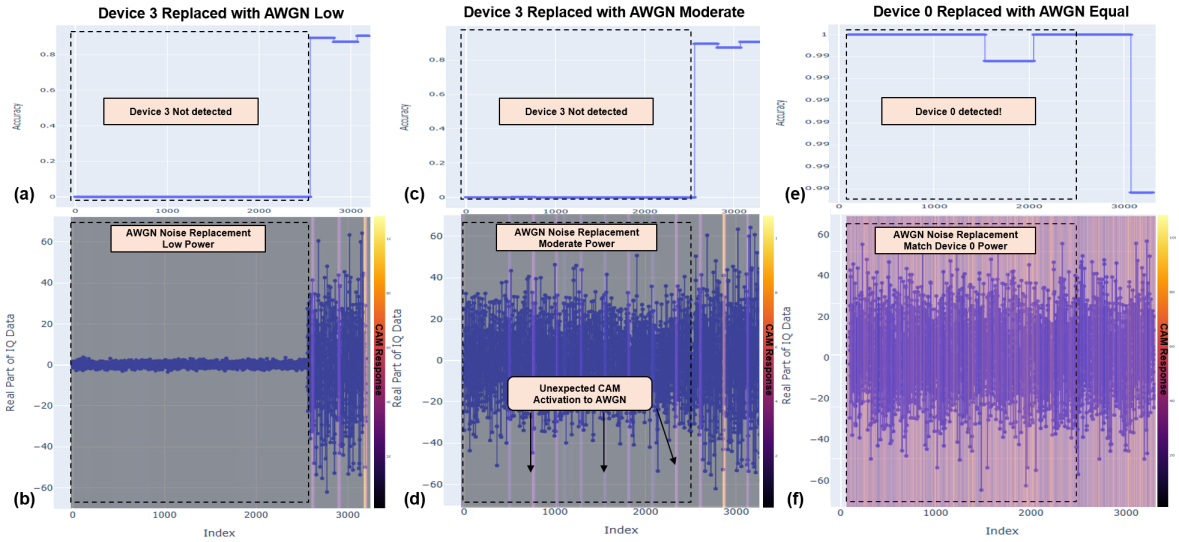

Figure 5: ORACLE time domain CAM under three AWGN replacement scenarios. Each scenario shows the model accuracy over the range of input slices (top row) and the model CAM response overlayed over the input data (bottom row).

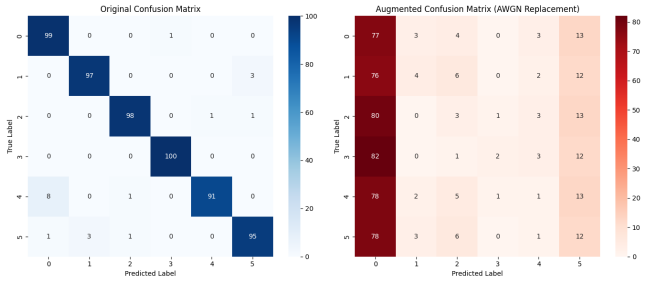

Figure 6: Confusion matrix of unaugmented data (left) and augmented data(right) in which the signal is replaced with AWGN matching class 0 power. The model can be seen to misclassify noise that matches the power level of one of the classes, suggesting power as the dominant feature learned.

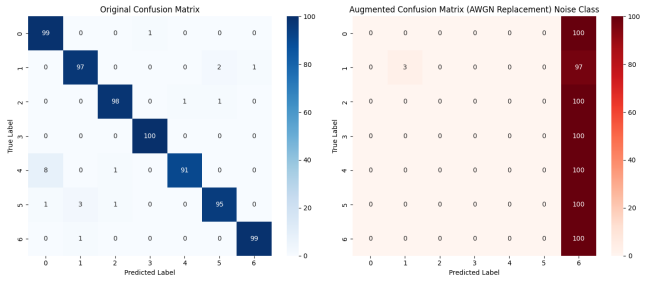

Figure 7: Mitigation with a noise class (6). Classes 0–5 as in Fig. 6; class 6 = noise. Power-matched AWGN is now correctly classified as noise rather than as class 0.

Unlike the original model behavior, where all augmented inputs were misclassified as class 0 if they matched class 0 power, the updated model consistently recognizes pure noise, regardless of power level.

