Anomaly Transmitter Recognition and Tracking

Tianyi Zhao*, Shamik Sarkar†, Yuan Tian* and Danijela Cabric*
*Electrical and Computer Engineering Department, University of California, Los Angeles, USA
†Department of Electronics and Communications Engineering, Indraprastha Institute of Information Technology Delhi, India
Email: zhaotianyi@ucla.edu, shamik@iiitd.ac.in, yuant@ucla.edu, danijela@ee.ucla.edu

***Abstract*—Device authentication and identification are important to ensure security in spectrum access and management. While prior works have studied radio frequency fingerprinting for such purpose and demonstrated its effectiveness, most works have been focusing on closed-set classification, where only a known set of authorized transmitters appears to be identified. However, unauthorized transmitters can also exist in a designated band. Therefore, recognizing and tracking those unauthorized transmitter behaviors is necessary to protect spectrum security. This work investigates this problem and proposes an Anomaly Recognition and Tracking (ART) framework. The ART framework first learns a closed-set classifier between the authorized transmitters and modifies the classifier to detect anomaly signals. Then, the framework collects the detected anomaly signals and performs unsupervised clustering to assign labels to the predicted anomaly transmitters. Finally, the framework incrementally learns the anomaly transmitter features with limited signals and updates the classifier accordingly. We evaluate our proposed framework with a WiFi dataset and show that with 10% false alarm rate, it can detect more than 99% anomaly signals. Moreover, our proposed framework can distinguish between and learn the features of different anomaly transmitters with as few as 10 signals received from each anomaly transmitter. Finally, the framework can update the classifier, track the anomaly transmitters, and classify among all authorized and anomaly transmitters with more than 99% accuracy.**

***Index Terms*—Radio frequency fingerprinting, deep learning, open-set recognition, anomaly transmitter recognition**

I. Introduction

A. Motivation

Spectrum sharing (SS) is a promising solution to address spectrum inefficiency and maximize spectrum utilization. SS technology allows unlicensed or secondary users (SUs) to opportunistically access the licensed bands, as long as they do not cause harmful interference to licensed or primary users (PUs) [1]. One example is the Citizen's Broadband Radio Service (CBRS) band. Established by the Federal Communications Commission (FCC) in 2015, the CBRS band ($3.55-3.70$ GHz) has a three-tiered access and authorization framework to accommodate shared federal and non-federal use of the band [2]. A frequency coordinator Spectrum Access System (SAS) manages the access and operations in the CBRS band based on the information from Environmental Sensing Capability (ESC), which monitors the band.

In any SS system, security issues are of great concern and impose unique challenges [3]. Therefore, an integral part of a spectrum-sharing system is spectrum monitoring. Spectrum monitoring aims to continuously determine whether one or more radio frequency (RF) transmitters are transmitting beyond their authority. The meaning of the word 'authority' can be diverse and specific to certain systems. Examples include unauthorized access to a frequency band, lower-tier users emulating higher-tier users, secondary transmitters not backing off in the presence of primary users, etc. For a given band and a set of legitimate transmitters, all of these monitoring problems can be solved if we can achieve the following two capabilities in a monitoring system.

- *Anomaly transmitter recognition:* Individually recognize all authorized transmitters and flag them as anomalous transmitters if not identified as authorized.
- *Anomaly transmitter tracking:* If anomalous transmitters are detected, track the subsequent activities of each individual anomalous transmitter.

An effective tool for anomaly transmitter recognition is RF fingerprinting, which has been proposed to enhance the security of wireless networks [4]. The idea of RF fingerprinting is to extract and leverage the additionally embedded information in the distorted RF signals, where the distortions come from transceiver hardware imperfections and wireless channels [5]. Therefore, RF fingerprinting essentially exploits the physical features of the transmitter circuitries, and is a physical layer authentication (PLA) technique. As a result, RF fingerprinting has the advantage of being robust against spoofing attacks [6] and being flexible with different wireless protocols. Prior works exploiting RF fingerprinting for transmitter identification have obtained inspiring results. For example, the authors in [7] showed the feasibility and effectiveness of RF fingerprinting to identify transmitters using raw I/Q data with different protocols and modulation schemes. Therefore, RF fingerprinting is able to augment existing security protocols in not only SS systems but any scenarios where transmitter authentication is required.

Many RF fingerprinting works focus on closed-set classification, where the task is to recognize a transmitter among a known set of authorized transmitters. While closed-set classification reasonably simplifies the problem formulation, it is inapplicable in many realistic deployments where unexpected/anomalous transmitters can be present. For example, in primary user emulation attacks, anomalous transmitters can mimic incumbent transmitters to enforce the frequency coordinator to vacate the specific band [8]. To address such potential attacks, anomaly transmitter recognition, also known

This work was supported by SpectrumX, which is an NSF Spectrum Innovation Center funded via Award 2132700.

as open-set recognition, is a crucial task.

Open-set recognition enables recognizing anomaly transmitters in addition to recognizing legitimate transmitters. Studies have been conducted in this area. For example, the authors in [9] examine the open-set transmitter recognition problem and discuss several approaches for solving it. However, none of the prior works analyzed the detected anomaly transmitters further.