Adding the null class enhances the model's ability to differentiate real signal inputs from noise, a crucial capability for deployment in real-world RF environments. By training the model to recognize noise as a distinct class, we effectively address the AWGN-induced distribution shift, improving generalization and robustness.

5 Technical Implementation

5.1 Input Augmentations

This section details the various signal augmentations implemented in the AURA framework. Each augmentation tests the pretrained RF machine learning model under different challenging conditions. The inputs to the models are assumed to be two-channel time-domain IQ samples, with channel 0 representing the in-phase (I) component and channel 1 the quadrature (Q) component.

5.1.1 Additive White Gaussian Noise (AWGN) Replacement. In the AWGN replacement augmentation, we substitute the original signal with white Gaussian noise to simulate an extremely low-SNR scenario. Concretely, we set $x_{\text{awgn}}[n] = \mathcal{N}(0, \sigma^2)$, where $\mathcal{N}(0, \sigma^2)$ denotes a Gaussian distribution with mean 0 and variance σ^2.

5.1.2 SNR Reduction. SNR reduction adds noise to the original signal to decrease the effective SNR. Specifically, $x_{\text{snr}}[n] = x[n] + \mathcal{N}(0, \sigma^2_{\text{target}})$, where $\sigma^2_{\text{target}} = \frac{P_{\text{signal}}}{\text{SNR}_{\text{target}}}$. Here, P_{signal} is the power of the original signal, and $\text{SNR}_{\text{target}}$ is the desired signal-to-noise ratio.

5.1.3 Frequency Shift. Frequency shifting tests the model's ability to handle offsets between transmitter and receiver. We apply $x_{\text{freq}}[n] = x[n] \cdot e^{j2\pi f_{\text{shift}} n}$, where f_{shift} is the shift in Hz.

5.1.4 Phase Flip. Phase flip inverts the phase of the signal as $x_{\text{phase\_flip}}[n] = x[n] \cdot e^{j\pi}$. This challenges the model's reliance on phase-sensitive features.

5.1.5 Phase Rotation. Phase rotation applies a constant phase shift θ to the entire signal: $x_{\text{phase\_rot}}[n] = x[n] \cdot e^{j\theta}$. This tests the model's sensitivity to uniform phase variations.

5.1.6 Free Space Path Loss. To simulate path-loss attenuation due to distance d, we scale the signal by $x_{\text{pathloss}}[n] = x[n] \cdot \frac{1}{d^{\alpha}}$. Here, α is the path loss exponent. This simplified approach omits frequency dependence but preserves the spirit of signal attenuation.

5.1.7 IQ Imbalance. We set $x_{\text{IQ\_imbalance}}[n] = (I[n] + \epsilon I[n]) + j\,(Q[n] + \epsilon Q[n])$, where ϵ is the gain/phase imbalance factor. IQ imbalance accounts for hardware imperfections in the in-phase and quadrature components.

5.1.8 Open Augmentation Function. AURA also supports user-defined or MATLAB-based signal processing effects, represented generally as $x_{\text{open}}[n] = f(x[n], \text{params})$, where f is a function applying the augmentation, and params controls its behavior.

5.2 Class Activation Mapping

SCORE-CAM [20] is an explainable AI (XAI) technique originally developed for image processing. Unlike gradient-based methods such as Grad-CAM [13], SCORE-CAM probes multiple perturbed inputs and observes changes in class logits. The resulting activation map highlights which regions most strongly influence the model classification. We use Score-CAM because it (i) does not require gradients at the target layer making it more robust to saturating activations and vanishing gradients, (ii) treats the model largely as a black box beyond the chosen layer, and (iii) yields stable maps under small perturbations, which we found important for 1D IQ slices. Grad-CAM is lightweight but depends on gradient quality; SHAP offers feature attribution but is expensive for dense time series and requires careful background choices. We adapt SCORE-CAM from 2D images to both time and frequency domains in 1D RF signals.

5.2.1 Time Domain. The time domain implementation (Algorithm 1) creates masks derived from the activation profiles in a chosen model layer, then measures how much each masked input decreases the logit of the target class relative to a baseline. These steps closely follow SCORE-CAM's original formulation, but are tailored for one-dimensional IQ data.

5.2.2 Frequency Domain. In the frequency-based approach (Algorithm 2), we first transform the signal into the spectral domain, then mask out individual frequency bins via notch filters. After an inverse transform, we compare the class logit to a baseline to gauge each bin's significance. The remainder of the weighting and final CAM aggregation mirrors the time-domain version, but operates on spectral components rather than per-channel time slices.

Key Differences. Although both versions use softmax weighting and a final ReLU-based combination of activation maps, the frequency-domain method (Algorithm 2) applies notch filters to individual bins. By comparing the drop in logit (relative to X_b), we deduce each bin's relevance to the model's decision. In contrast, the time-domain method (Algorithm 1) creates channel-wise masks directly on upsampled activation profiles. Both approaches ultimately highlight the regions (time or frequency) that most affect the target class logit.

Algorithm 1: SCORE-CAM for 1D Time Series

Input: 1D time-series X_0, model $f(X)$ with chosen layer l, baseline X_b, target class c
Output: Score-CAM output $L_c^{\text{Score-CAM}}$
1: **Initialization:** Load X_0 into $f(\cdot)$.
2: **Activation of Layer l:** Obtain activation profiles A^l for X_0.
3: **Channel Masking:** Let C be the number of channels. For each channel k:
- Upsample A_k^l to match X_0 in time length.
- Normalize to form M_k^l.
- Compute the masked input $M_k^l \circ X_0$ and collect in set M.

4: **Batch Processing:** Pass all masked signals M through f in batches.
5: **Class Logit Calculation:** $S_c \leftarrow f_c(M) - f_c(X_b)$.
6: **Weight Calculation:** $\alpha_c^k \leftarrow \frac{\exp(S_c^k)}{\sum_k \exp(S_c^k)}$.
7: **Score-CAM Output:** $L_c^{\text{Score-CAM}} \leftarrow \text{ReLU}\big(\sum_k \alpha_c^k A_k^l\big)$.

Algorithm 2: Frequency-Domain Adaptation for CAM

Input: 1D time-series X_0 (length N), model $f(\cdot)$ with chosen layer l, baseline X_b, target class c
Output: Frequency-based CAM output L_c^{FreqCAM}
1: **Initialization:** Load X_0 into $f(\cdot)$.
2: **Frequency Transform:** Compute X_f using DTFT, $X_f[k] = \sum_{n=0}^{N-1} X_0[n]\, e^{-j\frac{2\pi}{N}kn}$.
3: **Generate Frequency Masks:** For each bin k, create a mask M_k that selectively zeroes or scales $X_f[k]$. Collect all masks in set M.
4: **Inverse Transform:** For each masked $X_f \circ M_k$, apply $X_{0,k}[n] = \frac{1}{N} \sum_{m=0}^{N-1} (X_f[m]\, M_k[m])\, e^{j\frac{2\pi}{N}mn}$.
5: **Class Logit Calculation:** $S_c \leftarrow \big| f_c(X_b) - f_c(X_{0,k}) \big|$.
6: **Weight Calculation:** $\alpha_c^k \leftarrow \frac{\exp(S_c^k)}{\sum_k \exp(S_c^k)}$.
7: **Construct FreqCAM:** $L_c^{\text{FreqCAM}} \leftarrow \text{ReLU}\big(\sum_k \alpha_c^k A_k^l\big)$.

6 Evaluation

We first interpret behaviors with CAM, then present quantitative effects of augmentations on accuracy and embedding-space geometry. Across ORACLE (RF fingerprinting), frequency/phase distortions have minor impact while power-affecting transforms (AWGN replacement, severe SNR reduction, path-loss scaling) dominate accuracy loss (Fig. 4). For T-Prime (waveform classification), decreasing SNR and large CFO translate clusters in embedding space before collapsing toward a noise-like region (Figs. 10–11), suggesting embedding translation-based mitigation could be promising at moderate impairment levels.

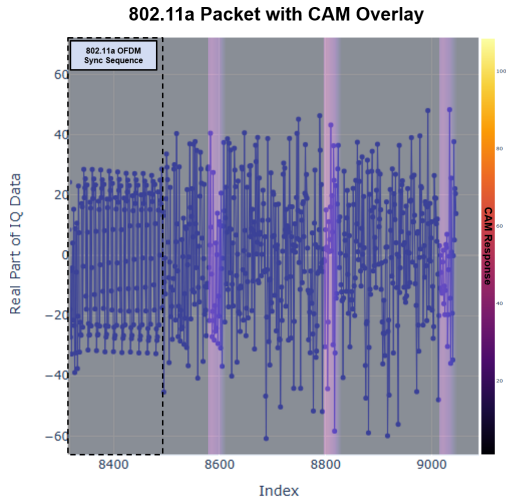

Figure 8: Aggregated Time domain CAM overlay of 802.11a sync sequence input seen by the ORACLE RF Fingerprinting Model. Sections of the packet which were maximally responsible for the classification are highlighted by the CAM.