In this work, we go beyond anomaly detection by identifying and remembering the anomaly transmitters to track their behaviors. By tracking their behaviors, we can potentially gain more insights into the possible threats presented by the anomaly transmitters and react accordingly. For example, in the context of the CBRS band, if we detect unauthorized transmitters causing interference to primary users, such as navy radar, then it is worth understanding whether the source of interference is one base station (BS) or multiple BSs and whether there are repeated violations by the same set of offenders. Therefore, it is crucial to monitor the ongoing transmissions so that the coordinator SAS can detect, classify, and localize the transmitters.

In summary, anomaly transmitter *recognition* enables us to take security measures against unauthorized transmitters. On top of that, *tracking* the behaviors of those anomalous transmitters allow further analysis, provide additional insights about the ongoing attacks and assist in future defenses. In this work, we study the anomaly transmitter recognition and tracking problem and propose an online incremental learning framework to continuously identify and remember anomaly transmitters in a given network. This is a non-trivial problem involving many challenges, such as unsupervised, few-shot, and incremental learning. To the best of the authors' knowledge, this is the first work considering unsupervised clustering for the task of labeling anomaly transmitters. Moreover, we propose a novel method to remove the false positive signals, which are the wrongly rejected signals from authorized transmitters, to improve the anomaly label assignment performance.

B. Contributions

Our contributions can be summarized as follows:

- We formulate the anomaly transmitter recognition and tracking problem. Based on the idea of open-set recognition, the problem consists of three stages. First, we want to classify and admit authorized transmitters. Second, we want to recognize and reject anomalous signals. Finally, we want to track the anomaly transmitters by identifying them in future transmissions.
- We develop a framework called Anomaly Recognition and Tracking (ART) to solve the above problem. Initially, ART is trained with signals from authorized transmitters for open-set recognition in the first stage. After the training, it can be deployed to reject anomalous signals while classifying authorized transmitters. Then, unsupervised clustering is performed on the signals, and the clustering results are processed for anomaly transmitter label assignment. Finally, the labeled anomalous signals are used in the last stage, where the framework incrementally learns the anomaly transmitters. By doing so, the ART framework is able to continuously update its knowledge so as to identify the anomaly transmitters and track their occurrences.
- We conduct extensive experiments to evaluate the proposed ART framework with real captured WiFi signals. First, we examine and compare different candidate algorithms in ART. We find that using Openmax [10] for anomaly detection, HDBSCAN [11] for anomaly label assignment, and transfer learning for incremental learning yields the best performance. Then, we conduct experiments to validate the ART framework. Our experimental results show that with 10% false alarm rate, ART is able to detect more than 99% anomalous signals. Moreover, it is able to distinguish between and learn the features of different anomaly transmitters with as few as 10 signals received from each anomaly transmitter. Finally, the framework can update the classifier, track the anomaly transmitters, and classify among all authorized and anomaly transmitters with more than 99% accuracy.

C. Organization of the paper

The rest of the paper is organized as follows. Section II formulates the anomaly transmitter recognition and tracking problem. Section III explains the proposed ART framework to solve the problem. Section IV discusses the experimental setup to evaluate ART and presents results. Finally, section V concludes the paper.

II. Problem Formulation

In this section, we describe the formulation of the anomaly transmitter recognition and tracking problem. As discussed in Section I-A, the problem consists of open-set recognition and anomaly transmitter tracking. We explain the problem using Figure 1, where both authorized and unauthorized transmitters exist in a network. The coordinator should be able to accomplish the following three tasks:

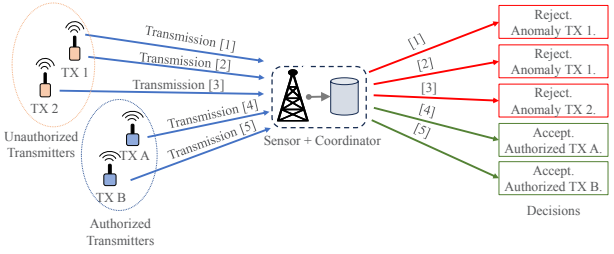


Fig. 1. Illustration of the anomaly transmitter recognition and tracking problem. A blue arrow indicates a transmitted signal, and the number indicates the order of transmission. Red and green arrows point to the decision made by the coordinator, and the numbers indicate the corresponding transmissions.

1) **Classify authorized transmitters**: The first task is to classify between the authorized transmitters, essentially a closed-set classification. Figure 1 illustrates this task with the first recognition and the acceptance of transmitters A and B.

2) **Recognize anomaly signals**: The second task is to detect and reject the anomaly signals sent from unauthorized transmitters. This task, along with the first task form the open-set recognition problem. As shown in Figure 1, this task is illustrated with the rejection of the anomaly signals.
3) **Track anomaly transmitters**: The third task is to remember previously seen unauthorized transmitters and identify them in future occurrences. This final task enables the tracking of the anomaly transmitter behaviors. As shown in Figure 1, this task is illustrated by identifying unauthorized transmitters 1 and 2 between all received anomaly signals.