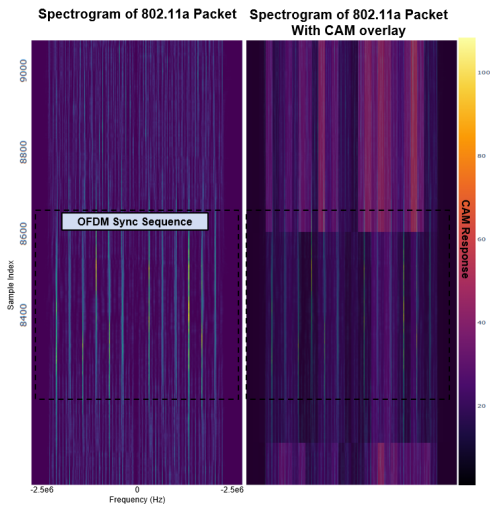

Figure 9: Spectrogram visualization of multiple sequential test input slices with no augmentation. Frequency CAM highlights regions of signal while ignoring regions of noise. The sync sequence highlighted is the same as Figure 8.

6.1 CAM Evaluation in Time and Frequency Domains

We employ CAM in both the time and frequency domains to examine which signal regions most strongly influence the model's classification decisions. This dual analysis also verifies the correctness of our CAM implementations by comparing known signal-processing expectations with the model's highlighted areas.

6.1.1 Time-Domain Analysis. A practical way to validate CAM in the time domain is to use known aspects of an 802.11a packet where we have strong intuition. Specifically, we target sequential slices leading up to the start of a packet, where we expect minimal CAM activation in noise and a clear activation once the true signal begins. As shown in Figure 5, the model correctly exhibits low activation in noise regions and higher activation where the signal is strongest, illustrating its effectiveness as a "negative hypothesis test" for unwanted features. We further investigate the commonly studied preamble section, also known as the sync sequence, which is identical across all transmitted packets. In principle, this region can reveal subtle device differences for RF fingerprinting [6, 14], yet our results (Figure 8) show that the ORACLE model places comparatively low emphasis on this lower-power preamble. This aligns with earlier observations that ORACLE over-relies on power-based cues. While other works have shown success with preamble-focused approaches [11], here it appears that preamble features do not strongly resonate with ORACLE's learned representation.

6.1.2 Frequency-Domain Analysis. We apply the same principle in the frequency domain by masking out specific frequency bins and observing changes in classification confidence. Figure 9 shows how the model highlights spectral regions corresponding to the packet's main signal while ignoring adjacent noise-only frequencies, indicating that it is not inadvertently latching onto out-of-band environment noise. Interestingly, like in the time-domain case, the model again shows only a moderate response to the sync sequence's subcarriers, suggesting it does not rely heavily on that portion of the waveform for final classification.

Taken together, these time-domain and frequency-domain CAM evaluations suggest that while the model responds correctly in broad signal versus noise discrimination, it prioritizes higher-power segments over subtle features like the preamble. This insight can be valuable for refining model architectures or training approaches to ensure future classifiers exploit more robust features than power-related cues.

6.2 Distribution Shifts in Augmented Data: A Case Study with T-Prime

In this subsection, we analyze the distribution shifts introduced by different augmentations and their impact on model behavior using embedding similarity metrics. To expand our evaluation, we introduce the T-Prime model, which is designed for over-the-air (OTA) Wi-Fi waveform classification [2]. Unlike the ORACLE model, which focuses on RF fingerprinting, T-Prime operates in a broader spectrum sensing context and has demonstrated real-world test cases. This makes this model better suited for evaluating channel-related augmentations such as signal-to-noise ratio (SNR) degradation and frequency offsets, as we can cross-validate our findings with their experiments. This shift in focus also highlights AURA's versatility in analyzing different model architectures and application domains.

To understand how different augmentations manifest as distribution shifts, we compare the embedding similarity between the original and augmented test data for three cases:

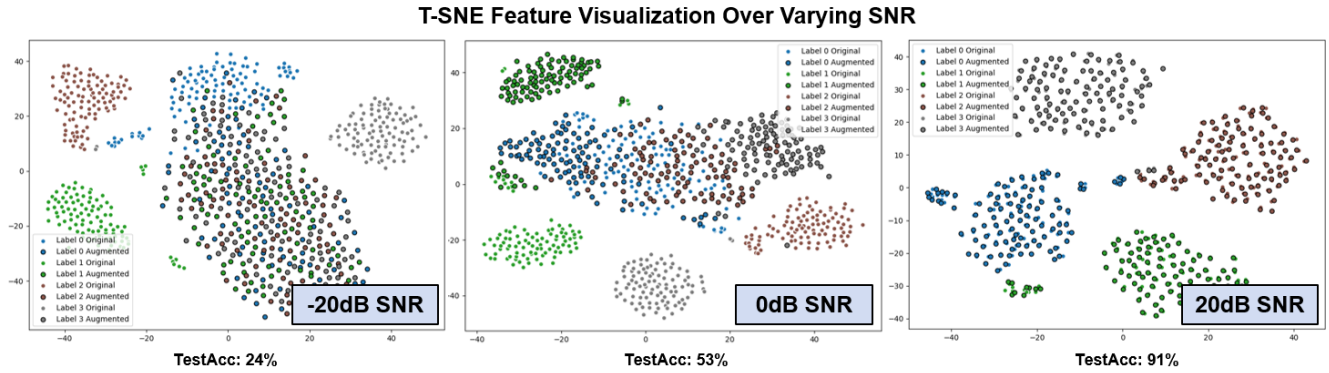

Figure 10: t-SNE plot of model embeddings as SNR decreases. At 20,dB, augmented data (black borders) remains near the original data (no borders). As SNR drops, embeddings diverge and ultimately converge into a noise-like cluster at −20,dB.

- **SNR Augmentation:** A control case where reduced SNR should cause increasing ambiguity in the decision boundaries. Designed to compare results directly to the original paper. Results in Figure 10
- **Frequency Offset Augmentation:** A scenario that simulates the practical challenge for Wi-Fi classification systems of monitoring broadly across all channels, where activity on an adjacent channel 5MHz away could be seen as a frequency offset from the center frequency, but ideally would still classify.
- **AWGN Augmentation:** The same scenario as presented to the ORACLE model to compare and contrast model behaviors.

Using t-SNE visualizations of the embedding space, we explore the degree and nature of the shifts induced by these augmentations. For each case, we analyze the behavior of the T-Prime model under varying augmentation levels to observe trends.

In Figure 10 we see the class clusters degrade as the SNR increases. This is the expected result, with very little change in the 20dB case and nearly complete separation in the -20dB scenario. Test accuracy is comparable to the original reported SNR tests in which they got 25%, 60% and 90%. For SNR, the loss of classification confidence is expected and points to a general limitation in model resilience under extreme conditions. The 0dB scenario, however, offers an interesting observation. In this scenario, we can see that the clusters have shifted but have largely maintained their respective shapes. The observed translation in the embedding space suggests that this augmentation can be mitigated during training with appropriate data augmentation strategies, improving robustness without fundamentally altering the learned feature space.

We can see additional support for this possibility in the Frequency augmentation case, seen in Figure 11. Here we see very similar embedding translation behavior in the extreme case, suggesting that both could be improved by data augmentation. Many works with substantial success have explored this [7, 16]. However, in the examples seen, the data augmentation strategies are used to improve the accuracy of the test dataset provided rather than compare the test accuracy to new real-world test scenarios unseen

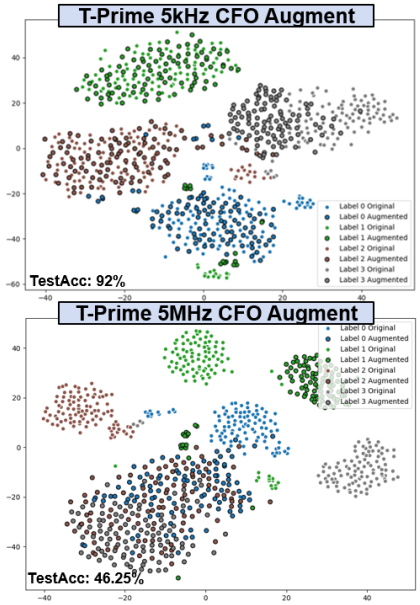

Figure 11: TSNE Plot of model embeddings under CFO augmentation.

during training. Future work aims to explore these techniques with a focus on real-world unseen scenarios.