In this work, we investigate the above tasks. Our goal is to build a coordinator f_c that can take any wireless signal x as an input and outputs a label $\hat{y} = f_c(x)$. Ideally, for an SS network with n authorized transmitters, the coordinator should predict as follows:

$$\hat{y} = \begin{cases} i, \text{ if } x \text{ is from the } i\text{-th authorized transmitter} \\ n+j, \text{ if } x \text{ is from the } j\text{-th anomaly transmitter} \end{cases}$$

where $i \in \{1, 2, \ldots, n\}$ and $j \in Z^+$.

The challenges and our solutions are discussed in the next section in detail.

III. Approach

In this section, we explain the proposed ART framework and discuss the detailed implementation of each stage in the framework.

A. ART Framework

To solve the anomaly transmitter recognition and tracking problem explained in Section II, we propose the ART framework as shown in Figure 2. The framework consists of two phases: offline training and online updating.

The offline training phase requires a sensor to collect training signals from the authorized transmitters. Then, deep learning is utilized to build a classifier with $n+1$ classes, where n is the number of authorized transmitters, and the $(n+1)^{th}$ class represents the anomaly. The resulting classifier is an open-set authenticator, which can detect anomaly signals as well as classify legitimate signals, thus addressing the open-set recognition problem. It should be noted that the ART framework requires prior information from all authorized transmitters to build the classifier. The detailed implementation of the classifier is explained in Section III-B.

After the one-time offline training phase, ART allows online updating during normal operation. The online updating phase has a three-stage architecture: anomaly detection, anomaly label assignment, and anomaly feature learning, respectively.

In the online updating phase, the sensor will receive signals from both authorized and anomaly transmitters during its operations. Therefore, it will continuously send the received signals to all three stages of ART. First, the open-set authenticator will perform anomaly detection by determining the transmitter labels for each input signal, as explained in detail in Section III-B. Then, the anomaly label assignment stage takes the labels produced by the previous stage as input in addition to the received signals. In the anomaly label assignment stage, unsupervised clustering is performed to assign labels to the transmitters associated with the anomaly signals. The details are explained in Section III-C. Finally, the anomaly feature learning stage uses the anomaly signals and their assigned labels to learn their features and update the open-set authenticator accordingly, as explained in detail in Section III-D. As a result, the framework can recognize the anomaly transmitters and track their future occurrences accordingly.

B. Anomaly Detection

The anomaly detection stage essentially solves the open-set recognition problem by building a classifier with $n+1$ classes, where n is the number of known classes. We implement such a classifier based on the Openmax method in [10]. We use this method because does not require any unauthorized transmitters in the training set or heavy training overheads.

First, a closed-set authenticator is trained with the training signals from n authorized transmitters using a cross-entropy loss. The authenticator consists of two components, a feature-extractor, and a classifier, as shown in Figure 3. The inputs to the feature extractor are equalized 400×2 I/Q samples, and the outputs are feature vectors of length T. Those feature vectors are then inputted into the classifier. The last dense layer in the classifier will have exactly n activations. Let v_i be the activation of class i where $i \in \{1, 2, \ldots, n\}$, and $v_i > v_j$ for all $j \in \{1, 2, \ldots, n\}, i \neq j$. Since the last layer before the output is a softmax layer, which maps the activations into probabilities for the corresponding classes, the largest activation thus corresponds to the highest probability. Therefore, a signal will either be classified as class i or $n+1$, the anomaly class. Accordingly, we can generate the activation for classes i and $n+1$ as follows:

$$v_i' = v_i w_i \tag{1}$$

$$v_{n+1}' = v_i(1 - w_i) \tag{2}$$

where v_i represents the original activation for class i, v_i' represents the modified activation for class i and w_i represents the confidence in class i, as explained below.

The confidence w_i can be interpreted as the similarity between the current activation v_i and the collection of the activations of training signals from class i. Mathematically, w_i is the probability that the activation v_i does not belong to the tail in the distribution of the activations of training signals from class i. Therefore, w_i is computed based on Weibull distribution as follows:

$$w_i = 1 - \exp\left(-\left(\frac{|v_i - \mu_i|}{\lambda_i} \right)^{k_i} \right) \tag{3}$$

where λ_i and k_i are the Weibull distribution parameters calculated from the activations of the training signals from class i, and μ_i is the average activation of the training signals from class i.

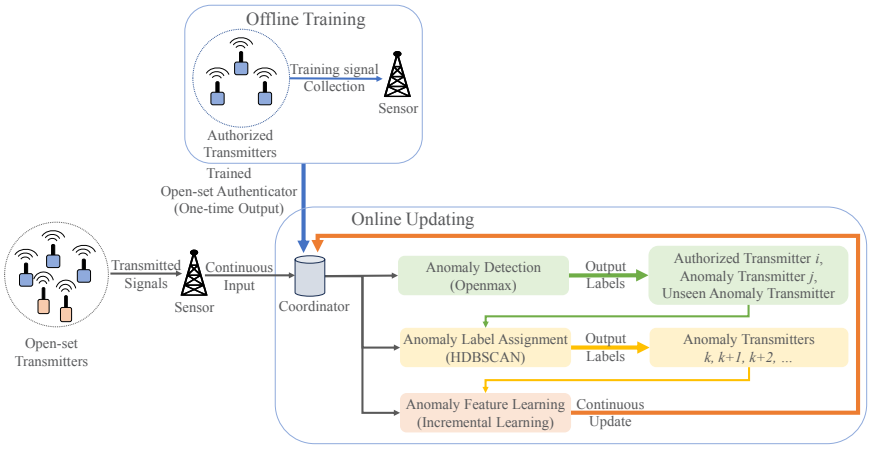
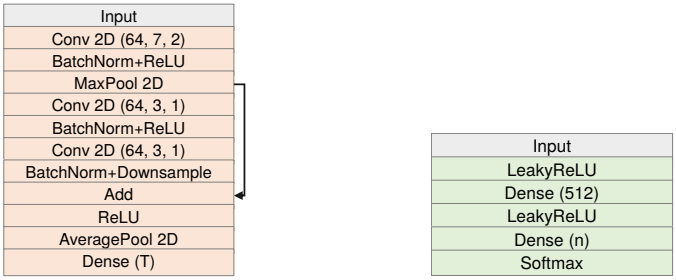


Fig. 2. Illustration of the ART framework. An open-set authenticator is trained in the offline phase and deployed on the coordinator in the online phase. During normal operations, the sensor continuously sends I/Q samples of received signals to the coordinator. The coordinator is able to classify signals from authorized transmitters, recognize signals from anomaly transmitters, and track the behaviors of the anomaly transmitters by online learning their features.



(a) Feature extractor. (b) Classifier.

Fig. 3. The architecture of a closed-set authenticator. The outputs of the feature extractor are feature vectors of length T and inputted into the classifier.

Finally, the classifier will determine the label of the input signal by comparing v'_i and v'_{n+1}. If $v'_i > v'_{n+1}$, the signal is classified as from transmitter i, and if $v'_i \leq v'_{n+1}$, the signal is classified as from an anomaly transmitter.

C. Anomaly Label Assignment

After detecting anomaly signals, the framework performs Hierarchical Density-Based Spatial Clustering of Applications with Noise (HDBSCAN) [11] to assign the labels for them. DBSCAN [12] is a density-based unsupervised clustering algorithm that does not require the number of clusters a priori. DBSCAN defines any two data points p and q as density-connected with respect to ϵ and ρ if they satisfy Equation (4) as follows:

$$p \in N_\epsilon(q) \text{ and } |N_\epsilon(q)| \geq \rho \tag{4}$$

where ρ is the minimum number of data points in a cluster, ϵ is a distance parameter and $N_\epsilon(q)$ is defined in Equation (5).

$$N_\epsilon(q) = \{\text{dist}(p, q) \leq \epsilon\} \tag{5}$$

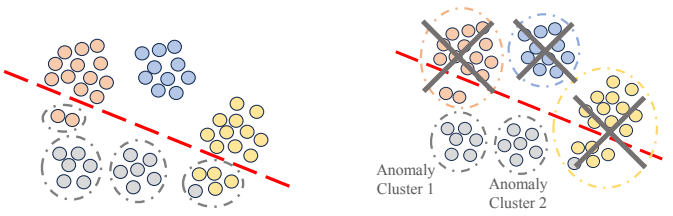
Accordingly, any two points p and q are density-reachable from each other if they are both density-connected to another point o. Then, DBSCAN classifies any data points p and q to the same cluster if p is density-connected to q or density-reachable from q.

Based on DBSCAN, HDBSCAN expands clusters by iteratively merging the data points with high density within the same neighborhood, where the distance ϵ is automatically chosen by the algorithm. Since the significance of distances between data points degrades in high-dimensional space, we want to reduce the dimensionality of the inputs into the clustering algorithm. Therefore, the clustering is performed on the signal features extracted by the feature extractor shown in Figure 3 (a), instead of the raw I/Q samples from the received signals.

However, simply performing clustering on the detected anomaly signals may lead to false labels. This is because the anomaly detection stage may yield false positive results, where the signals from authorized transmitters are wrongly rejected as anomaly signals. Therefore, a few legitimate signals are wrongly rejected after anomaly detection and thus undergo the clustering stage together with the real anomaly signals. As a result, there will exist a few clusters with the majority of signals being legitimate signals, as shown in Figure 4 (a). When the number of authorized transmitters increases, those false labels can be more and more confusing. Therefore, we propose the following approach to remove the misleading false labels, as shown in Figure 4 (b).

First, HDBSCAN is performed on features extracted from all received signals instead of only on rejected signals. Then, if a cluster consists of more than τ accepted legitimate signals, the framework will remove all the signals in that cluster. The rationale is that when a signal from an authorized transmitter j is wrongly rejected, its feature vector is still closest to the center of the cluster associated with transmitter j. This assumption is reasonable because even if a rejected signal belongs to the tail distribution of its true class j, its feature vector is still closest to the center of cluster j rather than any other cluster $i \neq j$. Therefore, the clustering algorithm can still group the wrongly rejected signals with the other correctly classified signals from transmitter j. τ is experimentally chosen to be 80%.

Finally, the clusters with less than $\rho = 5$ samples are



(a) Visualization of clustering results only on rejected signals.

(b) Visualization of clustering results on all received signals.