The final scenario we investigate is the AWGN test case, which compares T-Prime to ORACLE. These two models have vastly different applications and so they cannot truly be directly compared, but for this general AWGN case, it is interesting to discuss the difference. In Section 4.4 we found that by mimicking the power of any of the devices, we could cause a purely random input to be classified as that device with high confidence. We perform that same experiment on T-Prime to see if the models exhibit similar behavior. Interestingly, what we found instead was that no matter which waveform we attempt to mimic in power, the embeddings of the augmented noise samples consistently end up in the decision region of class 0. We hypothesize the reason for this as class 0 is

the most similar class to noise. This is due to class 0 in the T-Prime case being the 802.11g Wi-Fi protocol with 1024 OFDM subcarriers. Without synchronization in the time domain, this number of carriers shares a strong resemblance to pure noise. This highlights an interesting difference in learned features between the models.

Key takeaways.

- **Power sensitivity dominates** failure modes in ORACLE. Adding a noise/null class mitigates spurious decisions on power-matched AWGN.
- **Embedding translations precede collapse** under SNR/CFO in T-Prime, indicating room for robustness via targeted training augmentations.
- **CAM complements metrics:** time/frequency maps confirm that high-power regions drive decisions, while alternative power regions such as preamble-only receive lower emphasis in ORACLE.

7 Conclusion and Future Work

This paper introduced AURA, a systematic framework for evaluating and improving the robustness of machine learning models in RF applications. Using test-time augmentations, embedding-space analysis, and class activation mapping implementations in both time and frequency domains, AURA uncovers critical vulnerabilities—such as over-reliance on power-based features—and clarifies how models behave under realistic channel impairments. By bridging deep learning architectures with signal-processing intuition, AURA provides an interpretable view of model decision-making and highlights concrete paths for remediation. Our future goal is to enhance AURA's capabilities by incorporating more realistic channel effects during training, introducing noise-specific classes to mitigate misclassifications under adverse SNR, exploring advanced normalization techniques that adapt to diverse impairments, and systematically evaluating different model architectures to identify those best suited for real-world wireless environments. Through these iterative improvements, AURA enables the design of ML-driven RF systems that maintain high performance under unpredictable, distribution-shifting conditions in practical deployments.

Acknowledgments

The authors wish to thank MIT Lincoln Laboratory Lincoln Scholars Committee, Dr. Joey Botero, and Dr. Alexia Schulz for support and guidance. This research is funded in part by the US National Science Foundation under awards 2434043, 2526493 and 2112417.

DISTRIBUTION STATEMENT A. Approved for public release. Distribution is unlimited. This material is based upon work supported under Air Force Contract No. FA8702-15-D-0001. Any opinions, findings, conclusions, or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the U.S. Air Force. © 2025 Massachusetts Institute of Technology. Delivered to the U.S. Government with Unlimited Rights, as defined in DFARS Part 252.227-7013 or 7014 (Feb 2014). Notwithstanding any copyright notice, U.S. Government rights in this work are defined by DFARS 252.227-7013 or DFARS 252.227-7014 as detailed above. Use of this work other than as specifically authorized by the U.S. Government may violate any copyrights that exist in this work.

References

[1] 3rd Generation Partnership Project (3GPP). [n. d.]. Release 18 Features. https://www.3gpp.org/images/PDF/Release_18_features_tsg95_v03.pdf.

[2] Mauro Belgiovine, Joshua Groen, Miquel Sirera, Chinenye Tassie, Ayberk Yarkin Yildiz, Sage Trudeau, Stratis Ioannidis, and Kaushik Chowdhury. 2024. T-PRIME: Transformer-based Protocol Identification for Machine-learning at the Edge. *arXiv preprint arXiv:2401.04837* (2024).

[3] Rohit Duggal et al. 2023. Dependable Modulation Classification Explainer (DMCE): Interpreting CNN decisions in wireless signal classification. *IEEE Transactions on Wireless Communications* 22, 6 (2023), 4044–4058.

[4] Genesys Lab. [n. d.]. ORACLE Dataset. https://www.genesys-lab.org/oracle. Accessed: 2024-11-18.