Fig. 4. Illustrations of anomaly transmitter detection based on clustering. Each circle represents a feature vector of a received signal. Red, blue and yellow circles represent signals from three authorized transmitters, and gray represents anomalous signals. The dotted red line stands for anomaly detection, where the rejected signals are placed on the lower side, and dotted circles represent the clustering results. In (a), the clustering is performed only on rejected signals, so falsely rejected authorized signals can compose false anomaly clusters. In (b), the clustering is performed on all signals, and the clusters whose major components are accepted are removed, and the remaining clusters are detected as signals from distinct anomaly transmitters.

ignored, where ρ is the minimum number of signals for a cluster not to be considered as noise. A noise cluster consists of signals whose feature vectors are far from any other clusters. One possible reason is that some signals are distorted due to unpredictable variations, such as channels. In such cases, the resulting feature vectors can be misleading and are thus discarded. Another possible reason is that a new anomaly transmitter has just started to transmit and does not provide enough useful signals yet. If the anomaly transmitter keeps transmitting, the size of the cluster will increase and become significant enough to be considered as a new class in some time. If it stops transmitting, it will not affect the monitored spectrum much and thus can be safely ignored.

After the removal of authorized clusters and noise clusters, the remaining signals are considered significant anomaly signals. These signals are assigned labels given by the clustering results and will be used for anomaly feature learning in the next stage.

D. Anomaly Feature Learning

Finally, based on the clustering result and assigned labels, the framework learns the anomaly transmitter and incorporates that knowledge into the open-set authenticator. There exist a few challenges in this learning stage. First, the learning process should be efficient and lightweight since it is happening repeatedly in real time. At the same time, only limited anomaly signals may be available for learning since the anomaly transmissions are usually sparse, where a few-shot learning problem needs to be addressed. Finally, the catastrophic forgetting problem [13] should be considered, and learning new transmitters should not lead to forgetting learned authorized transmitters.

The idea of transfer learning [14] can be applied to address this problem. As shown in Figure 3, the feature extractor is well-trained in the offline training phase. By exploiting the feature extractor, we can update the classifier with limited training signals. In particular, we freeze the feature extractor and train a new classifier with a modified number of output classes. The modified output classes consist of all known transmitters and the newly recognized anomaly transmitters. At the same time, to avoid catastrophic forgetting, we include a subset of offline training signals as well as the newly labeled signals in the training set. The subset size of the offline training signals is determined by the assumption of the number of anomaly signals to address the imbalanced training set due to the limited anomaly transmissions. Finally, the classifier is able to be online updated, because the updating process only involves training a classifier with two dense layers on a small number of signals.

Once the classifier is updated, the open-set authenticator will be modified accordingly, as explained in Section III-B. Hence, the ART framework is able to continuously update while operating, as shown in Figure 2. We evaluate the ART framework in Section IV.

IV. Evaluations

In this section, we first introduce the dataset used in the experiments and then explain the experimental setup. Then, we examine each stage in ART to evaluate the framework.

A. Dataset and Experimental Setup

For evaluations, we use the WiSig dataset [15], because it includes signals sent from a large number of transmitters. The dataset contains high SNR WiFi signals captured over the air. It should be noted that while we use WiFi signals to evaluate the ART framework, it can be flexibly applied to different wireless protocols, an advantage of RF fingerprinting.

As shown in [15], the variations in channels and receivers can introduce additional noise into the data distribution. Therefore, for the purpose of the experiments in this work, we use the transmissions from 30 Atheros WiFi transmitters, which are received by a single USRP receiver in a single day.

Following our prior work [5], we use the received samples of preambles in a WiFi packet for all signals and pre-process the data with channel equalization. Each signal has 400×2 I/Q samples, including the Legacy Long Training Field (L-LTF) and Legacy Short Training Field (L-STF). After extracting the preambles, we first estimate and correct the frequency offset in L-STF, then estimate and equalize the channel using minimum mean-square error (MMSE) on L-LTF, and finally reapply the frequency offset to the signal. The signal processing for detection and channel estimation is applied using MATLAB R2019b WLAN toolbox with the default parameters, and the detailed pre-processing procedure can be found in [5].

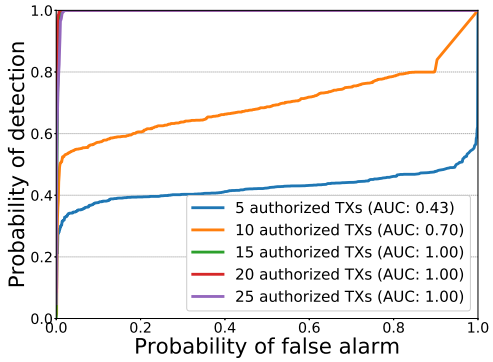
To evaluate the proposed framework, we conduct experiments to examine each stage accordingly. The purposes and factors considered for the experiments are presented in Table I. For all the experiments, we consider 3 anomaly transmitters, use 5 different randomized combinations of transmitters, and report the averaged performance.

B. Anomaly Detection

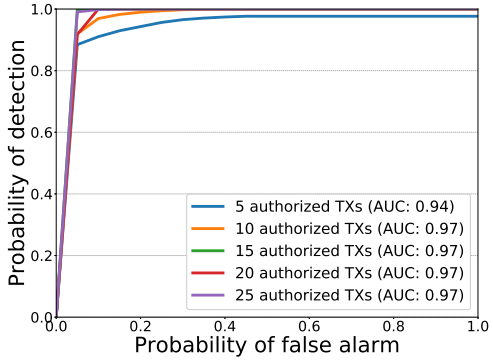
For the evaluations in this section, we consider 3 anomaly transmitters and different numbers of authorized transmitters.

TABLE I
PURPOSES AND CONSIDERED IMPACT FACTORS OF THE EXPERIMENTS.

Stage	Purpose	Considered Impact Factor
Anomaly Detection	Find best anomaly detection algorithm	Number of authorized transmitters
Anomaly Label Assignment	Find best unsupervised clustering algorithm	Number of authorized transmitters
Anomaly Label Assignment	Validate proposed method to remove false positive signals	Number of authorized transmitters
Anomaly Feature Learning	Validate incremental learning method	Number of received signals per anomaly transmitter
Anomaly Feature Learning	Examine overall ART performance	Number of received signals per anomaly transmitter



(a) Area under ROC curve for OVA.



(b) Area under ROC curve for Openmax.

Fig. 5. Comparison of OVA and Openmax for anomaly detection evaluation.

First, we consider two candidate algorithms for anomaly detection: Openmax, which is explained in Section III-B, and one-versus-all (OVA) [16]. OVA trains a distinct classifier for each authorized transmitter j, and each classifier predicts the probability P_j such that the input signal belongs to j. Therefore, we will have n classifiers and n probabilities P_j, where n is the number of authorized transmitters. Then, if the corresponding classifier for every transmitter j predicts an input signal as not belonging to class j, the signal is detected as an anomaly.

The area under Receiver Operating Characteristic (ROC) curves is evaluated for both algorithms in Figure 5. In an ROC curve, the true positive rate of detecting anomaly signals (probability of detection) is evaluated against the false positive rate (probability of false alarm) by varying the threshold values. It can be observed that the anomaly detection performance improves as the number of authorized transmitters increases for both algorithms. This is a reasonable trend because more authorized transmitters can provide more training signals and, thus, better generalization capability for the classifier. Moreover, for a smaller number of authorized transmitters, the anomaly detection performance of Openmax is better than that of OVA, and for larger number of authorized transmitters, the anomaly detection performances are comparable between the two methods. This might result from the differences in decision boundaries between OVA and Openmax. When the number of authorized transmitters is small, the binary classifiers in OVA result in loose decision boundaries for the signals. However, as Openmax detects anomalies based on the distribution of each separate class, the decision boundary is less affected by the number of authorized transmitters. Moreover, it can be observed from Figure 5 (b) that with a 10% false alarm rate, Openmax can achieve more than 99% anomaly signal detection rate.

C. Anomaly Label Assignment

To evaluate the clustering performance, we still consider 3 anomaly transmitters and different numbers of authorized transmitters. The clustering is based on the anomaly detection results of Openmax algorithm with a 10% false alarm rate.

After anomaly detection, we first compare the candidate unsupervised clustering algorithms for anomaly label assignment. The candidate algorithms are HDBSCAN, as discussed in Section III-C, Mean-Shift clustering [17], and Agglomerative clustering [18]. The Mean-Shift algorithm clusters by iteratively updating the candidates for centroids to be the mean of the points within a given region, and the Agglomerative algorithm clusters by recursively merging smaller clusters. o evaluate the clustering performance, we apply the V-Measure [19] score as the metric, which is shown in Equation (6).

$$V = \frac{(1+\beta) * h * c}{\beta * h + c} \tag{6}$$

In Equation (6), $\beta = 1$ is a hyper-parameter that balances the importance of h and c, where h represents homogeneity and c represents completeness. Given a set of classes, $C = \{c_i | i = 1, \ldots, n\}$ and a set of clusters, $K = \{k_i | i = 1, \ldots, m\}$, h and c are defined as following:

$$h = \begin{cases} 1 & , \text{if } H(C,K) = 0 \\ 1 - \frac{H(C|K)}{H(C)} & , \text{otherwise} \end{cases} \tag{7}$$

$$c = \begin{cases} 1 & , \text{if } H(K,C) = 0 \\ 1 - \frac{H(K|C)}{H(K)} & , \text{otherwise} \end{cases} \tag{8}$$

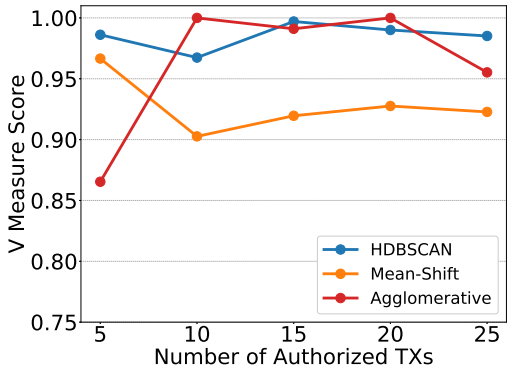


Fig. 6. Comparison of HDBSCAN, Mean-Shift and Agglomerative clustering in detected anomaly signals.