[5] Raphael Gontijo-Lopes, Sylvia J Smullin, Ekin D Cubuk, and Ethan Dyer. 2020. Affinity and diversity: Quantifying mechanisms of data augmentation. *arXiv preprint arXiv:2002.08973* (2020).

[6] Samer Hanna, Samurdhi Karunaratne, and Danijela Cabric. 2020. Open set wireless transmitter authorization: Deep learning approaches and dataset considerations. *IEEE Transactions on Cognitive Communications and Networking* 7, 1 (2020), 59–72.

[7] Liang Huang, Weijian Pan, You Zhang, LiPing Qian, Nan Gao, and Yuan Wu. 2019. Data Augmentation for Deep Learning-based Radio Modulation Classification. arXiv:1912.03026 [eess.SP] https://arxiv.org/abs/1912.03026

[8] Jane Kaplan et al. 2022. RF Domain Adaptation: Enhancing Deep Emitter Classification Performance Across Frequency Channels. *IEEE Transactions on Wireless Communications* 21, 9 (2022), 1234–1245.

[9] Peng Ma and Others. 2022. SHAP-based interpretability for device fingerprinting in dense IoT environments. *IEEE Internet of Things Journal* 9, 13 (2022), 10271–10284.

[10] Timothy James O'Shea, Tamoghna Roy, and T. Charles Clancy. 2018. Over-the-Air Deep Learning Based Radio Signal Classification. *IEEE Journal of Selected Topics in Signal Processing* 12, 1 (2018), 168–179. doi:10.1109/JSTSP.2018.2797022

[11] Guillem Reus-Muns and Kaushik Roy Chowdhury. 2021. Classifying UAVs With Proprietary Waveforms via Preamble Feature Extraction and Federated Learning. *IEEE Transactions on Vehicular Technology* 70, 7 (2021), 6279–6290. doi:10.1109/TVT.2021.3081049

[12] Kunal Sankhe, Mauro Belgiovine, Fan Zhou, Luca Angioloni, Frank Restuccia, Salvatore D'Oro, Tommaso Melodia, Stratis Ioannidis, and Kaushik Chowdhury. 2019. No radio left behind: Radio fingerprinting through deep learning of physical-layer hardware impairments. *IEEE Transactions on Cognitive Communications and Networking* 6, 1 (2019), 165–178.

[13] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. 2017. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*. 618–626.

[14] Guanxiong Shen, Junqing Zhang, Alan Marshall, Linning Peng, and Xianbin Wang. 2021. Radio frequency fingerprint identification for LoRa using deep learning. *IEEE Journal on Selected Areas in Communications* 39, 8 (2021), 2604–2616.

[15] Szymon Szott, Katarzyna Kosek-Szott, Piotr Gawłowicz, Jorge Torres Gómez, Boris Bellalta, Anatolij Zubow, and Falko Dressler. 2022. Wi-Fi meets ML: A survey on improving IEEE 802.11 performance with machine learning. *IEEE Communications Surveys & Tutorials* 24, 3 (2022), 1843–1893.

[16] Zhihao Tang, Mingliang Tao, Jia Su, Yanyun Gong, Yifei Fan, and Tao Li. 2021. Data Augmentation for Signal Modulation Classification using Generative Adverse Network. In *2021 IEEE 4th International Conference on Electronic Information and Communication Technology (ICEICT)*. 450–453. doi:10.1109/ICEICT53123.2021.9531296

[17] Inc. The MathWorks. 2024. *Signal Processing Toolbox*. The MathWorks, Inc., Natick, Massachusetts. https://www.mathworks.com/products/signal.html

[18] Sage Trudeau. 2024. RF-Models-CAM-Evaluation. https://github.com/Trudes808/RF-Models-CAM-Evaluation Accessed: 2024-11-26.

[19] Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *Journal of machine learning research* 9, 11 (2008).

[20] Haofan Wang, Zifan Wang, Mengnan Du, Fan Yang, Zijian Zhang, Sirui Ding, Piotr Mardziel, and Xia Hu. 2020. Score-CAM: Score-weighted visual explanations for convolutional neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*. 24–25.

[21] J. Zeng and Others. 2019. An end-to-end deep learning approach for polyphase code recognition in radar ESM systems. In *IEEE Access*, Vol. 7. 18619–18628.

[22] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. 2016. Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2921–2929.

Received 6 April 2024