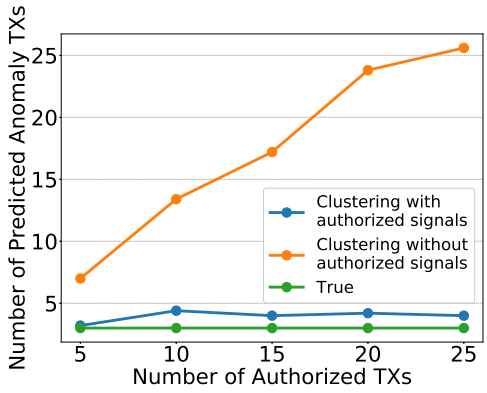


Fig. 7. Evaluation of clustering performance with and without signals from authorized transmitters. HDBSCAN is used to perform the clustering.

V-Measure score ranges from 0 to 1, and a higher score indicates a better clustering result.

The clustering performance is only evaluated on the detected anomaly signals after the removal of noise clusters and authorized clusters, as explained in Section III-C, and the comparison result is shown in Figure 6.

From Figure 6, we can observe that HDBSCAN yields a consistent and better clustering performance than the other two candidate algorithms. Unlike the other two algorithms, HDBSCAN requires a lower number of hyperparameters. This relaxation potentially enables the performance to be less dependent on the signal features being clustered and, thus, to have more consistent results. Therefore, we choose HDBSCAN as the algorithm for anomaly label assignment.

After deciding the clustering algorithm, we conduct the experiment to evaluate the approach in the anomaly label assignment stage. To be more specific, we compare the performances between clustering with and without the signals from authorized transmitters, as shown in Figure 7.

In Figure 7, it can be observed that if we perform clustering without signals from authorized transmitters, the number of detected anomaly transmitters will be dependent on the number of authorized transmitters. Because all authorized transmitters have signals that are wrongly rejected, the rejected signals

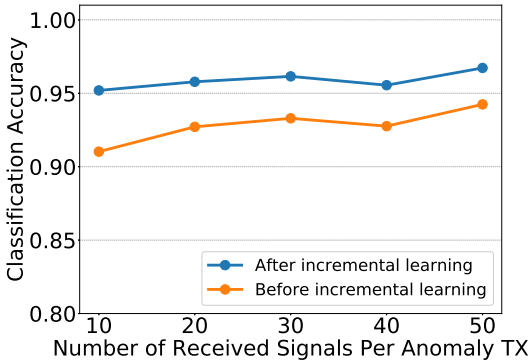


Fig. 8. Comparison of anomaly transmitter tracking performance before and after incremental learning, evaluated by classification accuracy among all detected anomaly signals.

from the same authorized transmitter are likely to be clustered together. Thus, those signals can result in an overestimated number of anomaly transmitters and confuse the framework. In conclusion, it can be shown that our proposed method, which performs clustering with signals from authorized transmitters and removes the noisy clusters, can significantly help to improve the anomaly label assignment performance. For the rest of the experiments, we use our proposed anomaly label assignment method with HDBSCAN as the clustering algorithm.

D. Anomaly Feature Learning

Next, we evaluate the anomaly transmitter tracking performance. For the evaluations, we consider 3 anomaly transmitters, 25 authorized transmitters, and different numbers of received signals from each anomaly transmitter. The input data for the feature learning are based on the anomaly labels assigned by HDBSCAN together with the proposed false positive signal removal method, where the anomaly label assignment is based on the anomaly detection results of Openmax algorithm with a 10% false alarm rate. We compare the performance before and after the incremental learning stage, where the performance is evaluated by the classification accuracy between the detected anomaly signals. The classification accuracy before the incremental learning stage is calculated based on the correctness of the labels assigned by the clustering algorithm. The comparison result is shown in Figure 8.

In Figure 8, there is an improvement in the accuracy after incremental learning with respect to with clustering only. This improvement in the classification accuracy can be the result of the additional information provided by the offline training signals, which the incremental learning stage utilizes to update the classifier. This improvement demonstrates the necessity of the incremental learning stage. Moreover, it can also be observed that the accuracy after incremental learning follows the same trend as the accuracy before incremental learning, which shows that the quality of clustering results will indeed affect the final performance.

Finally, we consider the overall recognition and tracking performance after incremental learning. For that purpose, we

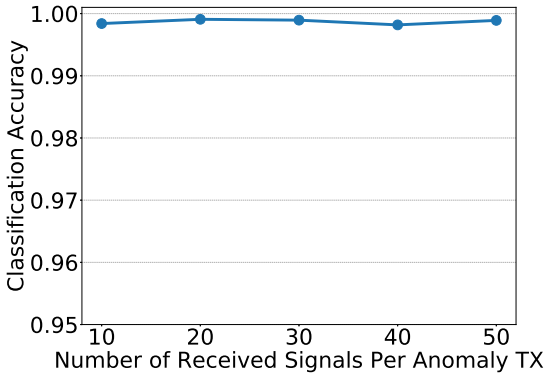


Fig. 9. Performance of the ART framework, evaluated by the classification accuracy between signals from all authorized and incrementally learned anomaly transmitters.

evaluate the classification accuracy among the signals from all authorized and anomaly transmitters that have appeared in normal operation. The result is shown in Figure 9.

In Figure 9, we can observe that the classification accuracy is consistently higher than 99%. This result shows that incremental learning does not lead to any performance degradation in identifying authorized transmitters. Overall, the ART framework is able to effectively recognize and track anomaly transmitters after receiving as few as 10 signals from each anomaly transmitter while being able to accurately identify authorized transmitters.

V. Conclusion

In this work, we considered the security concerns in spectrum sensing. Specifically, we discussed that to ensure spectrum security. It is crucial to enforce transmitter authority and track malicious anomaly transmitter activities for further analysis. Therefore, we formulated the problem of recognizing and tracking anomaly transmitters. To address this problem, we proposed the ART framework and evaluated it with a real captured WiFi dataset. We found that with 10% false alarm rate, ART can detect more than 99% anomaly signals. With 10 signals received from each anomaly transmitter, the framework can track the anomaly transmitters with more than 95% accuracy while being able to classify among the authorized transmitters with more than 99% accuracy.

While the performance in current evaluations is good, it still has some limitations. For example, the current ART framework requires prior information about all authorized transmitters. As a result, while it can incrementally learn features about new transmitters, it may not perfectly handle a dynamic network condition yet. Moreover, the current anomaly detection method depends on statistical modeling of signal feature distributions. Therefore, it might fail if the distributions become unstable or inconsistent due to reasons such as varying channels or signal-to-noise (SNR) ratios. As a result, more work will need to be done to ensure the robustness of the anomaly detection in different scenarios. At the same time, we can observe that the final tracking performance depends on the anomaly label assignment by clustering. Therefore, it is also important to develop more reliable clustering algorithms, as well as robust incremental learning algorithms against noisy label assignments. In future works, we will look into these aspects and improve the framework's robustness and reliability in different scenarios.

References

[1] S. Shi, Y. Xiao, W. Lou, C. Wang, X. Li, Y. T. Hou, and J. H. Reed, "Challenges and New Directions in Securing Spectrum Access Systems," *IEEE Internet of Things Journal*, vol. 8, no. 8, pp. 6498–6518, 2021.

[2] "3.5 GHz Band Overview." https://www.fcc.gov/wireless/bureau-divisions/mobility-division/35-ghz-band/35-ghz-band-overview, 2023.

[3] Q. Wang, H. Sun, R. Q. Hu, and A. Bhuyan, "When Machine Learning Meets Spectrum Sharing Security: Methodologies and Challenges," *IEEE Open Journal of the Communications Society*, vol. 3, pp. 176–208, 2022.

[4] O. Ureten and N. Serinken, "Wireless security through RF fingerprinting," *Canadian Journal of Electrical and Computer Engineering*, vol. 32, no. 1, pp. 27–33, 2007.

[5] 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*, pp. 1–1, 2023.

[6] N. Soltanieh, Y. Norouzi, Y. Yang, and N. C. Karmakar, "A Review of Radio Frequency Fingerprinting Techniques," *IEEE Journal of Radio Frequency Identification*, vol. 4, no. 3, pp. 222–233, 2020.

[7] I. Agadakos, N. Agadakos, J. Polakis, and M. R. Amer, "Chameleons' Oblivion: Complex-Valued Deep Neural Networks for Protocol-Agnostic RF Device Fingerprinting," in *2020 IEEE European Symposium on Security and Privacy (EuroSP)*, pp. 322–338, 2020.

[8] A. G. Fragkiadakis, E. Z. Tragos, and I. G. Askoxylakis, "A survey on security threats and detection techniques in cognitive radio networks," *IEEE Communications Surveys Tutorials*, vol. 15, no. 1, pp. 428–445, 2013.

[9] S. Hanna, S. Karunaratne, and D. Cabric, "Open Set Wireless Transmitter Authorization: Deep Learning Approaches and Dataset Considerations," *IEEE Transactions on Cognitive Communications and Networking*, vol. 7, no. 1, pp. 59–72, 2021.

[10] A. Bendale and T. E. Boult, "Towards Open Set Deep Networks," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1563–1572, 2016.

[11] R. J. G. B. Campello, D. Moulavi, A. Zimek, and J. Sander, "Hierarchical Density Estimates for Data Clustering, Visualization, and Outlier Detection," *ACM Trans. Knowl. Discov. Data*, vol. 10, jul 2015.

[12] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise," in *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, KDD'96, p. 226–231, AAAI Press, 1996.

[13] I. J. Goodfellow, M. Mirza, D. Xiao, A. Courville, and Y. Bengio, "An Empirical Investigation of Catastrophic Forgetting in Gradient-Based Neural Networks," 2015.

[14] S. J. Pan and Q. Yang, "A Survey on Transfer Learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.

[15] 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. 22808–22818, 2022.

[16] L. Shu, H. Xu, and B. Liu, "DOC: Deep Open Classification of Text Documents." arXiv:1709.08716, 2017.

[17] D. Comaniciu and P. Meer, "Mean shift: a robust approach toward feature space analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 603–619, 2002.

[18] D. Müllner, "Modern hierarchical, agglomerative clustering algorithms." arXiv:1109.2378, 2011.

[19] A. Rosenberg and J. Hirschberg, "V-Measure: A Conditional Entropy-Based External Cluster Evaluation Measure," in *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)* (J. Eisner, ed.), (Prague, Czech Republic), pp. 410–420, Association for Computational Linguistics, June 2007.